

PSM I in Simple English

27 free Scrum Master lessons, from first steps to exam day

For people whose first language is not English. A complete guide to the Professional Scrum Master I exam and the 2020 Scrum Guide: every lesson has a simple explanation, a picture, a real team decision, the common myths, the exam traps and exam-style questions with answers.

27

lessons

54

case decisions

108

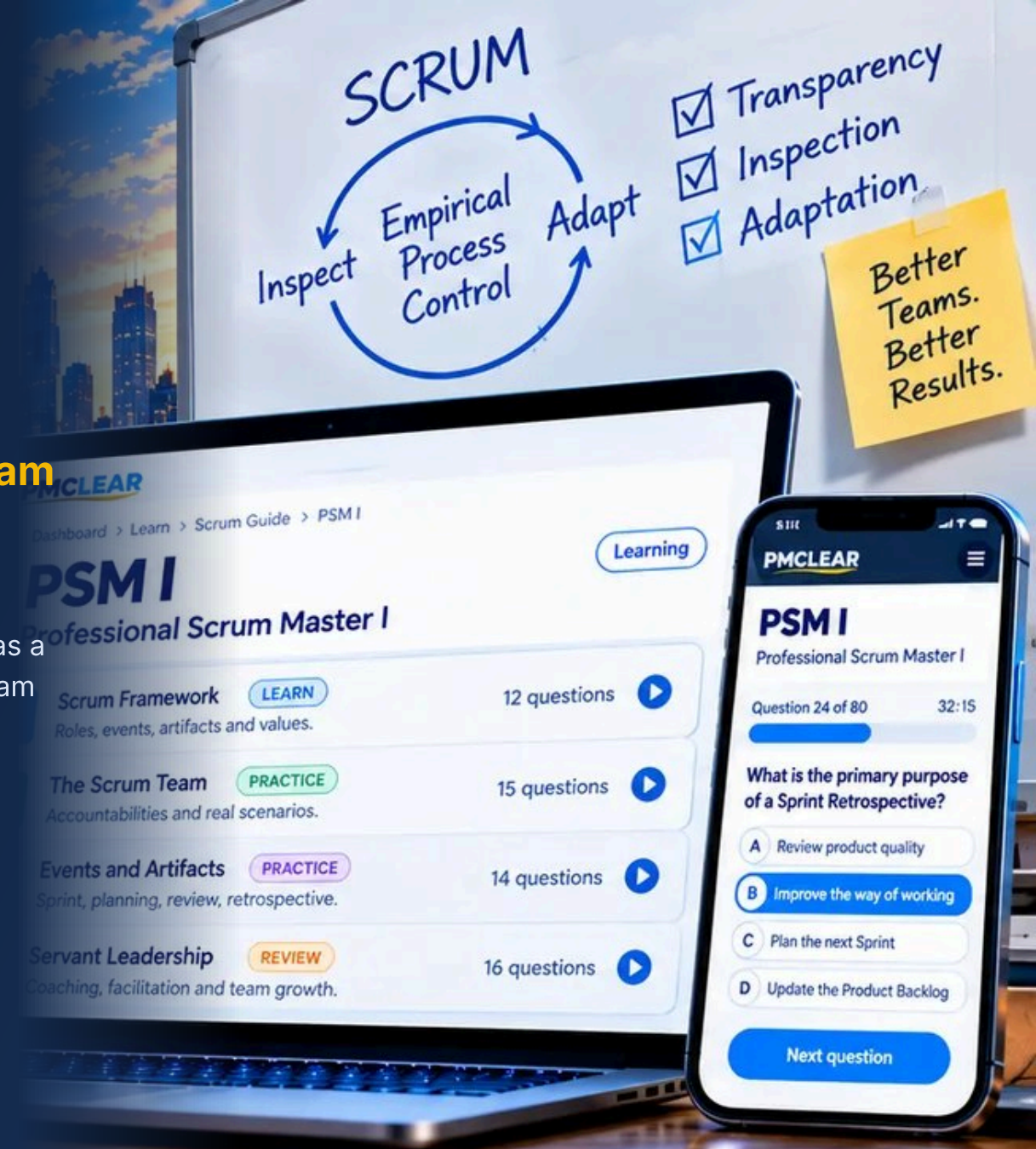
exam-style questions

2020

Scrum Guide

[Read the lessons online →](#)

[Free PSM I practice questions →](#)



What is inside this book click any title to jump to it

Part 0 · Your PSM I exam guide

What PSM I is, format, 2017 vs 2020, timeboxes, 12 rules, how to read a question, a 4-week plan

Toolkit

[Who decides what](#) · [events and artifacts on one page](#) · [glossary](#) · [top exam traps](#) · [exam-day checklist](#) · [progress tracker](#)

Every lesson, 5 pages

- 1 Idea and simple explanation
- 2 Picture and memory card
- 3 Deeper explanation
- 4 Case study: you decide
- 5 Myths, traps and exam questions

Part A · Start

- 0 The PSM I exam and how to study
- 1 Complex work and why Scrum exists
- 2 The language of Scrum

Part B · Foundations

- 3 The framework in one picture
- 4 Empiricism, lean and the three pillars
- 5 Scrum values in real decisions

Part C · The Scrum Team

- 6 The Scrum Team
- 7 Developers
- 8 Product Owner
- 9 Scrum Master
- 10 Self-management and leadership

Part D · Events

- 11 The Sprint
- 12 Sprint Planning
- 13 Daily Scrum
- 14 Sprint Review
- 15 Sprint Retrospective

Part E · Artifacts and commitments

- 16 Product Backlog, refinement and Product Goal
- 17 Sprint Backlog and Sprint Goal
- 18 Increment, release and Definition of Done

Part F · Advanced practice

- 19 Product value, stakeholders and forecasting
- 20 Quality and technical debt
- 21 Many teams, one product
- 22 Scrum in the organization
- 23 Scrum beyond software
- 24 One full Sprint, start to finish

Part G · Exam readiness

- 25 Reading PSM I questions; 2017 vs 2020
- 26 Final review: the hardest distinctions

How to use this book

1. **Read one lesson a day.** Each lesson is five pages: the idea, a picture, the deeper explanation, a real team decision and the exam traps. About 30 minutes.
2. **Decide before you read the answer.** In every case study, cover the green outcome with your hand and choose first. PSM I rewards judgement, not memory.
3. **Answer the four questions, then check.** The answers are at the bottom of the page. If you miss one, reread the memory card.
4. **Read the Scrum Guide itself.** It is short (about 14 pages) and free at scrumguides.org. Read it once now and again in your last week. This book explains it; it does not replace it.
5. **Practise under time pressure.** 80 questions in 60 minutes is fast. Use full timed practice exams before you book the real one.

FREE ACCOUNT

Sign up free and get 3 days of full access

PSM I practice: 1,000 exam-level questions, 80-question timed exams, progress by topic and a "my mistakes" list. Every question is explained in simple English.

Create your free account →

Online version of every lesson

The full lessons on pmclear.com have 12 to 15 questions each, with explanations of why each wrong answer is wrong, plus videos for the first lessons.

pmclear.com/psm/course →

PSM and Professional Scrum Master are trademarks of Scrum.org. PMCLEAR is an independent study resource, not affiliated with or endorsed by Scrum.org. See the small print at the end of this book.

PART 0 · YOUR PSM I EXAM GUIDE

What is PSM I?

PSM I (Professional Scrum Master I) is the first-level Scrum Master certification from **Scrum.org**. It shows that you understand the Scrum framework as the **2020 Scrum Guide** describes it, and can apply it in real team situations.

- **No prerequisites.** No course, degree or work experience is required.
- **Online, in English,** from your own computer, when you are ready.
- **One password = one attempt.** You buy the password from Scrum.org; it does not expire.
- **For life.** The certification does not expire and has no renewal fee.
- **Levels:** PSM I, then PSM II (advanced) and PSM III (distinguished).

Good news for non-native speakers

The Guide is short and its words are exact. Most wrong answers come from reading a word too fast, not from missing knowledge. Every lesson in this book trains the exact words.

Your road to PSM I in 5 steps

1

Learn the Guide

Read the 2020 Scrum Guide once, then lessons 0 to 10 of this book.

2

Understand every part

Lessons 11 to 26: events, artifacts, value, scaling and exam technique.

3

Practise

Answer exam-style questions every day. Learn from every wrong answer.

4

Simulate

Full 80-question exams in 60 minutes. Aim for 90% or more twice in a row.

5

Book and pass

Buy the password on Scrum.org and take the assessment when you are ready.

Fees, rules and format can change. Always confirm the current details on the official Scrum.org PSM I page before you pay.

The PSM I Exam

Understand the question. Understand Scrum.

EXAM FORMAT



80

questions

Multiple choice,
multiple answer and true/false



60

minutes

No extra time



85%

passing score

Need 68 correct answers



Based on the official Scrum Guide

Realistic questions and exam simulation

WHAT THE EXAM COVERS



Understanding and Applying the Scrum Framework

Empiricism, Scrum Values, Scrum Team, Events, Artifacts, Done



Developing People and Teams

Self-Managing Teams, Facilitation, Coaching



Managing Products with Agility

Product Backlog Management, Product Value, Stakeholders, Forecasting



Better
Teams
Better
Results

- ✓ Scrum Framework
- ✓ People and Teams
- ✓ Agile Product
- ✓ Real Exam Practice

The exam format, in detail numbers you must know before you start

80

questions

single answer, multiple answer,
true/false

60

minutes

one sitting, no breaks scheduled

85%

to pass

68 correct, at most 12 mistakes

45 s

per question

so speed of reading matters

Three question types

One correct answer (radio buttons).

Multiple answers: "choose 2" or "choose 3"; all must be right to score.

True / false: one statement, read every word.

Time strategy

Answer every question on the first pass, even when unsure. Mark hard ones and come back. A blank answer is always wrong; a reasoned guess often is not. Keep about 10 minutes for review.

What is tested

Almost everything comes from the **2020 Scrum Guide**: theory, values, the Scrum Team, events, artifacts and commitments, applied to real situations. When your workplace and the Guide disagree, **the Guide wins**.

Cost and attempts

Each attempt needs its own password from Scrum.org. At the time of writing a password cost about **US \$200**. Check the current price before you buy.

After you pass

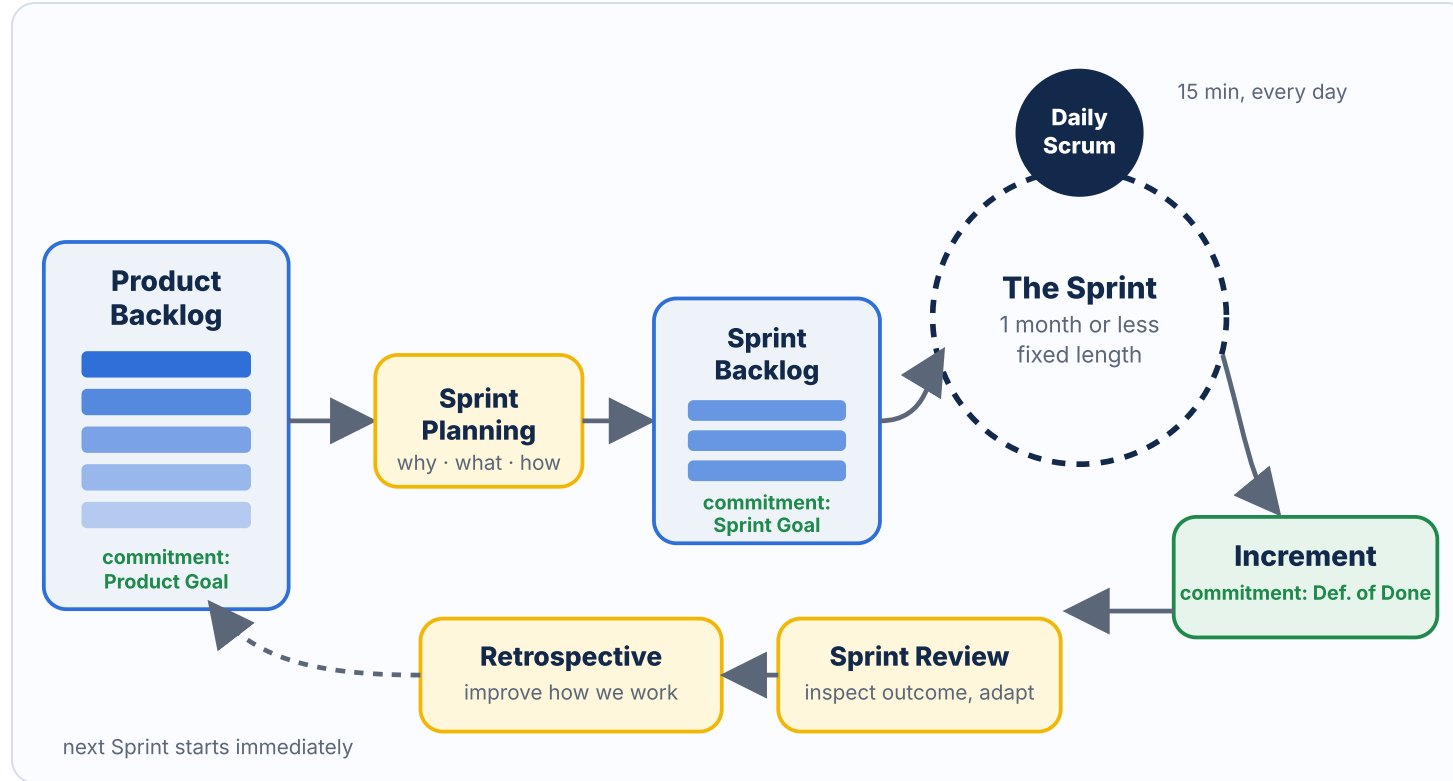
Scrum.org lists your name publicly and you can use the PSM I credential. It does not expire and has no annual fee.

Our readiness rule

Book the real exam only when you score **90% or more on two full timed practice exams in a row**. That margin protects you from exam-day nerves.

Scrum on one page

3 accountabilities · 5 events · 3 artifacts · 3 commitments · 5 values · 3 pillars



The Sprint is a container for all other events. Refinement is an ongoing activity, not an event. A release can happen at any time, not only at the Sprint Review.

The Scrum Team (10 or fewer people, typically)

Product Owner: maximizes the value of the product; orders the Product Backlog. One person, not a committee.

Scrum Master: establishes Scrum; accountable for the Scrum Team's effectiveness.

Developers: create a usable Increment every Sprint; own the Sprint plan.

3 pillars of empiricism

Transparency → Inspection → Adaptation

5 values

Commitment · Focus · Openness · Respect · Courage

Artifact → its commitment

Product Backlog → **Product Goal**

Sprint Backlog → **Sprint Goal**

Increment → **Definition of Done**

Events, timeboxes and who must be there

Maximum timebox for a one-month Sprint (shorter Sprints: usually shorter events)

Sprint Planning	8 h
Sprint Review	4 h
Retrospective	3 h
Daily Scrum	15 min, every day
Refinement	not an event · no timebox · ongoing activity

Exam trap

A timebox is a **maximum**, not a target. An event can end early when its purpose is achieved. The Sprint itself is the exception in one way: its length is **fixed** and it is never extended.

Event	Purpose	Who
The Sprint	Container for all events; turns ideas into value	Scrum Team
Sprint Planning	Why is this Sprint valuable? What can be Done? How will it get done?	Scrum Team; others may be invited to advise
Daily Scrum	Inspect progress toward the Sprint Goal; adapt the Sprint Backlog	Developers (PO and SM take part if they work on Sprint Backlog items)
Sprint Review	Inspect the outcome and decide future adaptations; a working session	Scrum Team and key stakeholders
Sprint Retrospective	Plan ways to increase quality and effectiveness	Scrum Team
Refinement (activity)	Break down and add detail to Product Backlog items	Developers and PO, as needed

Only the Product Owner can cancel a Sprint, and only if the Sprint Goal becomes obsolete.

2017 words vs 2020 words

old terms still appear in study material and in wrong answer options

Older guides (2017 and before)	2020 Scrum Guide (tested on PSM I)
Development Team (a team inside the team)	Developers; one Scrum Team, no sub-teams or hierarchies
Roles	Accountabilities
Self-organizing	Self-managing: the team decides who does what, when and how
No long-term goal for the backlog	Product Goal: the commitment for the Product Backlog
Commitments not named	Each artifact has a commitment: Product Goal, Sprint Goal, Definition of Done
Sprint Planning: what and how	A third topic first: why is this Sprint valuable?
Three Daily Scrum questions	Removed; Developers choose any structure that serves the purpose
Development Team of 3 to 9	Scrum Team typically 10 or fewer people
Refinement "usually no more than 10% of capacity"	No percentage; refinement is an ongoing activity
Scrum Master as "servant-leader"	Scrum Masters are "true leaders who serve"

How the exam uses this

A wrong option often sounds right because it was right in 2017: "the three questions of the Daily Scrum", "the Development Team decides", "refinement takes up to 10%". If an option uses an old term or an old rule, slow down and check it against 2020.

Lesson 25 goes deeper

Reading PSM I questions and the full 2017 vs 2020 comparison, with practice.

Go to lesson 25 →

Think like the Scrum Guide: 12 rules behind the right answers



1. The Guide wins

When your company and the Guide disagree, answer from the Guide.



2. Adapt as soon as possible

Do not wait for the next event when you learn something.



3. Make it visible first

Transparency before inspection; fix the real picture first.



4. Developers own the how

Who does what and how is the Developers' decision.



5. One Product Owner

One person orders the backlog; others influence, not decide.



6. The SM serves

Coach, teach, facilitate, remove impediments; never decide for the team.



7. Done means the DoD

Not Done = not in the Increment; it goes back to the Product Backlog.



8. Quality does not drop

Renegotiate scope with the PO, never the Definition of Done.



9. Fixed-length Sprints

A Sprint is never extended; only the PO can cancel it.



10. Accountability stays

Delegating work does not move accountability.



11. One product, one backlog

Many teams share one Product Goal, Product Backlog and PO.



12. Release any time

The Sprint Review is a working session, not a release gate.

Use the 12 rules as an elimination checklist: an option that breaks one of them is almost always wrong.

How to read a PSM I question: decode, then answer

EXAMPLE

On day 7 of a 10-day Sprint, the Developers learn that one selected item is much bigger than expected. The Sprint Goal can still be met without it. A manager suggests extending the Sprint by three days. **What should happen?**

- ☐ A Extend the Sprint by three days, this one time
- ☒ B The Developers renegotiate the scope with the Product Owner
- ☐ C Lower the Definition of Done for this item so it fits
- ☐ D The Scrum Master cancels the Sprint and starts a new one

Answer: B. Scope may be clarified and renegotiated with the Product Owner as more is learned. A: Sprints are never extended. C: quality does not decrease. D: only the PO can cancel, and only if the goal is obsolete.

- 1 Read the last sentence first**
What is asked? Best action, who, when, or which TWO statements?
- 2 Find the time and place**
Which event, which day? Is the Sprint Goal still possible?
- 3 Find who acts**
Developers, Product Owner, Scrum Master or someone outside the team?
- 4 Remove the noise**
Names, job titles and company details rarely change the answer.
- 5 Test each option with the 12 rules**
Cross out options that break a rule. Choose the one that follows the Guide exactly.

Your 4-week study plan

about 45 minutes a day, 6 days a week

Week 1

Lessons 0 to 10

Read the Scrum Guide once. Theory, values, the Scrum Team and its accountabilities.

Week 2

Lessons 11 to 18

Events, artifacts and commitments. Start 20 practice questions a day.

Week 3

Lessons 19 to 26

Value, quality, many teams, organization, exam technique. Try a lab.

Week 4

Mock exams

Full 80-question exams in 60 minutes. Review every mistake. Reread the Guide.

Every day

One lesson (5 pages) · its four questions · ten more practice questions · write down one thing you got wrong and why.

Ready to book?

90% or more on two full timed practice exams in a row, and you can explain every lesson's memory card without looking. [Practise on pmclear.com](#) →

Start

Know the exam, the problem Scrum solves and the exact words it uses.

Lesson 0 The PSM I exam and how to study

Lesson 1 Complex work and why Scrum exists

Lesson 2 The language of Scrum



LESSON 0 OF 26 · PART A · START

The PSM I exam and how to study

PSM I is a short, fast exam built on one document, the Scrum Guide. You pass it by **understanding the Guide deeply** and **reading each question for its exact meaning**, not by memorizing answers.

🕒 About 20 minutes, including the practice questions online

What you will learn

1. Describe the PSM I format: number of questions, time, pass mark and question types.
2. Explain why the 2020 Scrum Guide is the source of truth, even when your workplace does Scrum differently.
3. Use a time strategy that fits about 45 seconds per question.
4. Read answer options for meaning instead of reacting to single words.
5. Follow this course's path from lessons to mock exams to a readiness decision.

Why it matters on the exam

Nothing in this lesson is tested directly on PSM I. It is here so you study the right way. Exam details can change, so confirm the current facts on the official Scrum.org PSM I page before you buy.

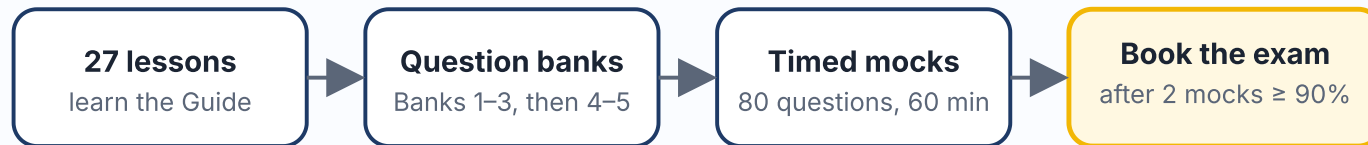
Simple explanation

PSM I means **Professional Scrum Master I**, the first-level Scrum Master certificate from Scrum.org. You take it online, from your own computer, in English. There are no prerequisites: no course, degree or work experience is required.

- **80 questions in 60 minutes**, so about 45 seconds per question.
- **85% to pass**: at least 68 correct answers, so no more than 12 mistakes.
- Question types: one correct answer, several correct answers ("choose 2" or "choose 3"), and true or false.
- The certificate does not expire, and there is no renewal fee.
- Each attempt is paid for separately. At the time of writing the fee was about US \$200; check the current price on Scrum.org.

Almost every question tests your understanding of **the Scrum Guide, 2020 version**. It is about 14 pages long. This course explains every part of it in simple English, then trains you on realistic situations.

See it in one picture Lesson 0 · The PSM I exam and how to study



Pass mark 85% (68 of 80) · about 45 seconds per question

The 90% target leaves a safety margin above the 85% pass mark.

Your memory card

1. PSM I: 80 questions, 60 minutes, 85% pass mark, in English, no prerequisites, no expiry.
2. The 2020 Scrum Guide is the only source of truth on the exam.
3. Read options for their meaning and strength, not for trigger words.
4. Answer every question; mark doubtful ones and return later.
5. Aim for 90% or more on two mocks in a row before booking.

Official source: Check current exam details on the official Scrum.org PSM I page. Read the Scrum Guide 2020 once before starting Lesson 1; it takes about 30 minutes.

[Open the full lesson →](#)

Deeper professional explanation

Lesson 0 · The PSM I exam and how to study

Why the Scrum Guide is the only source of truth

Many companies change Scrum: they add team leads, skip Retrospectives or use old terms. On the exam none of that matters. If your workplace and the Guide disagree, the Guide wins. People with years of "Scrum experience" often fail for exactly this reason.

How PSM I questions are built

Most questions describe a situation and ask for the *best* action, or ask which statements are accurate. Usually two options look reasonable. The difference is often one idea: **who** holds an accountability, **when** something happens, or whether the Guide **requires** something or merely **allows** it.

Read for meaning, not for single words

Some study guides teach that options with "always" or "never" are wrong. That is unreliable. The Guide itself uses strong words: Developers are *always* accountable for four things, and *only* the Product Owner can cancel a Sprint. Instead, ask: *is this statement true according to the Guide, in the exact strength it claims?*

Managing 45 seconds per question

Answer every question on the first pass, even when unsure. Mark difficult ones for review. With time left, return to the marked questions. An unanswered question is always wrong; a reasoned guess often is not.

How to use this course

Work through the lessons in order, answer every knowledge check, then use the question banks: Banks 1 to 3 first, then 4 and 5. Take full 80-question mock exams under time pressure. Book the real exam when you score 90% or more on at least two mocks in a row. That margin above 85% protects you from exam-day nerves.

Case study: Two candidates prepare Decide before you read the answer (cover the green part)

Decision 1: When to book

Two weeks of study

Priya works in a company that has used Scrum for three years. After two weeks of study she scores 78%, 81% and 80% on three timed mock exams. Her manager offers to pay for the exam if she books this week.

What is the wisest next step?

- ☐ A Book this week, since three years of Scrum work will carry her through
- ☒ B Study the topics she misses most, and book once mocks are consistently above 90%
- ☐ C Book now and treat the first attempt as a practice run
- ☐ D Stop doing mocks and reread the Guide once the night before

Best decision: B. Her scores sit below the 85% pass mark. Workplace experience can even hurt, because her company's habits may differ from the Guide. She reviews her weakest topics, reaches 92% and 94% on two mocks, books, and passes.

Why not the others? A trusts experience over evidence. C pays for an attempt she is likely to fail. D removes the timed practice that exposes weak spots.

Decision 2: The stubborn question

Exam day, minute 12

On question 18, Priya cannot decide between two options. She has spent 90 seconds on it. Sixty-two questions remain.

What should she do?

- ☐ A Keep thinking until she is certain, since accuracy matters most
- ☐ B Leave it blank and never come back to it
- ☒ C Choose the better-supported option, mark it for review, and move on
- ☐ D Pick the longest option, since it is usually the most complete

Best decision: C. She keeps her pace and returns at the end with fresh eyes.

Why not the others? A steals time from easier questions. B guarantees a wrong answer. D is a myth: length is not a reliable signal.

Common misunderstandings

- ✗ "I use Scrum at work, so I will pass easily."

✓ The exam tests the 2020 Scrum Guide, not your company's version. Workplace habits are a common cause of wrong answers.
- ✗ "Answers containing 'always' or 'never' are wrong."

✓ The Guide itself uses strong words. Judge each statement by whether it is true in the Guide, at the strength it claims.
- ✗ "I can look things up during the exam."

✓ With about 45 seconds per question there is no time. You must understand the Guide beforehand.
- ✗ "Older study material is fine."

✓ Material written before 2020 uses retired terms and rules, such as "Development Team" and "3 to 9 Developers".

Exam traps

The trap	How to read it
Answering from your workplace's practice.	Ask what the Scrum Guide says, even if your company does it differently.
Treating a strong word as automatically wrong.	Check the claim against the Guide. Some strong statements are exactly right.
Missing "choose 2" or "choose 3".	Read the end of the question first. A multi-answer question is wrong unless every selection is right.

Exam-style questions

12 questions in the online lesson

1. How many questions does PSM I contain, and what is the time limit?

A 180 questions in 230 minutes

B 60 questions in 80 minutes

C 100 questions in 90 minutes

D 80 questions in 60 minutes
2. A colleague says: "Our company calls the Developers 'the Development Team', so that's the right answer on the exam." What is the best response?

A Both terms are always marked correct on the exam

B The exam follows the 2020 Guide's term: Developers

C The exam avoids terminology questions altogether

D The company's term wins, since it reflects real practice
3. You have spent 90 seconds on one question and are still unsure. What is the best strategy?

A Leave it blank and continue

B Choose the longest option and move on

C Answer, mark it, and move on

D Keep working on it until you are sure
4. A Developer with five years of Scrum experience fails PSM I with 79%. Which explanations are most likely? (Choose 2)

A The exam tests software coding skills

B Her company's practices differ from the Scrum Guide

C She answered from habit rather than from the Guide

D The exam was made harder for experienced people

Answers

1: D PSM I has 80 questions with a 60-minute limit. · 2: B The 2020 Guide replaced "Development Team" with "Developers". The exam follows the Guide. · 3: C Keeping pace matters; return to marked questions at the end. · 4: B and C Experience can mislead when the workplace differs from the Guide.

All 12 questions with full explanations →

Free PSM I questions →

LESSON 1 OF 26 · PART A · START

Complex work and why Scrum exists

When work is **complex**, much of what matters is unknown at the start. Scrum handles this by working in **short cycles** that create something usable, so the team can **learn from real results** and adjust.

🕒 About 30 minutes, including the practice questions online

What you will learn

1. Distinguish work that can be planned in detail from work that is complex.
2. Explain when a detailed upfront plan works well, and when it fails.
3. Describe how short Sprints that produce usable results reduce risk.
4. State the Scrum Guide's definition of Scrum and what each key word means.
5. Recognize Scrum's use beyond software.

Why it matters on the exam

Questions rarely ask "what is complex work?" directly. They test whether you choose observation over prediction, and whether you recognize Scrum as a lightweight framework rather than a full method.

Simple explanation

Some work is **predictable**. A crew installs the same model of conveyor for the tenth time. The steps are known, the risks are known, and a detailed plan at the start works well.

Other work is **complex**. A company builds a product nobody has built before. Customers are unsure what they want, the technology may behave unexpectedly, and the market may change. A detailed one-year plan for this work will be wrong in ways nobody can see yet.

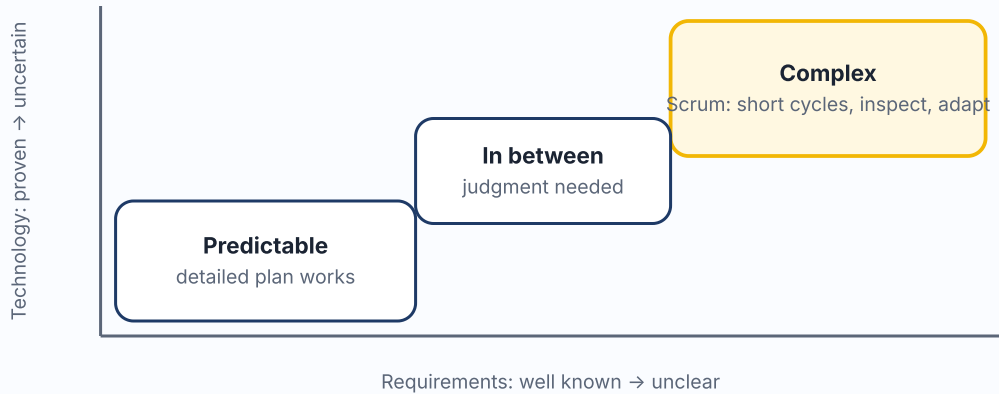
Scrum is designed for complex work. Instead of one long plan and one delivery at the end, the team works in **Sprints** of one month or less. Each Sprint produces a **usable** piece of the product. The team and its stakeholders inspect it, learn, and decide what to do next.

Mistakes surface in weeks, while they are still cheap to fix.

"Scrum is a lightweight framework that helps people, teams and organizations generate value through adaptive solutions for complex problems."

The Scrum Guide, 2020, Scrum Definition

See it in one picture
 Lesson 1 · Complex work and why Scrum exists

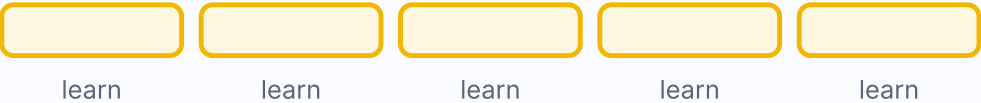


This chart is a common teaching model, not part of the Scrum Guide. It shows why one approach does not fit all work.

One long plan



Sprints



Each Sprint ends with something usable to inspect, so problems surface early instead of at the end.

Your memory card

- 1. Complex work has many unknowns; predictable work does not.
- 2. A detailed upfront plan fits predictable work, not complex work.
- 3. Scrum is a lightweight framework for adaptive solutions to complex problems.
- 4. Short Sprints produce usable results so problems surface early.
- 5. Scrum plans every Sprint; it does not abandon planning.
- 6. Scrum is used far beyond software.

Official source: Scrum Guide 2020: **Purpose of the Scrum Guide** and **Scrum Definition**. Read them at [scrumguides.org](https://www.scrumguides.org).

Open the full lesson →

Deeper professional explanation Lesson 1 · Complex work and why Scrum exists

Reading the definition word by word

Lightweight: few rules. **Framework:** a structure, not a complete method; teams add their own practices. **Generate value:** the aim is benefit, not activity. **Adaptive solutions:** the solution changes as the team learns. **Complex problems:** the setting where these choices make sense.

discover the film tears at speed, months before it would have appeared at the final test.

Predictive planning is not wrong; it fits different work

A detailed upfront plan is the right tool when requirements and technology are well understood. Many engineering and construction tasks are like this. The problem appears when the same tool is used for work full of unknowns: the plan looks precise, but the precision is false. Scrum does not reject planning. It plans every Sprint, keeps long-term direction in the Product Goal, and revises plans as evidence arrives.

Why short cycles reduce risk

Every Sprint ends with something usable that can be inspected. If an idea is wrong, the team loses at most one Sprint of effort, not a year. The Guide describes this as an iterative, incremental approach that optimizes predictability and controls risk.

Beyond software

Scrum started in software, but the Guide describes its use in hardware, embedded software, autonomous vehicles, schools, government, marketing and running organizations. Wherever the work is complex, the same reasoning applies.

Example: a new packaging machine A machine builder designs a packaging line for a product shape it has never handled. Instead of finishing all drawings first, the team builds and tests one station each Sprint with real packages. In Sprint 2 they

Case study: FreshCart's first attempt Decide before you read the answer (cover the green part)

Decision 1: Choosing the approach

Project start

FreshCart wants a grocery delivery app. Nobody on the team has built one, and the director admits he is not sure what customers will value most. A consultant proposes a 60-page specification and a 12-month plan, "to remove uncertainty upfront".

What is the best response from the team?

- ☐ A Accept the plan, since detailed specifications reduce uncertainty
- ☒ B Work in short cycles that each produce something usable to test with customers
- ☐ C Skip planning entirely and start coding whatever seems useful
- ☐ D Write the specification, but add 30% contingency to every estimate

Best decision: B. The team releases a basic ordering feature to 50 customers after four weeks. Customers ignore the recipe section the director expected to be popular, and ask for delivery-time tracking instead. The plan changes early and cheaply.

Why not the others? A treats unknowns as if they were known. C confuses Scrum with no planning; Scrum plans every Sprint. D adds buffer to a guess, but the guess is still a guess.

Decision 2: A predictable task

Month 3

FreshCart must also install identical refrigerated lockers at 20 pickup points. The supplier, the design and the installation steps are all proven.

How should this work be planned?

- ☒ A A detailed plan is reasonable, since the work is well understood
- ☐ B It must be done in Sprints, or it is not allowed in the company
- ☐ C It should not be planned at all
- ☐ D The Scrum Master should take over the installation

Best decision: A. Known, repeatable work suits a detailed plan. Using the right tool for each kind of work is part of professional judgment.

Why not the others? B turns Scrum into dogma. C ignores useful knowledge. D confuses accountabilities.

Common misunderstandings

- ✗ "Scrum means no planning."

✓ Scrum plans every Sprint and keeps long-term direction in the Product Goal. It avoids pretending to know everything upfront.
- ✗ "Predictive planning is always bad."

✓ It fits well-understood work. Scrum is designed for complex work.
- ✗ "Scrum is a complete methodology."

✓ It is a lightweight framework. Teams add their own practices inside it.
- ✗ "Scrum is only for software."

✓ The Guide describes Scrum in hardware, government, schools, marketing and more.

Exam traps

The trap	How to read it
An option that adds buffer or detail to a long-range prediction.	More detail on a guess is still a guess. Look for the option that decides from observed results.
An option calling Scrum a "methodology" with steps for everything.	Scrum is a lightweight framework that is purposefully incomplete.
An option claiming Scrum avoids planning.	Scrum plans continually; plans are revised as evidence arrives.

Exam-style questions

15 questions in the online lesson

1. According to the Scrum Guide, Scrum helps generate value through what?

A

 Adaptive solutions for complex problems

B

 Detailed plans that remove uncertainty upfront

C

 Fixed scope delivered on a fixed date

D

 Standard processes for every team
2. Which work is the best fit for a detailed upfront plan?

A

 Building a product that nobody has built before

B

 Designing an app whose users are unclear

C

 Researching a new material's properties

D

 Installing a proven machine for the tenth time
3. A director wants a 12-month fixed plan for a brand-new product with unclear customer needs. What is the main risk?

A

 Stakeholders will see results too early

B

 The plan will be too short

C

 Stakeholders will see results far too late

D

 The team will have too little to do
4. A team adds a 30% buffer to every estimate in a detailed one-year plan for a new product. Which statements are most accurate? (Choose 2)

A

 The buffer does not remove the underlying unknowns

B

 A larger buffer would solve the problem fully

C

 The buffer makes the plan empirical

D

 Frequent inspection of real results would reduce risk more

Answers

1: A This is the Guide's definition. · 2: D Known, repeatable work suits detailed planning. · 3: C Unknowns show up only at the end of a long plan. · 4: A and D A buffer is added to a guess; evidence reduces risk.

All 15 questions with full explanations →

Free PSM I questions →

LESSON 2 OF 26 · PART A · START

The language of Scrum

Scrum words have **exact meanings**. Many exam questions turn on one of them: **accountable** versus responsible, **Developers** versus programmers, a **timebox** as a maximum, and 2020 terms versus old ones.

🕒 About 30 minutes, including the practice questions online

What you will learn

1. Use the 2020 Scrum Guide terms precisely.
2. Distinguish accountable from responsible.
3. Explain timebox, Increment, artifact, commitment and self-managing.
4. Recognize retired 2017 terms and why answers using them need care.
5. Judge statements by meaning and strength, not by single words.

Why it matters on the exam

Vocabulary is rarely tested as "define this word". Instead, a single term inside a scenario decides the answer: who is accountable, whether something is a rule, or whether a term is from an older Guide.

Simple explanation

This lesson is your dictionary. Read it now and return to it whenever a later lesson uses a word you are unsure about.

Accountable

You answer for the result, even if others do some of the work.

Developers

Everyone in the Scrum Team who creates the product: programmers, designers, testers, engineers, analysts.

Self-managing

The team decides internally who does what, when and how.

Cross-functional

The team as a whole has all the skills needed to create value each Sprint.

Timebox

A maximum length of time. An event may end earlier, never later.

Artifact

Something that represents work or value: the Product Backlog, the Sprint Backlog, the Increment.

Commitment

A goal or standard attached to each artifact: the Product Goal, the Sprint Goal, the Definition of Done.

Increment

A usable step toward the Product Goal that meets the Definition of Done.

Definition of Done

The formal description of the quality the Increment must have.

Stakeholder

Anyone outside the Scrum Team with an interest in the product.

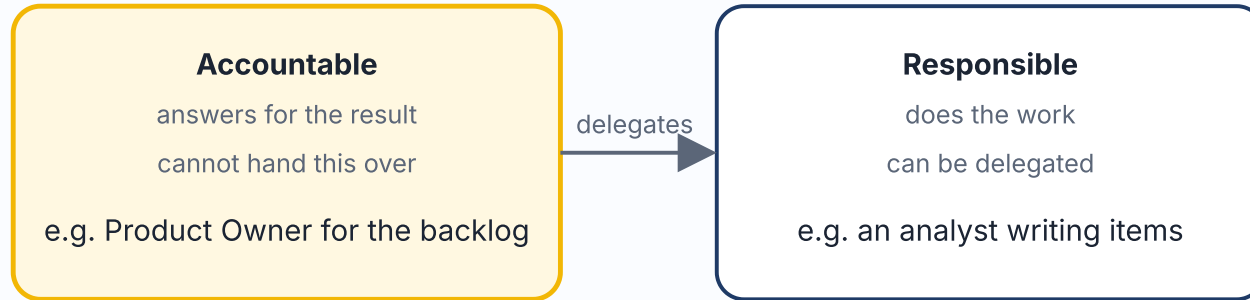
Empiricism

Deciding based on what is observed.

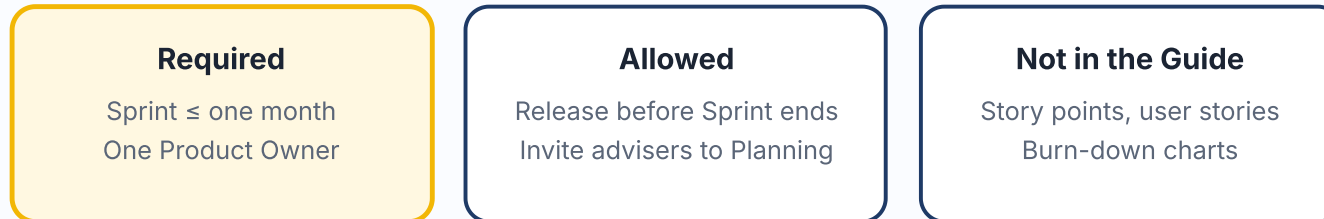
"The Product Owner may do the above work or may delegate the responsibility to others. Regardless, the Product Owner remains accountable."

The Scrum Guide, 2020, Product Owner

See it in one picture Lesson 2 · The language of Scrum



Work can move to others. Accountability stays where the Guide puts it.



Many wrong answers move an item from one box to another.

Your memory card

1. Accountable means answering for the result; delegating work does not move it.
2. Developers are everyone creating the Increment.
3. A timebox is a maximum.
4. Know the 2020 terms and the retired 2017 rules.
5. Distinguish required, allowed and not mentioned.
6. Judge statements by meaning and strength, not single words.

Official source: Scrum Guide 2020: **Scrum Team**, **Product Owner** and **Scrum Artifacts** at scrumguides.org.

[Open the full lesson →](#)

Deeper professional explanation

Lesson 2 · The language of Scrum

Accountable versus responsible

The quote above is the clearest example. The Product Owner may ask others to write Product Backlog items; those people are then responsible for doing that work. The Product Owner is still the one who answers for the result. When a question asks who is *accountable*, look for the person or group the Guide names, not whoever happens to do the task.

Terms retired in 2020

The 2020 Guide changed several words. Old study material still uses them, and some answers do too.

Older term	2020 term
Development Team	Developers (inside one Scrum Team)
Roles	Accountabilities
Self-organizing	Self-managing
3 to 9 Developers	Typically 10 or fewer people
Refinement about 10% of capacity	Refinement is an ongoing activity; no percentage

An answer using an older word is not automatically wrong. But if an answer depends on an old *rule*, such as the 3-to-9 range, it no longer matches the Guide.

Required, allowed, or not mentioned

Many questions turn on the strength of a claim. The Guide **requires** some things (the Sprint is one month or less). It **allows** others (an Increment may be delivered before the Sprint ends). And it is **silent** on many practices (story points, user

stories). A common trap is an option that turns "allowed" into "required", or invents a rule the Guide never states.

Strong words are not automatically wrong

The Guide says Developers are *always* accountable for four things, and *only* the Product Owner may cancel a Sprint. Judge each statement by whether it is true in the Guide at the strength it claims.

Case study: Words that change decisions Decide before you read the answer (cover the green part)

Decision 1: Who answers for it?

Sprint 2

FreshCart's Product Owner, Maya, asks a business analyst to write most Product Backlog items. Two months later, stakeholders complain that the backlog is confusing. The director asks: "Who is accountable for this?"

What is the accurate answer?

- ☒ **A** Maya, the Product Owner
- ☐ **B** The business analyst who wrote the items
- ☐ **C** The Scrum Master, who should have noticed
- ☐ **D** Maya and the analyst equally

Best answer: A. Maya delegated the work, but she remains accountable. She and the analyst agree a clearer format, and Maya reviews the backlog weekly.

Why not the others? B confuses doing the work with answering for it. C assigns an accountability the Guide does not give the Scrum Master. D splits an accountability the Guide keeps with one person.

Decision 2: The 15-minute rule

Sprint 3

The Daily Scrum finishes in 8 minutes. A new manager says: "The Daily Scrum is 15 minutes. You must use all of it."

What is the best response?

- ☐ **A** Agree, and fill the remaining time with updates
- ☒ **B** Explain that 15 minutes is a maximum, not a target
- ☐ **C** Extend future Daily Scrums to 20 minutes for safety
- ☐ **D** Ask the Scrum Master to decide each day

Best answer: B. A timebox is a maximum. Ending early when the purpose is met is good practice.

Why not the others? A wastes time. C breaks the timebox. D removes a decision the Guide already settles.

Common misunderstandings

- ✗ "Developers are programmers."

✓ Developers are everyone creating the Increment, whatever their skill.
- ✗ "A timebox is how long an event must last."

✓ It is a maximum; events may end earlier.
- ✗ "Whoever does the work is accountable for it."

✓ Accountability stays where the Guide places it, even when work is delegated.
- ✗ "Answers using the word 'never' are wrong."

✓ Some strong statements are exactly what the Guide says. Judge the meaning.

Exam traps

The trap	How to read it
"The Development Team is accountable for..."	An older term. Check whether the claim still matches the 2020 Guide.
An option turning "may" into "must".	Check whether the Guide requires it or only allows it.
An option inventing a rule, such as "Scrum forbids part-time members".	If the Guide does not state it, it is not a Scrum rule.

Exam-style questions 15 questions in the online lesson

1. What is a timebox?

A A tool for tracking Developer hours

B The minimum time an event must take

C The exact time an event must take

D The maximum time an event may take
2. A Product Owner asks a colleague to write most backlog items. Who is accountable for the Product Backlog?

A The whole Scrum Team equally

B The colleague who writes the items

C The Scrum Master

D The Product Owner
3. A UX designer on a Scrum Team asks whether she is a Developer. What is accurate?

A No, because Developers are the people who write code

B No, designers are stakeholders

C Yes, if she helps create the Increment

D Only if she also tests the product
4. Which statements are accurate? (Choose 2)

A Accountability is shared equally by all

B Delegating work does not move accountability

C Accountability belongs to whoever does the work

D The Guide names who is accountable for what

Answers

1: D A timebox is a maximum. · 2: D The PO remains accountable when delegating. · 3: C Developers are those creating any aspect of the Increment. · 4: B and D Accountabilities are assigned by the Guide and stay put.

All 15 questions with full explanations →

Free PSM I questions →

Foundations

The whole framework, the theory under it and the values that make it work.

Lesson 3 The framework in one picture

Lesson 4 Empiricism, lean and the three pillars

Lesson 5 Scrum values in real decisions



LESSON 3 OF 26 · PART B · FOUNDATIONS

The framework in one picture

Scrum has a small set of parts: **3 accountabilities, 5 events, 3 artifacts and 3 commitments**. Each part has a purpose. Teams may **add** practices, but if they **remove or change** parts, the result is not Scrum.

⌚ About 30 minutes, including the practice questions online

What you will learn

1. Name the 3 accountabilities, 5 events, 3 artifacts and 3 commitments.
2. Explain how the parts connect in one Sprint cycle.
3. Explain what "purposefully incomplete" allows and what it does not.
4. Tell an optional practice from a Scrum element.
5. Explain why leaving out parts covers up problems.

Why it matters on the exam

Exam questions often present a modified Scrum, such as a missing event, an extra role or a renamed accountability, and ask whether it is still Scrum or what the risk is.

Simple explanation

Scrum has four groups of parts:

- **3 accountabilities:** Product Owner, Scrum Master, Developers.
- **5 events:** the Sprint, which contains Sprint Planning, the Daily Scrum, the Sprint Review and the Sprint Retrospective.
- **3 artifacts:** Product Backlog, Sprint Backlog, Increment.
- **3 commitments, one per artifact:** Product Goal (Product Backlog), Sprint Goal (Sprint Backlog), Definition of Done (Increment).

The Guide summarizes Scrum in four steps. A Scrum Master fosters an environment where: a Product Owner orders the work for a complex problem into a Product Backlog; the Scrum Team turns a selection of the work into an Increment of value during a Sprint; the Scrum Team and its stakeholders inspect the results and adjust for the next Sprint; and this repeats.

"The Scrum framework is purposefully incomplete, only defining the parts required to implement Scrum theory."

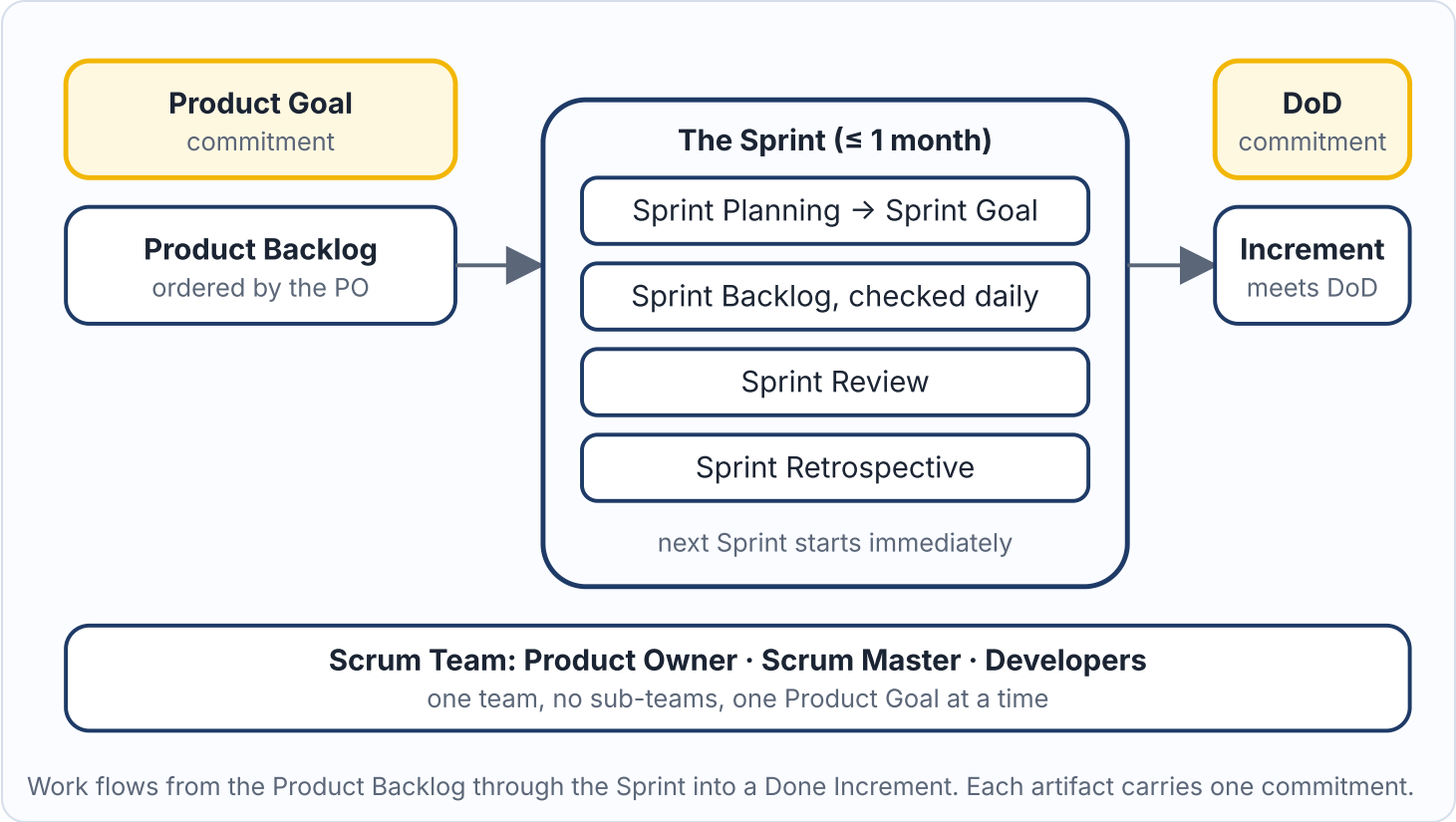
The Scrum Guide, 2020, Scrum Definition

"The Scrum framework, as outlined herein, is immutable. While implementing only parts of Scrum is possible, the result is not Scrum."

The Scrum Guide, 2020, End Note

See it in one picture

Lesson 3 · The framework in one picture



Your memory card

1. 3 accountabilities, 5 events, 3 artifacts, 3 commitments.
2. The Sprint contains the other four events.
3. Each artifact has one commitment.
4. Scrum is purposefully incomplete: add practices, don't remove parts.
5. Leaving out elements covers up problems and limits benefits.
6. Scrum makes the efficacy of current practices visible.

Official source: Scrum Guide 2020: **Scrum Definition**, **Scrum Events**, **Scrum Artifacts** and **End Note** at scrumguides.org.

Open the full lesson →

Deeper professional explanation Lesson 3 · The framework in one picture

Incomplete, but not editable

"Purposefully incomplete" means Scrum leaves the *how* open: estimation methods, board design, engineering practices. It does not mean its defined parts are optional. A team may add user stories, a Kanban board or automated tests. It may not drop the Retrospective, split the Product Owner into a committee, or add a team lead and still call it Scrum.

Why each part matters

The Guide says each element serves a specific purpose essential to the overall value. Remove the Retrospective and the team stops improving its own way of working. Remove the Sprint Review and stakeholders stop shaping the product. Remove the Definition of Done and nobody knows what "finished" means. The Guide warns that changing the core design or leaving out elements covers up problems and limits the benefits of Scrum.

Events replace other meetings

Each event is a formal opportunity to inspect and adapt. The Guide also says events create regularity and minimize the need for meetings not defined in Scrum. A team with many extra recurring meetings should first ask whether its Scrum events are doing their job.

Scrum wraps around existing practices

Scrum can host a team's existing techniques, or make some of them unnecessary. It also makes visible how effective current management, environment and work techniques are, so they can be improved. Sometimes what it reveals is uncomfortable, and that is intended.

Example: a hospital laboratory A lab team redesigning its sample-tracking process keeps its existing quality checklists inside Scrum. Within two Sprints, the Sprint Review reveals that one approval form adds two days and catches no errors. The team removes it with the lab director's agreement. Scrum did not ban the form; it made its cost visible.

Case study: The "lighter" Scrum Decide before you read the answer (cover the green part)

Decision 1: Dropping an event

Sprint 5

FreshCart's Developers are under deadline pressure. They propose skipping Sprint Retrospectives "until the release is out", arguing that Scrum is purposefully incomplete, so teams can adapt it.

How should the Scrum Master respond?

- ☐ A Agree, since incompleteness lets teams drop events under pressure
- ☒ B Explain that dropping an event is not Scrum, and help make the Retrospective short and useful
- ☐ C Hold the Retrospective alone and send the team a summary
- ☐ D Move the Retrospective to once per quarter as a compromise

Best decision: B. The team keeps a focused 45-minute Retrospective. It finds that half its overtime comes from unclear items reaching Sprint Planning, and fixes refinement. Pressure drops the following Sprint.

Why not the others? A misreads "incomplete". C removes the team from its own improvement. D weakens an event the Guide places at the end of every Sprint.

Decision 2: Adding parts

Sprint 7

A new manager wants two changes: every item written as a user story with story points, and a "deputy Product Owner" who can reorder the Product Backlog when Maya is busy.

Which assessment is most accurate?

- ☐ A Both changes are fine, since teams can add anything
- ☒ B User stories can be added as a practice; a second person ordering the backlog conflicts with Scrum
- ☐ C Both changes break Scrum, since the Guide forbids user stories
- ☐ D Neither change matters, since only events define Scrum

Best decision: B. The team adopts user stories as a practice. For Maya's busy weeks, she delegates writing items to the analyst but keeps ordering decisions, as the Guide allows.

Why not the others? A ignores that the Product Owner is one person. C invents a rule. D ignores accountabilities.

Common misunderstandings

- ✗ "Scrum has four events."

✓ Five: the Sprint itself is an event that contains the other four.
- ✗ "Purposefully incomplete means teams can drop parts."

✓ It means the how is open. Removing or changing parts means the result is not Scrum.
- ✗ "User stories and story points are part of Scrum."

✓ They are optional practices, not in the Guide.
- ✗ "Product Backlog refinement is an event."

✓ It is an ongoing activity.

Exam traps

The trap	How to read it
"Scrum Lite" or any variant with a missing part.	Implementing only parts of Scrum is possible, but the result is not Scrum.
A commitment matched to the wrong artifact.	Product Goal → Product Backlog; Sprint Goal → Sprint Backlog; DoD → Increment.
"Scrum has three roles."	The 2020 Guide calls them accountabilities.

Exam-style questions 15 questions in the online lesson

1. How many events does Scrum have?

A Four

B Six

C Three

D Five
2. A team uses a Kanban board and automated tests inside Scrum. What is accurate?

A These are practices a team may add

B These are forbidden in Scrum

C These replace the Sprint Backlog

D These must be approved by Scrum.org
3. A company keeps all events but has two people sharing the Product Owner role. Is it Scrum?

A Yes, if they agree on the order

B No; the PO is one person

C Yes, since all events happen

D Only if the Scrum Master approves
4. Which statements about "purposefully incomplete" are accurate? (Choose 2)

A Any team's variant still counts as Scrum

B Teams may remove events they find unhelpful

C Scrum defines only the parts needed for its theory

D Teams add their own practices within it

Answers

1: D The Sprint plus the four events inside it. · 2: A Various techniques can be used within the framework. · 3: B Changing an accountability means the result is not Scrum. · 4: C and D Scrum Definition; End Note.

All 15 questions with full explanations →

Free PSM I questions →

LESSON 4 OF 26 · PART B · FOUNDATIONS

Empiricism, lean and the three pillars

Scrum is founded on **empiricism** (decide from what you observe) and **lean thinking** (cut waste). Empiricism needs three pillars: **transparency**, **inspection** and **adaptation**. If one fails, the others lose their value.

🕒 About 30 minutes, including the practice questions online

What you will learn

1. Define empiricism and lean thinking as the Guide does.
2. Explain how transparency enables inspection, and inspection enables adaptation.
3. Diagnose which pillar fails in a realistic situation.
4. Explain why adaptation must happen as soon as possible.
5. Connect the pillars to Scrum's events and artifacts.

Why it matters on the exam

This is one of the most tested ideas. Scenarios describe a broken team; you must name the failing pillar or choose the action that restores it.

Simple explanation

Empiricism means knowledge comes from experience, and decisions are based on what is observed. **Lean thinking** means reducing waste and focusing on what matters.

Empiricism stands on three pillars:

- **Transparency.** The work and the process are visible to those doing the work and those receiving it.
- **Inspection.** The artifacts and progress toward goals are checked frequently and diligently, to find problems early.
- **Adaptation.** When something is off, the team adjusts as soon as possible.

They work as a chain. You cannot inspect well what you cannot see, and inspecting without acting changes nothing.

"Scrum is founded on empiricism and lean thinking. Empiricism asserts that knowledge comes from experience and making decisions based on what is observed. Lean thinking reduces waste and focuses on the essentials."

The Scrum Guide, 2020, Scrum Theory

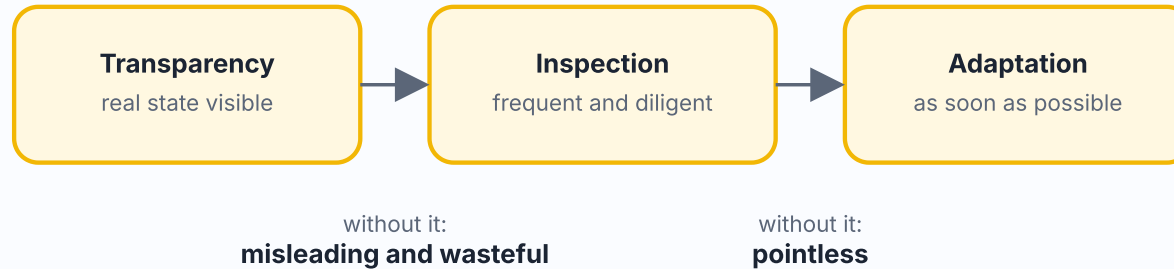
"Transparency enables inspection. Inspection without transparency is misleading and wasteful."

The Scrum Guide, 2020, Transparency

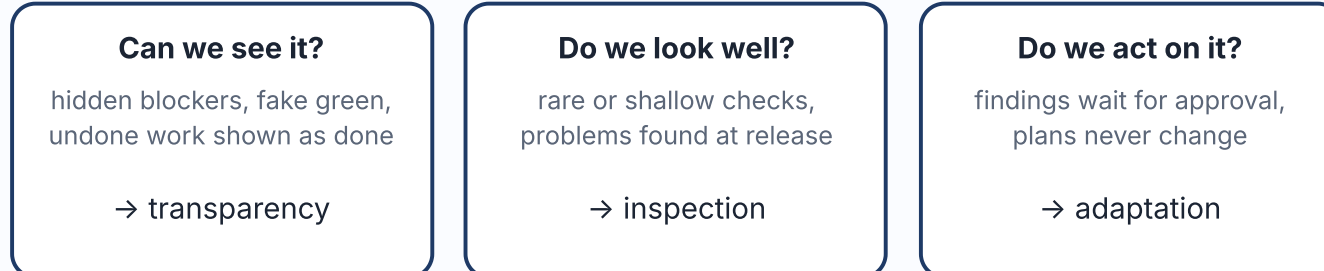
"Inspection enables adaptation. Inspection without adaptation is considered pointless."

The Scrum Guide, 2020, Inspection

See it in one picture Lesson 4 · Empiricism, lean and the three pillars



Each pillar enables the next. The Guide's own words describe what happens when a link breaks.



Ask the three questions in order to find the weakest pillar.

Your memory card

1. Scrum rests on empiricism and lean thinking.
2. Transparency enables inspection; inspection enables adaptation.
3. Inspection without transparency is misleading and wasteful.
4. Inspection without adaptation is pointless.
5. Adapt as soon as possible; empowerment makes this possible.
6. Forecasting tools help but never replace empiricism.

Official source: Scrum Guide 2020: **Scrum Theory** (Transparency, Inspection, Adaptation) at scrumguides.org.

[Open the full lesson →](#)

Deeper professional explanation Lesson 4 · Empiricism, lean and the three pillars

Diagnosing a failing pillar

Ask three questions in order. *Is the real state visible?* If a board says "almost done" while work is blocked, transparency fails. *Is anyone looking, carefully and often?* If problems surface only at release, inspection fails. *Does anything change after looking?* If findings wait for a quarterly committee, adaptation fails.

Adapt as soon as possible

The Guide says that if any aspect deviates outside acceptable limits, the process or the materials being produced must be adjusted as soon as possible to minimize further deviation. It also says a Scrum Team is expected to adapt the moment it learns anything new. Events are formal opportunities, not the only ones.

Empowerment makes adaptation possible

The Guide notes that adaptation becomes more difficult when the people involved are not empowered or self-managing. A team that sees a problem but needs three signatures to act will adapt slowly, however good its inspection.

Forecasting and empiricism

Practices such as burn-down charts can be useful, but they do not replace empiricism. In complex work, only what has already happened may be used for forward-looking decisions. A chart that shows "on track" while the team knows otherwise is a transparency failure, not a forecasting success.

Lean thinking in practice

Lean thinking asks what is essential. Detailed specifications for features two years away, reports nobody reads, and work waiting between handoffs are all waste. Removing them frees time for inspection and adaptation.

Case study: The green dashboard Decide before you read the answer (cover the green part)

Decision 1: The hidden blocker

Sprint 4, day 6

FreshCart's dashboard shows every item green. In private, Leo admits the payment integration has failed for three days. He hoped to fix it quietly before anyone noticed. Mr. Grant is about to promise investors a launch date based on the dashboard.

What should happen first?

- ☐ A Fix it quietly; the dashboard will be right by Friday
- ☒ B Make the real status visible now, so decisions use true information
- ☐ C Change the dashboard colour rules so it shows amber
- ☐ D Wait for the Sprint Review to raise it formally

Best decision: B. Leo updates the board and tells Maya. She asks Mr. Grant to wait two days before promising a date. The team swaps in a simpler payment provider within the Sprint.

Why not the others? A lets a false picture drive an investor promise. C changes the display, not the truth. D delays adaptation by four days.

Decision 2: The approval board

Sprint 8

Every Sprint Review produces good ideas. But any change to the Product Backlog order must be approved by a monthly steering board. By the time it approves, the information is stale. Developers say Reviews feel pointless.

Which pillar is failing, and what should the Scrum Master do?

- ☐ A Inspection; hold Reviews more often
- ☒ B Adaptation; help the organization let the PO adapt quickly
- ☐ C Transparency; add more detail to the backlog
- ☐ D None; the process works as designed

Best decision: B. The Scrum Master shows the steering board how long changes wait and what that costs. The board agrees to review strategy quarterly but leaves ordering to Maya. Reviews start changing the plan again.

Why not the others? A inspects more without acting. C improves visibility, but the block is in acting. D ignores pointless inspection.

Common misunderstandings

✗ **"The pillars are commitment, focus and courage."**

✓ Those are values. The pillars are transparency, inspection and adaptation.

✗ **"Adaptation happens at the Sprint Review."**

✓ Adapt as soon as possible, whenever something is learned.

✗ **"More data means more empiricism."**

✓ Empiricism is deciding from observation. Data nobody acts on changes nothing.

✗ **"A green dashboard means good transparency."**

✓ Transparency is about the real state, not a display.

Exam traps

The trap	How to read it
"Inspection without transparency is pointless."	The Guide calls it misleading and wasteful. "Pointless" is inspection without adaptation.
An option that waits for the next event to adapt.	The team adapts the moment it learns something new.
An option that adds more inspection when the block is in acting.	Find the pillar that fails; more checking does not fix blocked adaptation.

Exam-style questions

15 questions in the online lesson

1. What are the three pillars of empiricism?
- A

Commitment, focus, courage

B

Openness, respect, transparency

C

Planning, execution, monitoring and control

D

Transparency, inspection, adaptation
2. A board shows work "almost done" that has been blocked for days. Which pillar fails?
- A

Transparency

B

Adaptation

C

Inspection

D

Commitment
3. Problems are always found only during final testing before release. Which pillar is weakest?
- A

Adaptation

B

Inspection

C

Transparency

D

Lean thinking
4. Which conditions make adaptation harder, according to the Guide? **(Choose 2)**
- A

Stakeholders attend Reviews

B

Sprints are short

C

People are not empowered

D

People are not self-managing

Answers 1: **D** See, check, change. · 2: **A** The real state is not visible. · 3: **B** Inspection should be frequent and diligent. · 4: **C and D** Empowerment enables change.

All 15 questions with full explanations →

Free PSM I questions →

LESSON 5 OF 26 · PART B · FOUNDATIONS

Scrum values in real decisions

Five values guide how people behave in Scrum: **commitment, focus, openness, respect and courage**. When the Scrum Team and the people it works with live them, the pillars come to life and **trust** grows.

⌚ About 25 minutes, including the practice questions online

What you will learn

1. Name the five Scrum values and what each means in practice.
2. Identify which value is at stake in a realistic dilemma.
3. Explain how the values support the three pillars.
4. Recognize systems, such as reward schemes, that undermine the values.
5. Avoid confusing values with pillars.

Why it matters on the exam

Questions rarely ask you to list the values. They describe a behaviour or decision and ask which value it shows or harms, or what action best reflects the values.

Simple explanation

- **Commitment:** the Scrum Team commits to achieving its goals and to supporting each other.
- **Focus:** the main focus is the work of the Sprint, to make the best possible progress toward the goals.
- **Openness:** the Scrum Team and its stakeholders are open about the work and the challenges.
- **Respect:** team members respect each other as capable, independent people, and are respected that way by the people they work with.
- **Courage:** team members have the courage to do the right thing and to work on tough problems.

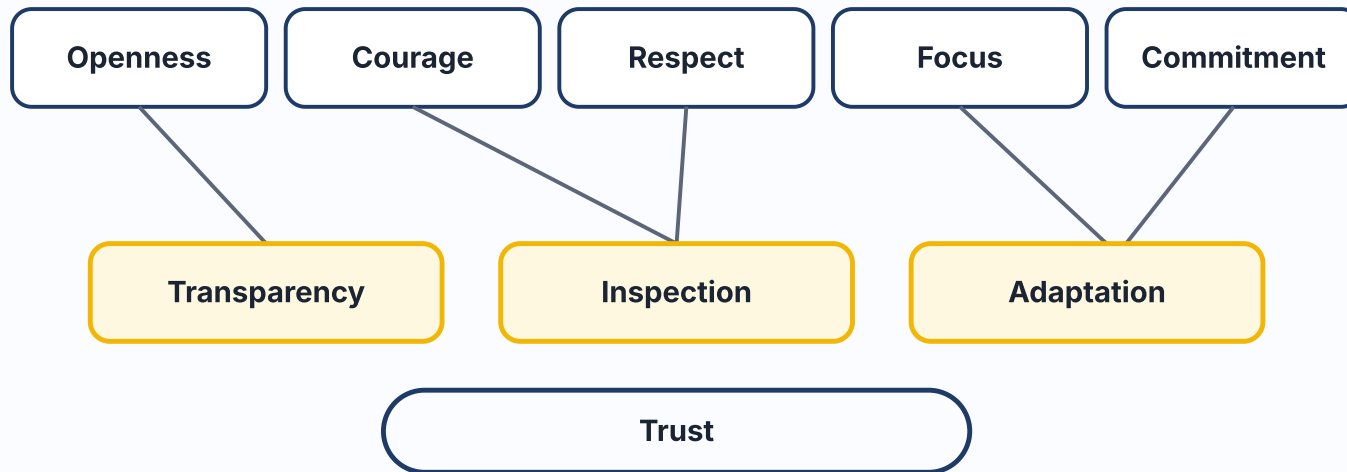
"Successful use of Scrum depends on people becoming more proficient in living five values: Commitment, Focus, Openness, Respect, and Courage."

The Scrum Guide, 2020, Scrum Values

"When these values are embodied by the Scrum Team and the people they work with, the empirical Scrum pillars of transparency, inspection, and adaptation come to life building trust."

The Scrum Guide, 2020, Scrum Values

See it in one picture Lesson 5 · Scrum values in real decisions



The lines show typical links, not a rule from the Guide. The Guide states only that living the values brings the pillars to life and builds trust.

Your memory card

1. Five values: commitment, focus, openness, respect, courage.
2. Living them brings the pillars to life and builds trust.
3. Commitment is to goals and each other, not to every item at any cost.
4. Values apply to the Scrum Team and the people it works with.
5. Systems and rewards should reinforce, not undermine, the values.
6. Values are learned by working with Scrum's events and artifacts.

Official source: Scrum Guide 2020: **Scrum Values** at scrumguides.org.

[Open the full lesson →](#)

Deeper professional explanation Lesson 5 · Scrum values in real decisions

Values make the pillars work

Transparency needs **openness**: people must be willing to show bad news. Honest inspection needs **courage**: saying "this is not Done" to a director takes nerve. Adaptation needs **commitment** and **focus**: the team changes course for the goal, not for comfort. **Respect** holds it together: people speak up when they expect to be taken seriously.

Commitment is not a guarantee of every item

Commitment means dedication to the goals and to each other. It does not mean finishing every selected item whatever happens, or working endless overtime. A team that hides strain to "keep its commitment" is failing openness, not living commitment.

Values apply beyond the team

The Guide speaks of the Scrum Team *and the people they work with*. Stakeholders who hide a budget cut, or managers who treat Developers as interchangeable resources, undermine the values too.

Systems can undermine values

The Guide says decisions, steps taken and the way Scrum is used should reinforce the values, not diminish or undermine them. A bonus scheme that rewards individual task counts quietly discourages helping each other. A Scrum Master should make such effects visible.

Values are learned through practice

The Guide says team members learn and explore the values as they work with the Scrum events and artifacts. Nobody masters them before starting.

Example: a construction design team An architect finds a structural clash that will delay drawings by a week. Telling the client before the review takes courage and openness. Because she does, the client adjusts the permit schedule in time. Hiding it would have cost a month.

Case study: Pressure before the investor demo Decide before you read the answer (cover the green part)

Decision 1: "Tell me it's finished"

Day before the Sprint Review

Mr. Grant will show the search feature to investors tomorrow. It works in the office, but it has not passed performance testing, which the Definition of Done requires. He asks Tom: "Just tell me it's finished."

What response best reflects the Scrum values?

- ☐ A Say it is finished, to protect the team's reputation
- ☒ B Explain openly that it is not Done, and offer to show it as clearly unfinished work outside the Increment
- ☐ C Ask the Scrum Master to answer instead
- ☐ D Quietly skip the performance test so it can be called Done

Best decision: B. Tom shows courage and openness. Mr. Grant tells investors the feature is in final testing. A week later it ships properly, and his trust in the team's reports grows.

Why not the others? A is dishonest and harms trust. C avoids the hard conversation. D lowers quality to fake a result.

Decision 2: The helpful interruption

Sprint 6, day 3

The sales director asks Priya to spend three days building a demo for a prospect. It is unrelated to the Sprint Goal. Priya wants to help, but the team needs her on the goal.

What is the best way forward?

- ☐ A Priya builds the demo quietly in the evenings
- ☐ B Priya refuses and tells sales never to ask again
- ☒ C Priya brings the request to the Product Owner, so it is weighed openly against the Sprint Goal
- ☐ D Priya asks the Scrum Master to decide for her

Best decision: C. Maya, the Product Owner, sees the prospect is valuable. She adds a smaller demo item to the Product Backlog for the next Sprint, and the current Sprint Goal stays protected.

Why not the others? A hides work and harms sustainable pace. B lacks respect for a real business need. D hands away a conversation she can lead herself.

Common misunderstandings

✗ "Transparency is a Scrum value."

✓ It is a pillar. The closest value is openness.

✗ "Commitment means finishing every selected item."

✓ It means commitment to the goals and to each other.

✗ "Values are only for the Scrum Team."

✓ The Guide includes the people the team works with.

✗ "Focus means ignoring everyone outside the team."

✓ Focus is on the best progress toward goals, which can include collaboration.

Exam traps

The trap	How to read it
A list mixing values and pillars.	Values: commitment, focus, openness, respect, courage. Pillars: transparency, inspection, adaptation.
"Committed, so we work overtime until every item is done."	Commitment is to goals and each other; sustainable pace still matters.
A behaviour that seems brave but hides information.	Courage includes openness; acting alone and quietly is not courage.

Exam-style questions 15 questions in the online lesson

1. Which list contains only Scrum values?

- A

 Commitment, trust, quality, openness, focus
- B

 Commitment, focus, openness, respect, courage
- C

 Respect, speed, openness, courage, focus
- D

 Transparency, inspection, adaptation, focus, courage

2. A Developer says in the Daily Scrum: "I'm stuck and I need help." Which value is most visible?

- A

 Focus
- B

 Commitment
- C

 Respect
- D

 Openness

3. A tester refuses to call an untested feature Done, despite pressure from a director. Which value is shown?

- A

 Courage
- B

 Focus
- C

 Respect
- D

 Commitment

4. A company pays bonuses for individual completed tasks. What is the main risk?

- A

 It discourages helping one another
- B

 It is required for transparency
- C

 It has no link to the values
- D

 It improves focus on the Sprint Goal

Answers 1: B The five values in the Guide. · 2: D Being open about challenges. · 3: A Doing the right thing on a tough problem. · 4: A The way Scrum is used should reinforce the values.

All 15 questions with full explanations →

Free PSM I questions →

The Scrum Team

Three accountabilities, one team, no hierarchy, and who decides what.

Lesson 6 The Scrum Team

Lesson 7 Developers

Lesson 8 Product Owner

Lesson 9 Scrum Master

Lesson 10 Self-management and leadership



LESSON 6 OF 26 · PART C · THE SCRUM TEAM

The Scrum Team

A Scrum Team is **one Scrum Master, one Product Owner and Developers**, working as one unit on one Product Goal. It has every skill it needs, it decides how to do its own work, and the whole team answers for the result.

🕒 online

What you will learn

1. Describe who is in a Scrum Team, and why it has no sub-teams and no hierarchy.
2. Explain cross-functional and self-managing, including what self-management does *not* cover.
3. Apply the size guidance and the rule for several teams working on one product.
4. Tell the accountability of the whole team apart from the three specific accountabilities.
5. Recognize what organizations do that weakens a Scrum Team, and what a Scrum Master does about it.

Why it matters on the exam

The Scrum Team appears in many PSM I questions, often hidden inside a situation about managers, team leads, scaling or quality. Questions rarely ask "who is in the team?" They ask what to do when someone breaks one of these rules.

Simple explanation

A Scrum Team is a small group of professionals who build a product together. It has three accountabilities:

- **One Product Owner.** Accountable for making the product as valuable as possible.
- **One Scrum Master.** Accountable for the team's effectiveness and for making Scrum work.
- **Developers.** Everyone who creates the product: programmers, designers, testers, engineers, analysts.

Inside the team there are **no sub-teams** (no "test team" or "design team") and **no hierarchy** (no boss, no team lead). Everyone works toward **one objective at a time: the Product Goal**.

The team is **cross-functional**. Together, the members have all the skills needed to create value every Sprint. Not every person can do everything, but the team as a whole can.

The team is **self-managing**. It decides internally who does what, when and how. Nobody outside the team assigns tasks to its members.

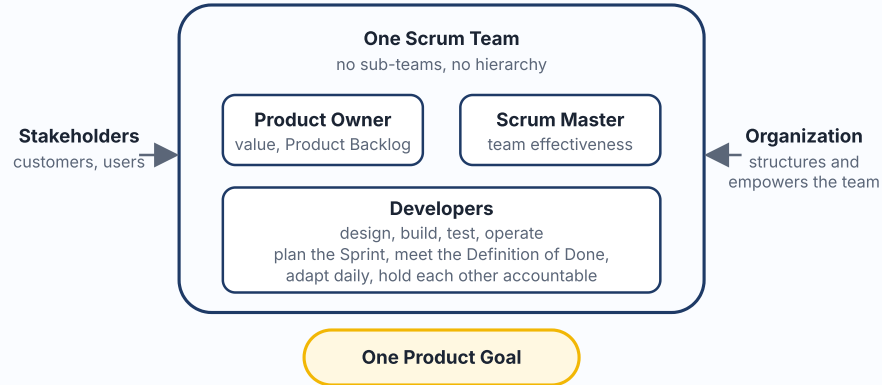
The team is **small**, typically 10 or fewer people. If it grows too large, it should consider splitting into several teams that work on the same product. Those teams share **one Product Goal, one Product Backlog and one Product Owner**.

Finally, the **entire Scrum Team** is accountable for creating a valuable, useful Increment every Sprint. Not only the Developers. Everyone.

"The Scrum Team consists of one Scrum Master, one Product Owner, and Developers. Within a Scrum Team, there are no sub-teams or hierarchies. It is a cohesive unit of professionals focused on one objective at a time, the Product Goal."

The Scrum Guide, 2020, Scrum Team

See it in one picture Lesson 6 · The Scrum Team



Everyone inside the boundary shares one goal and one result. Stakeholders and managers are outside the team: they collaborate with it, but they do not direct its daily work.



Paths = $n \times (n - 1) \div 2$. Tripling the team from 5 to 15 people multiplies the connections by more than ten. This is why the Guide says smaller teams communicate better. The formula is a general rule of thumb, not part of the Scrum Guide.

Your memory card

1. A Scrum Team is one Scrum Master, one Product Owner and Developers, with no sub-teams and no hierarchy.
2. It focuses on one objective at a time: the Product Goal.
3. Cross-functional: the team as a whole has every skill needed, and shares or acquires skills as needed.
4. Self-managing: the team decides who does what, when and how, within the Scrum framework and the organization's standards.
5. Typically 10 or fewer people. Several teams on one product share one Product Goal, one Product Backlog and one Product Owner.
6. The team is responsible for all product-related activities, and the entire team is accountable for a valuable, useful Increment every Sprint.
7. The Scrum Master coaches self-management and causes the removal of impediments; they do not manage the team's work.

Official source: The Scrum Guide 2020, section **Scrum Team** (with Developers, Product Owner and Scrum Master). Read it at scrumguides.org or as the official PDF. Read the section once after this lesson; it is about one page.

[Open the full lesson →](#)

Deeper professional explanation Lesson 6 · The Scrum Team

Knowing the rules is not enough for PSM I. You need to know *why* each rule exists, because exam situations test what happens when someone breaks it.

Why no sub-teams?

A sub-team creates a handoff. Work waits in a queue between "development" and "testing". Problems hide in that queue, so the team's picture of its progress is wrong. That is a **transparency** problem. Handoffs also split responsibility: each group finishes "its part", and nobody owns the usable result. Scrum avoids this by making the whole team answer for one Increment.

Why no hierarchy?

When a team lead decides, the people closest to the work wait for permission. Learning still happens, but decisions slow down. The Scrum Guide says adaptation becomes more difficult when people are not empowered or self-managing. A hierarchy inside the team also weakens peer accountability. Developers are expected to hold each other accountable as professionals. That is equal-to-equal feedback, not management.

What self-management covers, and what it does not

Self-managing means the team internally decides **who does what, when, and how**. This is real authority over its own work. It is not freedom from everything:

- The **Product Owner** orders the Product Backlog. The team does not set product priorities alone.
- The **organization** may set standards. If the Definition of Done is an organizational standard, every Scrum Team must follow it as a minimum.
- The **Scrum framework** itself is not negotiable. A self-managing team cannot decide to drop the Sprint Review.

The Guide also says Scrum Teams are "structured and empowered by the organization to manage their own work." Self-management depends on the organization. A team cannot self-manage if managers keep assigning its tasks.

Cross-functional does not mean "everyone does everything"

The team as a whole has all the skills to create value each Sprint. Scrum theory adds that teams "share or acquire such skills as needed". If a skill is missing, the team makes that visible and closes the gap: by learning, pairing, or bringing the skill into the team. Waiting weeks for an outside department every Sprint is a sign the team is not yet cross-functional.

Deeper professional explanation (continued)

Lesson 6 · The Scrum Team

Why "typically 10 or fewer", and not a hard limit

Read the wording carefully: the team is *typically* 10 or fewer, and large teams *should consider* reorganizing. This is guidance based on experience, not a fixed number. The reason is communication. The number of person-to-person connections in a group grows quickly: 5 people have 10 connections, 10 people have 45, and 15 people have 105. More connections mean more coordination and slower decisions. The diagram in section E shows this.

Responsible for everything the product needs

The Guide says the Scrum Team is responsible for all product-related activities: stakeholder collaboration, verification, maintenance, operation, experimentation, research and development, and anything else required. If a separate group maintains the product after release, feedback reaches the Scrum Team late and the team learns slowly. A Scrum Master should make this kind of split visible.

Whole-team accountability

The entire Scrum Team is accountable for a valuable, useful Increment every Sprint. Inside that shared outcome, each accountability has its own focus: the Product Owner for value and the Product Backlog, the Developers for the Sprint plan, quality and daily adaptation, and the Scrum Master for the team's effectiveness. When an Increment is weak, the useful question is not "whose fault?" but "what will we inspect and adapt as a team?"

What a Scrum Master actually does here

The Guide says the Scrum Master serves the team by coaching members in self-management and cross-functionality, and by causing the removal of impediments. In practice, that means noticing a team lead forming, a sub-team appearing or a

manager assigning work, and helping the team and the organization fix it. Often the Scrum Master does this by teaching and coaching, not by deciding.

How this compares with traditional project management

In many traditional settings, a project manager plans the work and assigns tasks. That can work well when the work is well known and stable, for example repeating a proven installation. Scrum is built for complex work, where the people doing the work learn the most, the fastest. So in Scrum, the Developers plan and assign their own work, and the Scrum Master builds the team's ability to do this. Neither approach is "bad". They fit different kinds of work.

Beyond software: a robotic cell project An automation company builds a robotic welding cell. Traditionally, mechanical design finishes, then hands drawings to electrical, then to PLC programming, then to robot programming. Each handoff hides problems until integration. A cross-functional Scrum Team puts a mechanical designer, an electrical designer, a PLC programmer and a robot programmer in one team with one goal. Each Sprint ends with a tested, usable piece, for example a gripper station that runs safely on its own. Problems appear in weeks instead of at the factory acceptance test.

Case study: FreshCart grows Decide before you read the answer (cover the green part)

Decision 1: The reorganization

After Sprint 4

The FreshCart app launched in two cities, and orders are growing fast. Mr. Grant, the director, has hired 9 more people. The team now has Maya (Product Owner), Sam (Scrum Master) and 14 Developers.

Mr. Grant sends Sam his plan: "Leo becomes development lead and assigns the daily work. Tom leads a new 4-person QA team that tests everything before release. And to stop people stepping on each other, let's split the Product Backlog into an 'app' backlog and a 'delivery' backlog."

Some Developers like the plan. They are tired of long Daily Scrums with 16 people.

What should Sam propose?

- ☐ A Accept the QA team, because testing really is a bottleneck, but ask to keep one Product Backlog.
- ☒ B Propose two cross-functional Scrum Teams, each able to create Done Increments, sharing one Product Goal, one Product Backlog and Maya as Product Owner, with no team lead, and let the Developers help decide how to form the teams.
- ☐ C Keep one team of 16 and solve the long Daily Scrum by strictly enforcing 15 minutes.
- ☐ D Accept Leo as lead for now, since the team needs coordination, and review the structure in a few Sprints.

Best decision: B. Sam proposes two teams, Ordering and Delivery. Each has every skill needed, including testing. Both teams pull from Maya's single Product Backlog and work toward the same Product Goal. The Developers choose the team split themselves in a workshop. Within two Sprints, Daily Scrums are short again, and nothing waits for a separate test team.

Why not the others? A creates a sub-team and a handoff, which the Guide does not allow. C keeps a team that is too large to communicate well; enforcing the timebox treats the symptom, not the cause. D introduces a hierarchy; "for now" structures tend to become permanent, and the Developers stop self-managing.

Principle: when a team grows too large, it should consider reorganizing into multiple cohesive Scrum Teams on the same product, sharing the same Product Goal, Product Backlog and Product Owner.

Decision 2: The director's shortcut

Sprint 6, day 3

The Ordering team's Sprint Goal is "Customers can apply a promotion code at checkout". On day 3, Mr. Grant walks to Priya's desk. A large corporate customer is complaining that invoices show the wrong tax. "Drop the promotion work and fix the invoices today. I've already told the customer it will be fixed tomorrow."

Priya asks Sam what to do. Maya, the Product Owner, is in a meeting until the afternoon.

What is the best response?

- ☐ A Priya fixes the invoices immediately. The director has the authority, and an angry customer is a real business risk.
- ☐ B Sam tells Mr. Grant the request must wait for the next Sprint, because the Sprint Backlog cannot change.
- ☒ C Bring the request to Maya as soon as possible. Maya decides how important it is against other work; the Developers decide how to fit it in, adapting their plan with Maya if the Sprint scope must change. Sam later helps Mr. Grant route requests through the Product Owner.
- ☐ D Sam fixes the invoice bug himself, so the Developers can stay focused on the Sprint Goal.

Best decision: C. Sam sends Maya a short message. Maya steps out of her meeting for ten minutes and agrees the tax error is urgent. The Developers decide that Kenji fixes the invoices while the others continue the promotion code work. They tell Maya that one low-priority part of the promotion feature may move to the next Sprint, and Maya agrees. The Sprint Goal is still reached.

Why not the others? A bypasses the Product Owner and the Developers' own planning. It also teaches the director that the fastest way to get work done is to go around the team. B is too rigid: the Guide allows scope to be clarified and renegotiated with the Product Owner during the Sprint, as long as the Sprint Goal is not endangered. D is a well-meant mistake: it hides the work from the team's plan and makes the Scrum Master a backup developer instead of helping the system work.

Principle: those who want to change the Product Backlog do so by convincing the Product Owner. During the Sprint, scope may be renegotiated with the Product Owner without endangering the Sprint Goal.

Common misunderstandings

- ✗ **"Self-managing means nobody can tell the team anything."**

✓ Self-management covers who does what, when and how. The Product Owner still orders the Product Backlog, organizational standards for the Definition of Done still apply, and the Scrum framework still holds.
- ✗ **"Cross-functional means every person must be able to do every job."**

✓ The team as a whole has the skills. Individuals can specialize, and the team shares or acquires skills as needed.
- ✗ **"A Scrum Team of 11 people is breaking the rules."**

✓ The Guide says typically 10 or fewer, and that large teams should consider reorganizing. It is guidance to be applied with evidence, not a hard limit.
- ✗ **"The Scrum Master is the team's manager."**

✓ The Scrum Master is accountable for the team's effectiveness, and serves the team, the Product Owner and the organization. Managing tasks and people's daily work is not part of the accountability.
- ✗ **"Only the Developers are accountable for the Increment."**

✓ The entire Scrum Team is accountable for creating a valuable, useful Increment every Sprint.

Exam traps

The trap	How to read it
Confusing "the Scrum Team" with "the Developers".	Check which group the question names. The whole Scrum Team is accountable for the Increment; specific accountabilities belong to Developers, Product Owner or Scrum Master.
Treating "should consider reorganizing" as "must split".	The Guide's size wording is advisory. An answer that turns guidance into a strict rule overstates the Guide.
Treating advice as forbidden. Example: "Scrum forbids part-time team members."	The Guide does not say this. Look for the answer that names the real problem (focus, transparency) without inventing a rule.
Reading peer feedback as hierarchy.	Developers holding each other accountable is required, and it is not management.
Accepting a separate maintenance or QA group as normal.	The Scrum Team is responsible for all product-related activities, including verification, maintenance and operation.
Old terms: "Development Team", "self-organizing", "roles".	The 2020 Guide uses Developers, self-managing and accountabilities. Answers using old terms are not automatically wrong, but read them carefully.

Exam-style questions 15 questions in the online lesson

1. Which statement about the Scrum Team is accurate?

- ☐ A It consists of one Scrum Master, one Product Owner and Developers, with no sub-teams or hierarchies.
- ☐ B It consists of a Product Owner, a Scrum Master, Developers and a technical lead who coordinates the Developers.
- ☐ C It consists of the Developers; the Product Owner and Scrum Master support it from outside.
- ☐ D It consists of specialist sub-teams that coordinate through the Scrum Master.

2. In a new Scrum Team, Developers report to a "development lead" who assigns their tasks every morning. The team struggles to adapt when problems appear mid-Sprint. What should the Scrum Master do first?

- ☐ A Ask the development lead to assign tasks for the whole Sprint during Sprint Planning, so the team has a stable plan.
- ☐ B Coach everyone on self-management, and treat the hierarchy as an impediment to remove.
- ☐ C Take over task assignment, so the development lead can focus on technical work.
- ☐ D Accept it, because Scrum is purposefully incomplete and lets organizations decide how work is assigned.

3. A team of 7 has no database skills. Every Sprint it waits for an external database department, and items are often not Done by the end of the Sprint. Which response is most consistent with Scrum?

- ☐ A Accept the waiting time and plan for it, since outside specialists are normal in large companies.
- ☐ B Create a database sub-team.
- ☐ C Make the dependency visible and bring the skill into the team.
- ☐ D Ask the Product Owner to stop ordering database-related items near the top of the Product Backlog.

4. Three Scrum Teams work on one product. To avoid conflicts, each team asks for its own Product Backlog, and two teams want their own Product Owners. Management agrees, saying it increases team autonomy. What is the best response?

- ☐ A Support it. Self-managing teams may choose the structure that works best for them.
- ☐ B One PO, separate backlogs.
- ☐ C Keep one Product Goal, backlog and PO; resolve conflicts together.
- ☐ D Merge the three teams into one large team, so there is nothing to coordinate.

Answers 1: **A** This is the Guide's own description of the Scrum Team. · 2: **B** The root problem is a hierarchy inside the team. Coaching self-management and helping the organization remove the impediment are both in the Scrum Master's accountability. · 3: **C** A cross-functional team has all the skills needed to create value each Sprint, and Scrum teams share or acquire skills as needed. · 4: **C** Teams focused on the same product share the same Product Goal, Product Backlog and Product Owner. The conflicts are real, but splitting the backlog hides them instead of solving them.

[All 15 questions with full explanations →](#)

[Free PSM I questions →](#)

LESSON 7 OF 26 · PART C · THE SCRUM TEAM

Developers

Developers are everyone in the Scrum Team who creates the product. They are **always accountable** for four things: **the Sprint plan, quality through the Definition of Done, adapting the plan daily** toward the Sprint Goal, and **holding each other accountable**.

⌚ About 30 minutes, including the practice questions online

What you will learn

1. Name the four accountabilities of the Developers.
2. Explain why "Developers" covers every skill, not only programming.
3. Separate what Developers decide alone from what they agree with the Product Owner.
4. Explain peer accountability without hierarchy.
5. Recognize outside attempts to direct the Developers' work.

Why it matters on the exam

Questions test who decides what. Developers decide how and who; the Product Owner orders the work; scope changes during the Sprint are negotiated between them.

Simple explanation

Developers are the people in the Scrum Team committed to creating any aspect of a usable Increment each Sprint. Their skills vary with the domain: programmers, designers, testers, engineers, analysts, researchers.

The Guide says Developers are always accountable for:

1. Creating a plan for the Sprint, the Sprint Backlog.
2. Instilling quality by adhering to a Definition of Done.
3. Adapting their plan each day toward the Sprint Goal.
4. Holding each other accountable as professionals.

"Developers are the people in the Scrum Team that are committed to creating any aspect of a usable Increment each Sprint."

The Scrum Guide, 2020, Developers

See it in one picture Lesson 7 · Developers

Developers alone

how to build it
who does which task
tools and techniques
sizing items
daily plan changes

Developers + PO

which items to select
the Sprint Goal (whole team)
scope changes in the Sprint

Product Owner

order of the backlog
the Product Goal
cancelling a Sprint

Most exam questions about Developers come down to which column a decision belongs in.

Your memory card

1. Developers are everyone creating the Increment.
2. Always accountable for: the Sprint plan, quality via the DoD, daily adaptation, and holding each other accountable.
3. How and who are decided by the Developers alone.
4. Item selection and scope changes are agreed with the Product Owner.
5. Developers size Product Backlog items.
6. Peer accountability is not hierarchy.

Official source: Scrum Guide 2020: **Developers**, **Sprint Planning** and **Definition of Done** at scrumguides.org.

[Open the full lesson →](#)

Deeper professional explanation Lesson 7 · Developers

What Developers decide alone

How to turn selected items into an Increment is at the sole discretion of the Developers. So is who does which task, when, and with which tools, within any organizational standards. Nobody, including the Product Owner or Scrum Master, tells them how.

What Developers agree with the Product Owner

The Developers select items for the Sprint through discussion with the Product Owner. If the work turns out different than expected during the Sprint, they negotiate the scope with the Product Owner, without endangering the Sprint Goal. Adding or dropping items is a scope change; changing tasks inside the plan is not.

Quality belongs to the Developers

The Developers are required to conform to the Definition of Done. Quality is not something checked afterwards by a separate department or approved by the Product Owner. Work that does not meet the Definition of Done is not part of the Increment.

Peer accountability is not hierarchy

Developers hold each other accountable as professionals: "this does not meet our Definition of Done" between equals. That is different from a team lead approving work. The first strengthens self-management; the second replaces it.

Sizing

The Developers who will do the work are responsible for sizing Product Backlog items. The Product Owner may influence them by helping them understand and select trade-offs, but does not set the size.

Example: a marketing team A campaign team uses Scrum. Its copywriter, designer and data analyst are all Developers. When the analyst finds that the planned landing page will not load fast enough on phones, the Developers change their approach the same day. Because the Sprint Goal (more sign-ups from mobile users) is unchanged, they only inform the Product Owner.

Case study: Who decides? Decide before you read the answer (cover the green part)

Decision 1: The design order

Sprint 9

Maya, the Product Owner, tells the Developers to build the new checkout with a specific technical framework she read about, saying it will "increase value".

How should the Developers respond?

- ☐ A Use her framework, since she is accountable for value
- ☒ B Listen to her value concern, and decide the technical approach themselves
- ☐ C Ask the Scrum Master to choose the framework
- ☐ D Escalate to Mr. Grant to settle who decides

Best decision: B. The Developers explain why their chosen approach meets her concern about checkout speed. Maya is satisfied once she sees the trade-off.

Why not the others? A gives away the Developers' decision. C asks the wrong person. D escalates something the Guide already settles.

Decision 2: The bigger item

Sprint 9, day 5

The Developers find that one selected item will take three times longer than forecast. The Sprint Goal can still be reached without it.

What should they do?

- ☐ A Drop it quietly, since the plan is theirs
- ☐ B Work overtime so every item is finished
- ☒ C Tell the Product Owner and renegotiate the scope, keeping the goal
- ☐ D Lower the Definition of Done for that item

Best decision: C. Maya agrees to move the item to the next Sprint. The Sprint Goal is met at a sustainable pace.

Why not the others? A changes scope without the Product Owner. B ignores sustainable pace. D lowers quality.

Common misunderstandings

- ✗ "Developers are programmers."

✓ Developers are everyone creating the Increment, whatever their skill.
- ✗ "The Product Owner approves quality."

✓ The Developers conform to the Definition of Done; quality is built in.
- ✗ "Developers can change the Sprint's scope on their own."

✓ How and who are theirs; adding or dropping items is negotiated with the PO.
- ✗ "Holding each other accountable needs a team lead."

✓ Peers do it as professionals, without hierarchy.

Exam traps

The trap	How to read it
"The Product Owner decides the technical approach because of value."	How is at the Developers' sole discretion.
"Developers may drop selected items silently."	Scope changes are negotiated with the Product Owner.
"Developers adapt the plan only at the Daily Scrum."	They adapt each day, and whenever they learn something.

Exam-style questions 12 questions in the online lesson

1. Which is one of the four things Developers are always accountable for?

A Approving each Increment

B Adapting their plan daily toward the goal

C Keeping every Scrum event within its timebox

D Ordering the Product Backlog for the next Sprint
2. A data analyst on a Scrum Team asks if she counts as a Developer. What is accurate?

A Yes, if she helps create the Increment

B Only if she also writes some of the production code

C No, only engineers who build software are Developers

D No, analysts are outside it
3. The Developers want to switch tools mid-Sprint to reach the goal faster. Whose approval do they need?

A The Scrum Master's

B The line manager's, who controls the tool budget

C Nobody's, within org standards

D The Product Owner's, since cost affects value
4. Which decisions belong to the Developers alone? (Choose 2)

A The order of the Product Backlog

B Whether to cancel the Sprint

C Who does which task

D How to build the selected items

Answers 1: B Plan, quality, adapt, peers. · 2: A Developers create any aspect of the Increment. · 3: C How is at the Developers' discretion. · 4: C and D How and who are self-management decisions.

All 12 questions with full explanations →

Free PSM I questions →

LESSON 8 OF 26 · PART C · THE SCRUM TEAM

Product Owner

The Product Owner is **one person**, accountable for **maximizing the value of the product** and for **effective Product Backlog management**. The organization must **respect their decisions**, and stakeholders influence the backlog by **convincing the Product Owner**.

🕒 About 35 minutes, including the practice questions online

What you will learn

1. State the Product Owner's accountabilities and the four parts of backlog management.
2. Explain delegation without losing accountability.
3. Handle conflicting stakeholders while keeping one ordered Product Backlog.
4. Explain why the Product Owner is one person, not a committee.
5. Recognize when the organization undermines the Product Owner.

Why it matters on the exam

Common traps: committees, deputy Product Owners, stakeholders going around the Product Owner, and Product Owners telling Developers how to work.

Simple explanation

The Product Owner is accountable for maximizing the value of the product resulting from the work of the Scrum Team. They are also accountable for effective Product Backlog management, which includes:

- Developing and explicitly communicating the Product Goal.
- Creating and clearly communicating Product Backlog items.
- Ordering Product Backlog items.
- Ensuring the Product Backlog is transparent, visible and understood.

The Product Owner may do this work or delegate it to others. Either way, they remain accountable.

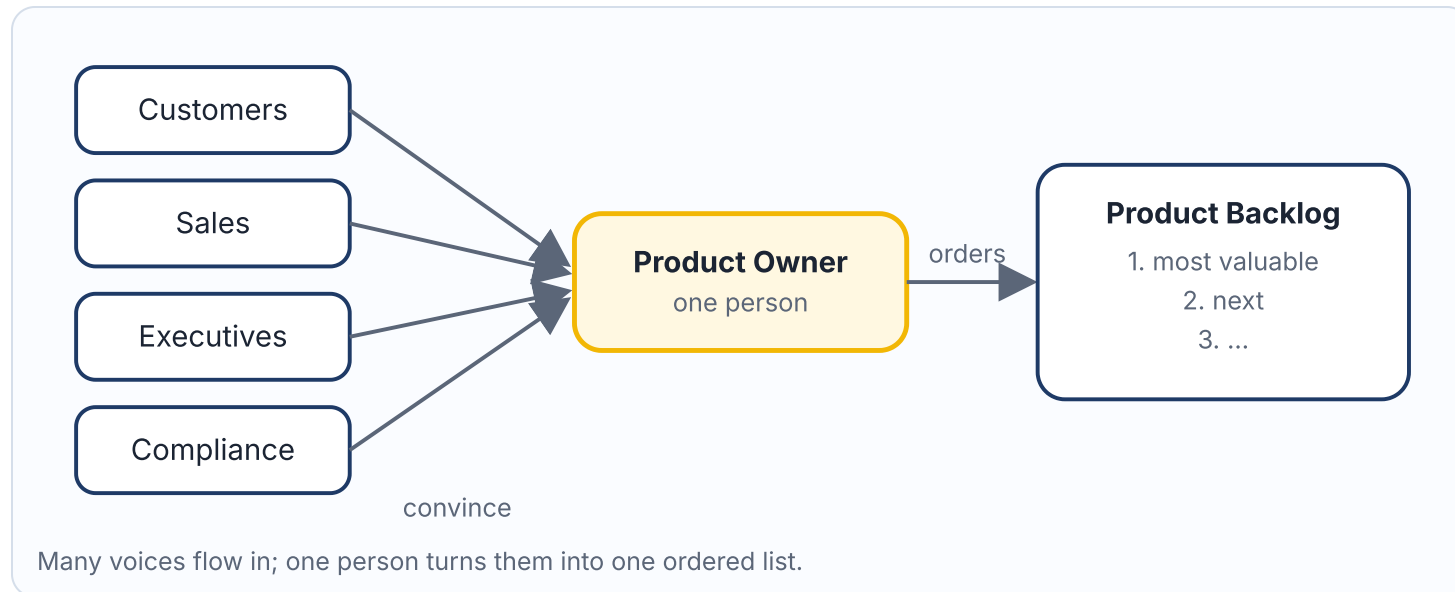
"The Product Owner is one person, not a committee. The Product Owner may represent the needs of many stakeholders in the Product Backlog. Those wanting to change the Product Backlog can do so by trying to convince the Product Owner."

The Scrum Guide, 2020, Product Owner

"For Product Owners to succeed, the entire organization must respect their decisions."

The Scrum Guide, 2020, Product Owner

See it in one picture Lesson 8 · Product Owner



Your memory card

1. The Product Owner is one person, accountable for maximizing product value.
2. Backlog management: Product Goal, clear items, ordering, transparency.
3. Delegating work does not delegate accountability.
4. Stakeholders influence by convincing the Product Owner.
5. The organization must respect the Product Owner's decisions.
6. Only the Product Owner may cancel a Sprint, when its goal is obsolete.

Official source: Scrum Guide 2020: **Product Owner** and **Product Backlog** at scrumguides.org.

[Open the full lesson →](#)

Deeper professional explanation Lesson 8 · Product Owner

Value, not volume

Maximizing value is not the same as delivering everything stakeholders ask for, or keeping everyone happy. It often means saying no, and reordering when evidence shows a feature is not used. Value can also include things users never see, such as compliance work that allows the product to be sold.

Why one person

A committee spreads accountability so that nobody truly owns the order of work. One person can decide quickly and be held to account. That person may represent many stakeholders; they just make the final ordering decision.

Where the Product Owner's decisions show

The Guide says the Product Owner's decisions are visible in the content and ordering of the Product Backlog, and through the inspectable Increment at the Sprint Review.

Limits of the role

The Product Owner does not tell Developers how to build, does not size items, and does not assign tasks to people. During the Sprint, they cannot add work to the Sprint Backlog on their own; scope is renegotiated with the Developers, protecting the Sprint Goal. The Product Owner alone has the authority to cancel a Sprint, and only when the Sprint Goal becomes obsolete.

When the organization undermines the Product Owner

Executives assigning work directly to Developers, steering committees reordering the backlog, and separate departmental backlogs all weaken the accountability. The Scrum Master helps the organization respect it.

Example: a hospital scheduling product Doctors want faster booking, nurses want fewer double bookings, administrators want cost reports. The Product Owner orders the backlog by value, explains the reasoning at the Sprint Review, and changes the order when real usage data shows double bookings cause the most harm.

Case study: Many voices, one order Decide before you read the answer (cover the green part)

Decision 1: The steering committee

Month 4

FreshCart's leadership forms a steering committee and proposes that it approve the Product Backlog order every month, "to give Maya support".

What is the best response from the Scrum Master?

- ☐ A Accept, if Maya chairs the committee
- ☒ B Keep Maya as the one person who orders the backlog; let the committee influence her
- ☐ C Let the committee own the backlog, with Maya as coordinator
- ☐ D Ask the Developers to vote between committee and PO

Best decision: B. The committee meets monthly to share strategy with Maya. She orders the backlog and explains her decisions at Sprint Reviews. Decisions stay fast.

Why not the others? A and C turn the Product Owner into a committee. D gives ordering to the wrong people.

Decision 2: The CEO's shortcut

Sprint 11

The CEO tells two Developers to start a new feature immediately, without speaking to Maya. The current Sprint Goal is still valuable.

What should the Developers do?

- ☐ A Start the feature, because the CEO outranks Maya
- ☐ B Refuse to discuss it
- ☒ C Bring the request to Maya, who weighs it against other work
- ☐ D Ask the Scrum Master to approve it

Best decision: C. Maya adds the feature to the Product Backlog near the top for the next Sprint. The Scrum Master later helps the CEO route requests through Maya.

Why not the others? A bypasses the Product Owner. B lacks respect. D gives the Scrum Master an authority it does not have.

Common misunderstandings

- ✗ "A committee can be the Product Owner."

✓ The Product Owner is one person, not a committee.
- ✗ "Delegating backlog work moves the accountability."

✓ The Product Owner remains accountable.
- ✗ "The Product Owner's job is to keep stakeholders happy."

✓ The job is maximizing value, which often means saying no.
- ✗ "The Product Owner can add work to the current Sprint at any time."

✓ Scope during the Sprint is renegotiated with the Developers, protecting the Sprint Goal.

Exam traps

The trap	How to read it
"Each team on the same product may have its own PO."	Teams on one product share one Product Owner and one Product Backlog.
"The Product Owner sizes items."	Developers size; the PO may help with trade-offs.
"Senior stakeholders may reorder the backlog."	They influence by convincing the Product Owner.

Exam-style questions 12 questions in the online lesson

1. What is the Product Owner accountable for?

A

Sizing Product Backlog items

B

Maximizing product value

C

The Sprint plan

D

The Scrum Team's effectiveness
2. A Product Owner asks an analyst to write most backlog items. Who is accountable?

A

The Product Owner

B

The Scrum Master

C

The analyst who writes them

D

The PO and the analyst equally
3. Two directors demand different items at the top of the backlog. Who decides?

A

The Scrum Master

B

The Developers by vote

C

The Product Owner

D

The more senior director
4. A steering committee wants to approve the backlog order monthly. What is the best response?

A

Make the committee members deputy Product Owners

B

Accept if the PO chairs it

C

Let the committee order, with the PO acting as an adviser

D

One PO orders; the committee influences

Answers

1: B Value. · 2: A Delegation does not move accountability. · 3: C Stakeholders influence by convincing the PO. · 4: D The PO is one person, not a committee.

- All 12 questions with full explanations →
- Free PSM I questions →

LESSON 9 OF 26 · PART C · THE SCRUM TEAM

Scrum Master

The Scrum Master is accountable for **establishing Scrum** and for the **Scrum Team's effectiveness**. They are **true leaders who serve** the Scrum Team, the Product Owner and the wider organization, mostly by teaching, coaching, facilitating and **causing the removal of impediments**.

🕒 About 40 minutes, including the practice questions online

What you will learn

1. State the two accountabilities of the Scrum Master.
2. List how the Scrum Master serves the team, the Product Owner and the organization.
3. Explain "causing the removal of impediments" without doing everything personally.
4. Tell serving leadership from managing.
5. Choose between teaching, coaching, facilitating and stepping back.

Why it matters on the exam

Many questions offer a tempting option where the Scrum Master decides, assigns or approves. Those are almost never the best answer. The best answer usually enables others to act.

Simple explanation

The Scrum Master is accountable for establishing Scrum as defined in the Scrum Guide, by helping everyone understand Scrum theory and practice, both within the Scrum Team and the organization. They are also accountable for the Scrum Team's effectiveness, by enabling the team to improve its practices within the Scrum framework.

Serving the Scrum Team: coaching self-management and cross-functionality; helping focus on high-value Increments that meet the Definition of Done; causing the removal of impediments; ensuring events take place and are positive, productive and within the timebox.

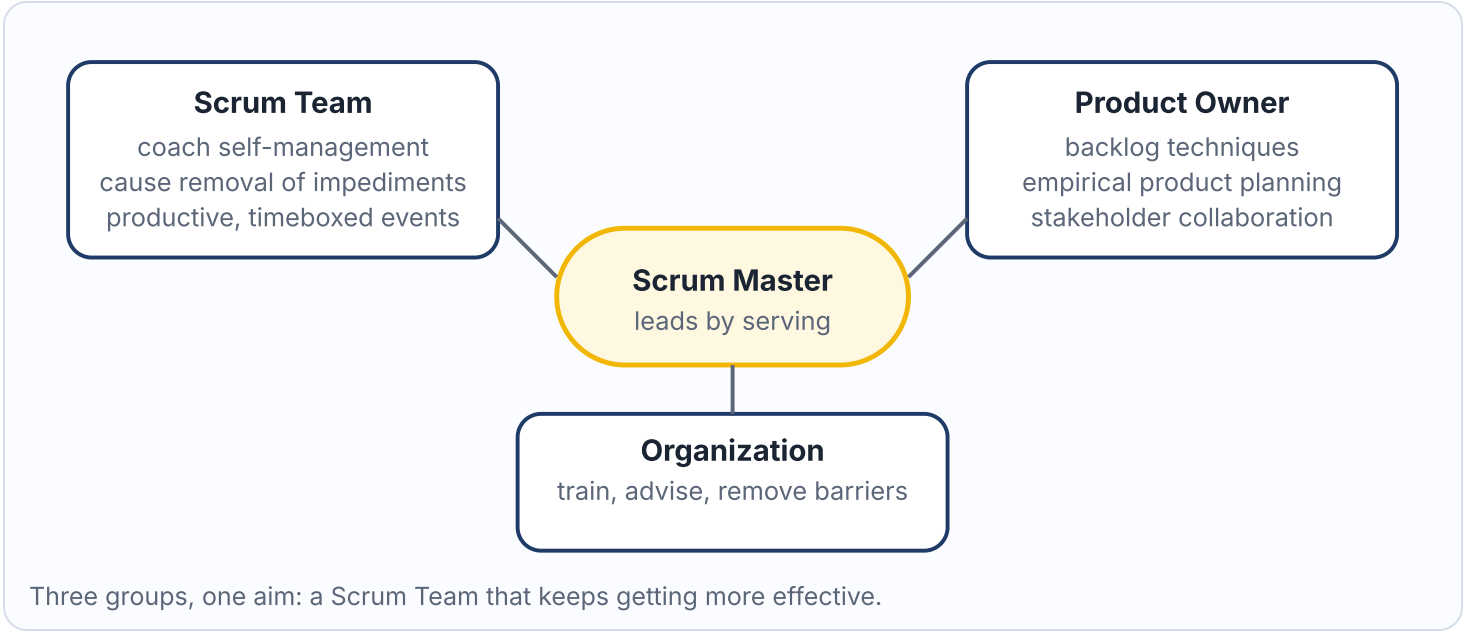
Serving the Product Owner: helping find techniques for the Product Goal and backlog management; helping the team understand the need for clear items; helping establish empirical product planning; facilitating stakeholder collaboration as requested or needed.

Serving the organization: leading, training and coaching Scrum adoption; planning and advising Scrum implementations; helping people understand an empirical approach; removing barriers between stakeholders and Scrum Teams.

"Scrum Masters are true leaders who serve the Scrum Team and the larger organization."

The Scrum Guide, 2020, Scrum Master

See it in one picture Lesson 9 · Scrum Master



Your memory card

1. The Scrum Master is accountable for establishing Scrum and the team's effectiveness.
2. True leaders who serve the team, the Product Owner and the organization.
3. They cause the removal of impediments, often by enabling others.
4. They do not assign tasks, rate people or make technical decisions.
5. Choose the right stance: teach, coach, facilitate or step back.
6. Working on Sprint items does not pause the Scrum Master accountability.

Official source: Scrum Guide 2020: **Scrum Master** at scrumguides.org.

[Open the full lesson →](#)

Deeper professional explanation Lesson 9 · Scrum Master

Leader, not manager

The Scrum Master does not assign tasks, approve plans, rate people or make technical decisions. Their influence comes from teaching, coaching, facilitating and changing the system around the team. When a Scrum Master starts deciding for the team, self-management weakens and the next problem comes back to them too.

"Causing the removal" of impediments

The wording matters. Sometimes the Scrum Master removes an impediment directly, for example by persuading a department to change a policy. Often the best move is helping the Developers solve it themselves. A Scrum Master who personally fixes everything creates dependence.

Choosing a stance

A new team often needs **teaching**: explaining Scrum. A growing team needs **coaching**: asking questions so people find their own answers. Difficult conversations need **facilitating**. Mature teams often need the Scrum Master to **step back**. The skill is choosing the right one for the moment.

Working on Sprint items

A Scrum Master may also work on Sprint Backlog items; in the Daily Scrum they then participate as a Developer. The Scrum Master accountability does not pause while they do.

A Scrum Master compared with a traditional project manager

In many traditional settings a project manager plans and assigns work, which can suit stable, well-known work. In Scrum the Developers plan and assign their own

work, and the Scrum Master builds their ability to do so. Neither is "bad"; they fit different kinds of work.

Case study: Sam earns his role Decide before you read the answer (cover the green part)

Decision 1: The task dispute

Sprint 12

Two Developers argue about who should take a difficult integration task. They ask Sam, the Scrum Master, to decide so they can move on.

What should Sam do?

- ☐ A Decide, since he is accountable for effectiveness
- ☒ B Coach them to agree between themselves
- ☐ C Ask Maya to assign the task
- ☐ D Escalate to their line managers

Best decision: B. Sam asks what each needs to succeed. They agree to pair on the task. Next time, they settle it without him.

Why not the others? A removes a self-management decision. C gives the PO a role the Guide does not. D adds hierarchy.

Decision 2: The release approval

Sprint 14

Every release needs a signature from a security office that meets once a month. Done Increments wait weeks to reach users.

What is Sam's best approach?

- ☐ A Sign the approvals himself to save time
- ☐ B Tell the Developers to skip the approval
- ☒ C Work with the security office and the team on a faster way to meet the standard
- ☐ D Ask Maya to remove security work from the backlog

Best decision: C. Sam brings the security lead to a Sprint Review. They agree to add security checks to the Definition of Done, and the monthly signature is replaced by a review of the checks. Releases go out weekly.

Why not the others? A exceeds his authority. B breaks a real standard. D ignores a real requirement.

Common misunderstandings

- ✗ "The Scrum Master is the team's manager."

✓ They serve and coach; they do not assign or rate.
- ✗ "The Scrum Master must remove every impediment personally."

✓ They cause the removal, often by enabling others.
- ✗ "The Scrum Master is a meeting scheduler."

✓ They are accountable for establishing Scrum and team effectiveness.
- ✗ "A Scrum Master may serve only one team."

✓ The Guide does not say this; what matters is that each team is served well.

Exam traps

The trap	How to read it
An option where the Scrum Master decides for the team.	Usually wrong; look for the option that helps the team decide.
"The Scrum Master orders the backlog when the PO is busy."	Ordering is the PO's accountability; the SM helps with techniques.
"The Scrum Master approves the Sprint Backlog."	The Sprint Backlog belongs to the Developers.

Exam-style questions 12 questions in the online lesson

1. What is the Scrum Master accountable for?

A Approving each Increment before release

B Establishing Scrum and team effectiveness

C Creating the Sprint plan with the Developers

D Maximizing the value of the product for users
2. Developers ask the Scrum Master to decide who takes a task. What is best?

A Coach them to decide

B Ask the Product Owner to assign the task

C Decide to save time

D Set up a fair task rotation for them
3. A security sign-off delays every release by weeks. What fits the Scrum Master best?

A Tell the Developers to skip the sign-off for now

B Ask the PO to remove security work from the backlog

C Work with security on a faster way

D Sign them personally
4. After a year, a team cannot hold any event without its Scrum Master. What does this suggest?

A The team clearly needs a second Scrum Master

B Events are too long

C Flawless facilitation is the core of the SM's job

D It built dependence, not capability

Answers

1: B Scrum established, team effective. · 2: A Self-managing teams decide who does what. · 3: C Cause removal of impediments; remove barriers. · 4: D The SM coaches self-management.

All 12 questions with full explanations →

Free PSM I questions →

LESSON 10 OF 26 · PART C · THE SCRUM TEAM

Self-management and leadership

A self-managing team decides **who does what, when and how**. That freedom works **inside the Scrum framework** and the organization's standards, and it needs an organization that **structures and empowers** the team. Leadership in Scrum means **creating the conditions** for this, not directing the work.

🕒 About 30 minutes, including the practice questions online

What you will learn

1. Define self-management and its boundaries.
2. Explain the organization's role in empowering Scrum Teams.
3. Describe what managers and leaders can usefully do around a Scrum Team.
4. Recognize behaviours that quietly remove self-management.
5. Label common facilitation and decision techniques as optional practices.

Why it matters on the exam

Scenarios often show a manager, team lead or Scrum Master taking a decision that belongs to the team. The best answer restores the team's authority while respecting real limits such as organizational standards.

Simple explanation

Self-managing means the Scrum Team internally decides who does what, when and how. Nobody outside assigns its tasks.

Self-management has boundaries:

- The **Scrum framework** itself is not up for change; the team cannot drop events or accountabilities.
- The **Product Owner** orders the Product Backlog.
- Organizational standards still apply; for example, an organizational Definition of Done is a minimum.

The Guide says Scrum Teams are **structured and empowered by the organization** to manage their own work. Self-management is not something a team can do alone.

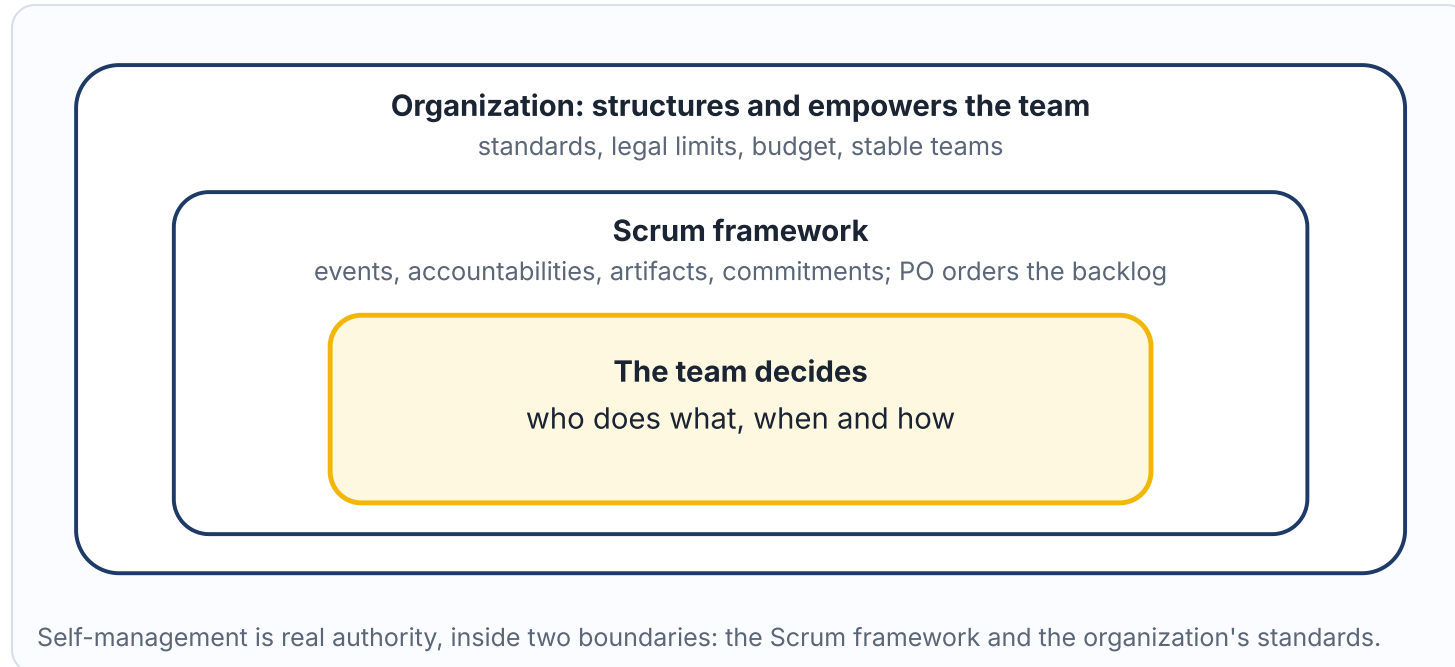
"They are also self-managing, meaning they internally decide who does what, when, and how."

The Scrum Guide, 2020, Scrum Team

"Scrum Teams are structured and empowered by the organization to manage their own work."

The Scrum Guide, 2020, Scrum Team

See it in one picture Lesson 10 · Self-management and leadership



Your memory card

1. Self-managing teams decide who does what, when and how.
2. That authority sits inside the Scrum framework and organizational standards.
3. Organizations structure and empower teams; self-management cannot work alone.
4. Leaders create conditions: stable teams, clear constraints, fewer barriers.
5. Self-management erodes through small, well-meant interventions.
6. Facilitation and voting techniques are optional practices.

Official source: Scrum Guide 2020: **Scrum Team**, **Scrum Master** and **Adaptation** at scrumguides.org.

[Open the full lesson →](#)

Deeper professional explanation Lesson 10 · Self-management and leadership

How self-management quietly disappears

It rarely ends with an announcement. It erodes when a manager assigns "just this one" task, when a senior Developer's approval becomes expected, when the Scrum Master decides to save time, or when every tool change needs a signature. Each step looks harmless; together they move decisions away from the people closest to the work. The Guide notes that adaptation becomes more difficult when people are not empowered or self-managing.

What leaders can do

The Guide does not describe a manager accountability inside Scrum. Leaders around Scrum Teams add value by setting clear goals and constraints, keeping teams stable, removing organizational barriers, providing tools and training, and respecting the Product Owner's decisions. They step out of daily task decisions.

The Scrum Master's role in self-management

The Scrum Master coaches the team in self-management and cross-functionality. That includes not becoming the team's decision-maker, and helping the organization change structures that block self-management, such as individual reward schemes or constant reassignment.

Facilitation and decision techniques

Teams often use techniques such as time-boxed discussions, voting methods or consent-based decisions to decide quickly together. These are helpful tools, not part of the Scrum Guide. Use whatever helps the team decide well within its timeboxes.

Disagreement is healthy

Self-managing teams disagree. The goal is not to remove conflict but to resolve it productively, guided by the Scrum values of respect, openness and courage. A team that never disagrees is often hiding problems.

Example: a municipal permits team A city's permits team adopted Scrum, but each change to a form needed the director's approval. Sprint Reviews kept showing problems nobody could fix quickly. The director agreed to set the constraints (legal requirements, budget) and let the team decide changes within them. Waiting times for permits dropped within three Sprints.

Case study: The helpful manager Decide before you read the answer (cover the green part)

Decision 1: "Just this one task"

Sprint 15

FreshCart's operations manager often walks to Kenji's desk with "just one small task". Kenji likes helping. The team's plan keeps slipping, and nobody knows why.

What is the best first step?

- ☐ A Kenji keeps helping, since the tasks are small
- ☒ B Make the pattern visible in the Retrospective and route requests through the Product Owner
- ☐ C The Scrum Master forbids the manager from entering the team area
- ☐ D The team adds a buffer of hidden capacity for such tasks

Best decision: B. The team discovers these "small" tasks took a full day per week. The manager now sends requests to Maya, who orders them with everything else. The team's forecasts become reliable again.

Why not the others? A hides work. C is hostile and solves nothing. D hides the problem instead of removing it.

Decision 2: The team that wants rules

Sprint 16

New team members feel lost and ask Sam, the Scrum Master, to "just tell us who does what each Sprint".

What is the most helpful response?

- ☐ A Assign tasks for a few Sprints, then stop
- ☒ B Teach how the team can plan its own work, and coach them as they start
- ☐ C Ask Maya to assign tasks instead
- ☐ D Tell them self-management means figuring it out alone

Best decision: B. Sam runs a short session on planning the Sprint together, then coaches the first two Plannings. By the third Sprint the team plans without him.

Why not the others? A creates dependence that is hard to undo. C moves the dependence to the PO. D abandons a team that needs support.

Common misunderstandings

- ✖ "Self-managing teams need no management at all."

 ✔ Organizations still structure and empower teams and set standards.
- ✖ "Self-management means the team can drop Scrum events."

 ✔ It works within the framework, not over it.
- ✖ "A good Scrum Master decides quickly for the team."

 ✔ The Scrum Master coaches self-management; deciding for the team weakens it.
- ✖ "Voting methods are part of Scrum."

 ✔ They are optional practices, not in the Guide.

Exam traps

The trap	How to read it
"Self-managing teams may set aside organizational standards."	An organizational DoD is a minimum; standards still apply.
An option where a manager assigns daily tasks "to help".	Who does what is decided inside the team.
"Self-managing" vs "self-organizing".	The 2020 Guide uses self-managing.

Exam-style questions 12 questions in the online lesson

- What does self-managing mean in the Scrum Guide?

A Deciding who does what, when and how
 B Working without a Scrum Master once mature
 C Changing Scrum's rules
 D Setting its own Product Goal without the PO
- A manager often gives Developers "small" tasks directly. What is the best first step?

A Accept it; tasks are small
 B Keep some hidden spare capacity for such requests
 C Forbid the manager from talking to the Developers
 D Make it visible; route requests via the PO
- The organization requires a security standard in the DoD. A Developer says this violates self-management. What is accurate?

A The Scrum Master may waive the standard for a Sprint
 B Standards set a minimum for the team
 C Only the PO decides
 D Self-managing teams may set aside slow standards
- Which behaviours quietly remove self-management? (Choose 2)

A Developers pairing to share skills
 B The team choosing its own tools
 C A senior Developer's approval becoming expected
 D Managers assigning tasks "just this once"

Answers
 1: A Who, what, when, how.
 · 2: D Transparency, and work enters through the PO.
 · 3: B Definition of Done section.
 · 4: C and D Decisions move away from the team.

All 12 questions with full explanations →

Free PSM I questions →

Events

The Sprint and the four events inside it: purpose, timebox and traps.

Lesson 11 The Sprint

Lesson 12 Sprint Planning

Lesson 13 Daily Scrum

Lesson 14 Sprint Review

Lesson 15 Sprint Retrospective



LESSON 11 OF 26 · PART D · EVENTS

The Sprint

The Sprint is a **fixed-length event of one month or less** that contains all the other events. A new Sprint starts **immediately** after the last. During the Sprint the **Sprint Goal is protected, quality does not decrease**, and scope can be clarified with the Product Owner.

⌚ About 30 minutes, including the practice questions online

What you will learn

1. State the Sprint's length rule and why it is fixed.
2. Apply the four rules that hold during a Sprint.
3. Explain when and by whom a Sprint can be cancelled.
4. Explain how Sprints create predictability and limit risk.
5. Use forecasting practices without replacing empiricism.

Why it matters on the exam

Expect scenarios about extending Sprints, gaps between Sprints, mid-Sprint scope changes, cancellation, and quality shortcuts near the end of a Sprint.

Simple explanation

Sprints are the heartbeat of Scrum. Each one is a fixed length of one month or less, and a new Sprint starts right after the previous one ends. Sprint Planning, the Daily Scrums, the Sprint Review and the Sprint Retrospective all happen inside it.

During the Sprint:

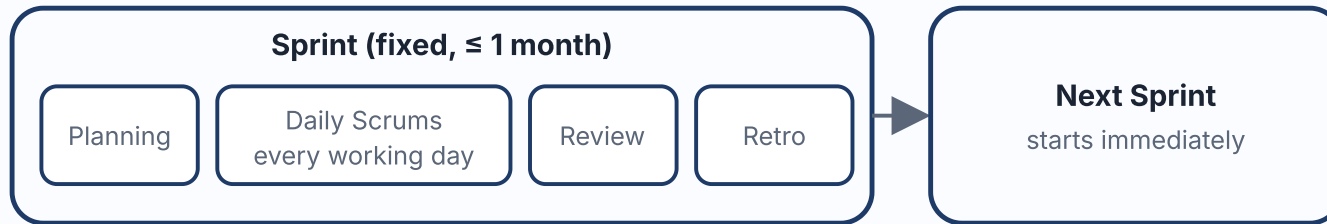
- No changes are made that would endanger the Sprint Goal.
- Quality does not decrease.
- The Product Backlog is refined as needed.
- Scope may be clarified and renegotiated with the Product Owner as more is learned.

A Sprint could be cancelled if the Sprint Goal becomes obsolete. Only the Product Owner has the authority to cancel it.

"Sprints enable predictability by ensuring inspection and adaptation of progress toward a Product Goal at least every calendar month."

The Scrum Guide, 2020, The Sprint

See it in one picture Lesson 11 · The Sprint



Goal protected · quality kept · scope may be renegotiated with the PO

No gaps between Sprints, and no events outside them.

Your memory card

1. Sprints are fixed length, one month or less, back to back.
2. The Sprint contains all other events and all the work.
3. Goal protected, quality kept, backlog refined, scope renegotiable with the PO.
4. Only the PO may cancel, and only for an obsolete goal.
5. Sprints create predictability through monthly or faster inspection.
6. Forecasting tools help but never replace empiricism.

Official source: Scrum Guide 2020: **The Sprint** at scrumguides.org.

[Open the full lesson →](#)

Deeper professional explanation Lesson 11 · The Sprint

Why fixed length

Fixed length creates consistency. If a Sprint stretches whenever work is unfinished, it stops being a timebox and becomes a reaction to scope. The rhythm is what lets the team and stakeholders inspect progress reliably.

Why one month or less

When a Sprint's horizon is too long, the Guide says the Sprint Goal may become invalid, complexity may rise and risk may increase. Shorter Sprints generate more learning cycles and limit the risk of cost and effort to a smaller time frame. Each Sprint may be considered a short project.

Scope is flexible, the goal and quality are not

As the team learns, the exact work may change: items can be clarified, split or swapped with the Product Owner's agreement. What stays fixed is the Sprint Goal (unless it becomes obsolete) and the quality bar. Cutting a test the Definition of Done requires is never the way to finish on time.

Cancellation is rare and specific

A Sprint is cancelled only when its goal becomes obsolete, for example after a market or regulatory change. A goal that is merely hard to reach is not obsolete; the team adapts instead. The 2020 Guide does not detail what happens to Done work after cancellation; keeping useful Done work is sensible.

Forecasting practices

Burn-downs, burn-ups and cumulative flows may be useful, but they do not replace empiricism. In complex work, only what has already happened may be used for forward-looking decisions.

Case study: A Sprint under pressure Decide before you read the answer (cover the green part)

Decision 1: "Just three more days"

Last day of Sprint 17

Two items are almost finished. Mr. Grant asks the team to extend the Sprint by three days so the Review looks better.

What is the best response?

- ☐ A Extend once, since the work is nearly finished
- ☒ B Keep the fixed length; unfinished items return to the Product Backlog
- ☐ C Mark the items Done and finish them next Sprint
- ☐ D Skip the Review and start the next Sprint early

Best decision: B. The Sprint ends on time. The Review shows the Done work honestly, and Maya places the two items at the top of the backlog.

Why not the others? A breaks the timebox. C misrepresents quality. D removes an inspection event.

Decision 2: The new regulation

Sprint 18, day 4

A new regulation bans the payment method the whole Sprint Goal depends on. The Developers have already finished one useful refactoring.

What is most appropriate?

- ☐ A The Developers cancel the Sprint immediately
- ☒ B The Product Owner may cancel the Sprint, since its goal is obsolete
- ☐ C The team continues, since Sprints are never cancelled
- ☐ D The Scrum Master extends the Sprint to replan

Best decision: B. Maya cancels the Sprint. The team keeps the Done refactoring and starts a new Sprint Planning with a fresh goal.

Why not the others? A and D use authority they do not have. C ignores an obsolete goal.

Common misunderstandings

- ✗ "A Sprint can be extended to finish the work."

✓ Sprints are fixed length; unfinished work returns to the Product Backlog.
- ✗ "A hard Sprint Goal is a reason to cancel."

✓ Only an obsolete goal is; a hard goal calls for adaptation.
- ✗ "Teams need a gap between Sprints to prepare."

✓ A new Sprint starts immediately after the previous one.
- ✗ "Burn-down charts are required."

✓ They are optional practices that never replace empiricism.

Exam traps

The trap	How to read it
"Quality may be reduced to protect the Sprint Goal."	Quality does not decrease during the Sprint.
"The Developers can cancel the Sprint."	Only the Product Owner has that authority.
"Sprints must be two weeks."	Any fixed length of one month or less.

Exam-style questions 12 questions in the online lesson

1. What is the maximum length of a Sprint?

A Two weeks

B Six weeks

C Up to the PO

D One month
2. On day 9 of 10, two items are not Done. A manager asks to extend the Sprint. What is best?

A Extend by a few days this once

B Keep the length; return the items to the backlog

C Cancel the Sprint and restart it

D Mark the items Done now and quietly finish them next Sprint
3. Mid-Sprint, the team finds an item is much bigger than expected. The goal can still be met. What is allowed?

A Lower the DoD for that one item so it fits

B Ask the SM to cancel

C Extend the Sprint

D Renegotiate scope with the PO
4. Which statements hold during a Sprint? **(Choose 2)**

A Quality does not decrease

B The goal may change daily

C The Product Backlog is refined as needed

D The Sprint Backlog is frozen

Answers

1: D One month or less. · 2: B Sprints are fixed length. · 3: D Scope may be clarified and renegotiated. · 4: A and C Quality holds; refinement continues.

All 12 questions with full explanations →

Free PSM I questions →

LESSON 12 OF 26 · PART D · EVENTS

Sprint Planning

Sprint Planning starts the Sprint by answering three questions: **why** is this Sprint valuable, **what** can be Done, and **how** will it get done. The result is the **Sprint Backlog**: the Sprint Goal, the selected items and the plan.

⌚ About 30 minutes, including the practice questions online

What you will learn

1. Describe the three topics of Sprint Planning.
2. Explain who proposes, who decides and who plans.
3. Use past performance, capacity and the Definition of Done to forecast.
4. Write a Sprint Goal that gives focus and flexibility.
5. Apply the timebox and know what may be left to emerge.

Why it matters on the exam

Questions test ownership: the Product Owner proposes value, the whole Scrum Team defines the Sprint Goal, the Developers select items with the PO and decide how.

Simple explanation

Sprint Planning is attended by the whole Scrum Team; others may be invited to give advice. The Product Owner makes sure attendees are prepared to discuss the most important items and how they connect to the Product Goal.

- **Topic one, why:** the Product Owner proposes how the product could increase its value; the whole team defines a Sprint Goal.
- **Topic two, what:** through discussion with the Product Owner, the Developers select items. They may refine items during this process.
- **Topic three, how:** the Developers plan the work needed to create a Done Increment, often by breaking items into work of one day or less.

Sprint Planning is timeboxed to a maximum of eight hours for a one-month Sprint, usually shorter for shorter Sprints. The Sprint Goal must be finalized before it ends.

"How this is done is at the sole discretion of the Developers. No one else tells them how to turn Product Backlog items into Increments of value."

The Scrum Guide, 2020, Sprint Planning

See it in one picture Lesson 12 · Sprint Planning

Why?

PO proposes value
whole team defines
the Sprint Goal

What?

Developers select items
with the PO
refine if needed

How?

Developers plan
at their sole discretion
often day-sized work

Result: the Sprint Backlog (goal + items + plan)

Each topic has a clear owner. Most Planning questions test exactly that.

Your memory card

1. Three topics: why, what, how.
2. The PO proposes value; the whole team defines the Sprint Goal.
3. Developers select items with the PO and decide how.
4. Forecast from past performance, capacity and the DoD.
5. Up to eight hours for a one-month Sprint; the goal is fixed before it ends.
6. The plan emerges; start with enough detail.

Official source: Scrum Guide 2020: **Sprint Planning** and **Sprint Backlog** at scrumguides.org.

[Open the full lesson →](#)

Deeper professional explanation Lesson 12 · Sprint Planning

Forecasting with evidence

The Guide says the more the Developers know about their past performance, their upcoming capacity and their Definition of Done, the more confident their forecast will be. A manager's hope or a stakeholder's wish is not a basis for selection.

A good Sprint Goal

"Finish items 1 to 6" is a list, not a goal. A good Sprint Goal states an outcome, such as "Customers can track their delivery in real time". It explains why the Sprint matters and leaves room to change the exact work. That flexibility is what makes mid-Sprint adaptation possible without losing direction.

The plan does not need to be complete

The Sprint Backlog emerges during the Sprint. Planning needs enough detail to start and to inspect progress at the Daily Scrum; the rest is planned as the team learns.

When the Product Owner is unprepared

If the Product Owner cannot explain why the top items matter, the Sprint Goal will be weak. The Scrum Master can help the Product Owner with techniques, and continuous refinement keeps top items ready.

Case study: Planning that works Decide before you read the answer (cover the green part)

Decision 1: Twelve items or eight

Sprint Planning 19

Maya wants twelve items in the Sprint for a big customer. Based on recent Sprints and two holidays, the Developers forecast eight.

Who decides, and how?

- ☐ A Maya, because she is accountable for value
- ☒ B The Developers select, after discussing trade-offs with Maya
- ☐ C The Scrum Master, as a neutral party
- ☐ D The team votes, with Maya counted twice

Best decision: B. The Developers take eight items and a clear Sprint Goal. Maya tells the customer what to expect. The forecast is met.

Why not the others? A overrides the Developers. C and D are not the Guide's process.

Decision 2: A goal that is a list

Sprint Planning 20

The team's draft Sprint Goal reads: "Complete items 14, 15, 16 and 17."

What is the best improvement?

- ☐ A Keep it; the list is clear
- ☒ B Restate it as the outcome those items achieve
- ☐ C Add all task details to the goal
- ☐ D Let the Scrum Master write the goal

Best decision: B. The goal becomes "Repeat customers can reorder in one tap". When item 16 proves harder than expected, the team finds a simpler way to reach the same outcome.

Why not the others? A removes flexibility. C confuses goal and plan. D moves ownership.

Common misunderstandings

✗ **"The Product Owner decides how much work goes into the Sprint."**

✓ The Developers select, through discussion with the PO.

✗ **"The plan must cover every task before the Sprint starts."**

✓ The Sprint Backlog emerges; enough detail to start is enough.

✗ **"Sprint Planning always takes eight hours."**

✓ Eight hours is the maximum for a one-month Sprint.

✗ **"The Sprint Goal is the list of selected items."**

✓ It is the single objective that gives the items purpose.

Exam traps

The trap	How to read it
"Each item must be broken into tasks of exactly one day."	The Guide says this is often done, not required.
"The Sprint Goal can be finalized at the first Daily Scrum."	It must be finalized before Sprint Planning ends.
"Only the Scrum Team may attend."	Others may be invited to provide advice.

Exam-style questions 12 questions in the online lesson

1. What are the three topics of Sprint Planning?

A Why, what and how it gets done

B Plan, do and check

C Review, plan and commit

D Who, when and where
2. Who defines the Sprint Goal?

A The Product Owner alone

B The Developers alone

C The Scrum Master

D The whole Scrum Team
3. The PO insists on twelve items; the Developers forecast eight. Who selects?

A The Scrum Master

B The Developers, with the PO

C A team vote

D The PO, since value is theirs
4. Planning ends with a clear goal but only half the tasks planned. What is accurate?

A The Sprint must be cancelled

B The SM finishes the plan

C The Sprint starts; the plan keeps emerging

D Planning must continue past the timebox until it is complete

Answers · **1: A** Sprint Planning section. · **2: D** PO proposes; the whole team defines. · **3: B** Developers select. · **4: C** The Sprint Backlog is updated as more is learned.

All 12 questions with full explanations →

Free PSM I questions →

LESSON 13 OF 26 · PART D · EVENTS

Daily Scrum

The Daily Scrum is a **15-minute event for the Developers**, held every working day. Its purpose is to **inspect progress toward the Sprint Goal** and **adapt the Sprint Backlog**, producing an actionable plan for the next day.

🕒 About 25 minutes, including the practice questions online

What you will learn

1. State the purpose, timebox and attendees of the Daily Scrum.
2. Explain why the Developers choose its format.
3. Fix a Daily Scrum that has become a status meeting.
4. Explain that the plan can change at any time, not only at the Daily Scrum.
5. Describe how the Product Owner and Scrum Master take part.

Why it matters on the exam

Classic traps: the three questions as a rule, reporting to the Scrum Master or a manager, extending the timebox for problem solving, and believing plans change only at this event.

Simple explanation

The Daily Scrum is held at the same time and place every working day of the Sprint, to reduce complexity. It lasts 15 minutes. The Developers can use any structure and techniques, as long as it focuses on progress toward the Sprint Goal and produces an actionable plan for the next day.

If the Product Owner or Scrum Master are actively working on Sprint Backlog items, they take part as Developers.

Daily Scrums improve communication, identify impediments, promote quick decisions, and so remove the need for other meetings.

"The Daily Scrum is not the only time Developers are allowed to adjust their plan. They often meet throughout the day for more detailed discussions about adapting or re-planning the rest of the Sprint's work."

The Scrum Guide, 2020, Daily Scrum

See it in one picture Lesson 13 · Daily Scrum

Status meeting (avoid)

people report to one person
focus on yesterday's tasks
plan never changes
problems "noted"

Daily Scrum (aim for)

Developers talk to each other
focus on the Sprint Goal
plan adapted for tomorrow
impediments acted on

Same 15 minutes, very different value.

Your memory card

1. 15 minutes, every working day, same time and place.
2. An event for the Developers; they choose the format.
3. Purpose: inspect progress toward the Sprint Goal and adapt the plan.
4. Output: an actionable plan for the next day.
5. Plans can change at any time; detailed talks happen afterward.
6. PO and SM join as Developers only when working on Sprint items.

Official source: Scrum Guide 2020: **Daily Scrum** at scrumguides.org.

[Open the full lesson →](#)

Deeper professional explanation Lesson 13 · Daily Scrum

From status meeting to planning event

A Daily Scrum where each person reports to the Scrum Master, the Product Owner or a manager has lost its purpose. Signs: people look at one person while speaking, nobody changes the plan, and problems are "noted" rather than acted on. The fix is to refocus on one question: *given where we are, what is our plan for reaching the Sprint Goal?*

The three questions are gone

Older Guides suggested three questions. The 2020 Guide removed them and lets the Developers choose any format: walking the board, focusing on the riskiest item, or something else.

Keep it short; solve problems after

The timebox is fixed. Detailed technical discussions happen afterward, with only the people who need to be there. A 40-minute Daily Scrum usually means problem solving has moved into the wrong place.

Every working day

Teams sometimes skip it when "nothing changed". But the event exists to discover whether something has. Skipping events loses opportunities to inspect and adapt.

Case study: Fixing the morning meeting Decide before you read the answer (cover the green part)

Decision 1: Reporting to Sam

Sprint 21

Each Developer reads yesterday's tasks aloud to Sam, the Scrum Master. Nobody discusses the Sprint Goal, and integration problems keep appearing at the end of the Sprint.

What should Sam do?

- ☐ A Ask for more detailed task reports
- ☒ B Help the Developers refocus on progress toward the goal, and step back from leading
- ☐ C Replace the event with a written update
- ☐ D Ask Maya to run it instead

Best decision: B. Sam stops facilitating. The Developers start each Daily Scrum by looking at what still blocks the goal. Integration issues surface on day 2 instead of day 9.

Why not the others? A deepens the reporting. C removes the event. D changes the audience, not the purpose.

Decision 2: The 40-minute Daily Scrum

Sprint 22

The Daily Scrum regularly runs 40 minutes because two Developers debate technical details.

What is the best fix?

- ☐ A Extend the timebox for complex work
- ☒ B Keep it to 15 minutes; those involved continue afterward
- ☐ C Ban technical topics
- ☐ D Hold it only twice a week

Best decision: B. The Daily Scrum ends in 12 minutes. The two Developers and one colleague solve the issue straight after.

Why not the others? A breaks the timebox. C overcorrects. D removes daily inspection.

Common misunderstandings

- ✗ "The Daily Scrum uses three fixed questions."

✓ The 2020 Guide lets the Developers choose any structure.
- ✗ "The Scrum Master must attend and run it."

✓ The SM ensures it happens; the Developers own it.
- ✗ "Plans may only change at the Daily Scrum."

✓ Developers can adjust their plan at any time.
- ✗ "A short Daily Scrum is too short."

✓ 15 minutes is a maximum, not a target.

Exam traps

The trap	How to read it
"The timebox can grow for larger teams."	It is 15 minutes regardless.
"The Daily Scrum's purpose is to update the PO."	Its purpose is inspecting progress toward the Sprint Goal and adapting the plan.
"Skip it when nothing has changed."	It is held every working day.

Exam-style questions 12 questions in the online lesson

1. What is the purpose of the Daily Scrum?

A Solve all technical problems

B Inspect goal progress and adapt the plan

C Update the PO on every item

D Report each person's status to the Scrum Master
2. Who decides the format of the Daily Scrum?

A The Developers

B The Product Owner

C The Scrum Master

D The manager
3. A Developer finds a blocker at 2 p.m. A colleague says to wait for tomorrow's Daily Scrum. What is accurate?

A Ask the SM for approval first

B Call an extra Daily Scrum

C The plan can be changed right now

D Wait for the Daily Scrum
4. A Daily Scrum always runs 40 minutes due to technical debates. What fits best?

A Ban technical discussions entirely

B Hold it twice a week instead

C Extend the timebox so that complex work can be discussed

D Keep 15 minutes; discuss details after

Answers 1: B Inspect and adapt daily. · 2: A Developers choose structure and techniques. · 3: C The Daily Scrum is not the only time to adjust. · 4: D Developers meet throughout the day for detail.

All 12 questions with full explanations →

Free PSM I questions →

LESSON 14 OF 26 · PART D · EVENTS

Sprint Review

The Sprint Review is a **working session** where the Scrum Team and key stakeholders **inspect the outcome of the Sprint** and **decide what to do next**. The Product Backlog may be adjusted. It is **not a release gate** and not just a demo.

🕒 About 30 minutes, including the practice questions online

What you will learn

1. State the purpose, timing and attendees of the Sprint Review.
2. Turn a one-way demo into a collaborative working session.
3. Explain why undone work is not presented.
4. Explain why the Review is never a gate to release.
5. Distinguish the Review from the Retrospective.

Why it matters on the exam

Common traps: treating the Review as an approval gate, showing undone work, presenting slides only, skipping it when the Increment is small, and confusing it with the Retrospective.

Simple explanation

The Sprint Review is the second-to-last event of the Sprint, timeboxed to a maximum of four hours for a one-month Sprint (usually shorter for shorter Sprints).

The Scrum Team presents the results of its work to key stakeholders, and progress toward the Product Goal is discussed. Together they review what was accomplished and what has changed in the environment, and collaborate on what to do next. The Product Backlog may be adjusted to meet new opportunities.

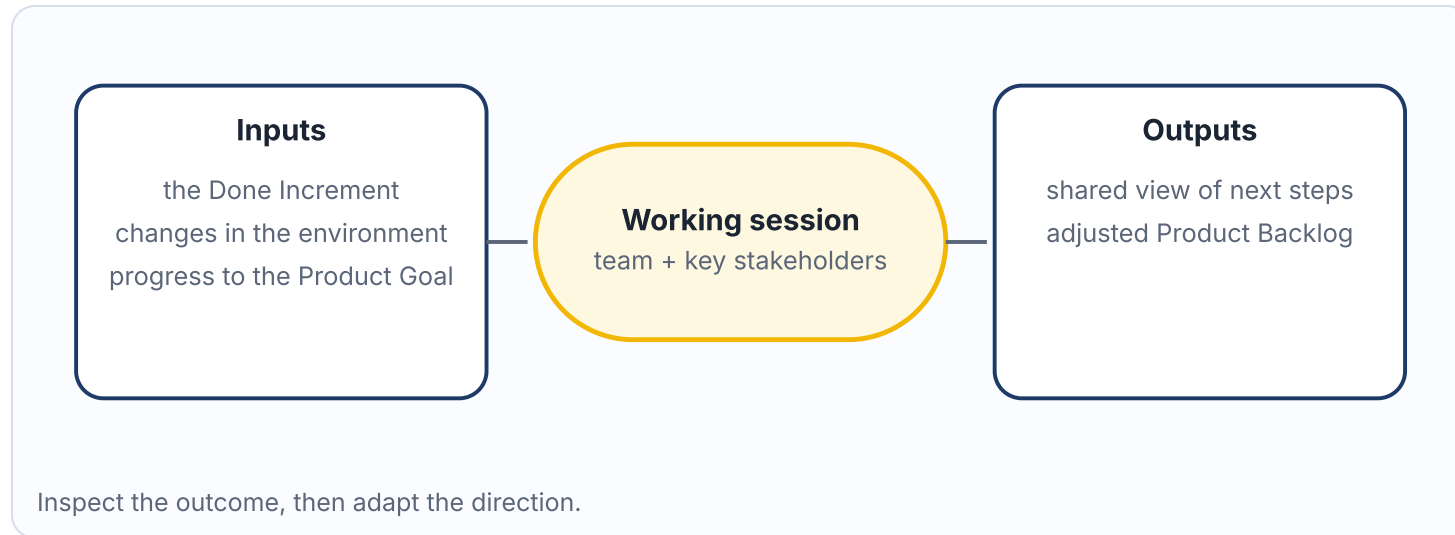
"The Sprint Review should never be considered a gate to releasing value."

The Scrum Guide, 2020, Increment

"The Scrum Team should avoid limiting it to a presentation."

The Scrum Guide, 2020, Sprint Review

See it in one picture Lesson 14 · Sprint Review



Your memory card

1. Purpose: inspect the Sprint's outcome and decide future adaptations.
2. A working session with key stakeholders, not a presentation.
3. Covers what was done, what changed in the environment and progress to the Product Goal.
4. The Product Backlog may be adjusted.
5. Undone work is not presented; the Review is never a release gate.
6. Up to four hours for a one-month Sprint; second to last event.

Official source: Scrum Guide 2020: **Sprint Review** and **Increment** at scrumguides.org.

[Open the full lesson →](#)

Deeper professional explanation Lesson 14 · Sprint Review

From demo to working session

A polished presentation with polite applause misses the point. A good Review shows the real Increment, discusses what changed in the market, budget or timeline, and ends with a shared view of what comes next, often a reordered Product Backlog.

Only Done work is shown as the Increment

If an item does not meet the Definition of Done, it cannot be released or even presented at the Review; it returns to the Product Backlog. Showing "almost done" work creates a false picture of progress.

Not a release gate

Increments may be delivered to stakeholders before the Sprint ends. The Review inspects the outcome and the direction; it does not approve releases. A director who uses it to sign off features is turning it into a gate.

Review versus Retrospective

The Review looks at the **product** and what to do next, with stakeholders. The Retrospective looks at **how the team works**, within the Scrum Team.

Case study: The Review that changed direction Decide before you read the answer (cover the green part)

Decision 1: The polished demo

Sprint 23

FreshCart's Reviews are slide shows. Stakeholders clap and leave. Usage data shows a new feature is barely used, but nobody has mentioned it.

What would most improve the Review?

- ☐ A Better slides and a longer demo
- ☒ B A working session on the real Increment, usage data, and next steps
- ☐ C Emailing the slides instead
- ☐ D Holding Reviews only for big releases

Best decision: B. The next Review shows the live app and the usage numbers. Stakeholders agree to drop two planned features and improve search instead. Maya reorders the backlog on the spot.

Why not the others? A and C keep it one-way. D removes inspection.

Decision 2: "Release it anyway"

Sprint 24 Review

A key customer sees a feature that works but has not passed security testing required by the Definition of Done, and demands it be released today.

What is the correct response?

- ☐ A Release it with a warning
- ☒ B It is not Done, so it is not released; it returns to the backlog
- ☐ C Release it if Maya agrees
- ☐ D Let the Scrum Master decide

Best decision: B. Maya places it at the top of the backlog. It passes testing and ships early in the next Sprint.

Why not the others? A and C release undone work. D is not the SM's decision.

Common misunderstandings

- ✗ "The Sprint Review is a demo."

✓ It is a working session; the team should avoid limiting it to a presentation.
- ✗ "Features are approved for release at the Review."

✓ The Review is never a gate to releasing value.
- ✗ "Almost-done work can be shown if labelled."

✓ Work not meeting the DoD is not presented as part of the Increment.
- ✗ "Small Increments don't need a Review."

✓ The Review inspects the outcome and the environment every Sprint.

Exam traps

The trap	How to read it
"The Review is the last event of the Sprint."	It is second to last; the Retrospective concludes the Sprint.
"The team inspects its processes at the Review."	That belongs to the Retrospective.
"The Review timebox is exactly 4 hours."	Four hours is the maximum for a one-month Sprint.

Exam-style questions 12 questions in the online lesson

1. What is the purpose of the Sprint Review?

A

 Plan the next Sprint's tasks

B

 Inspect the outcome and decide next steps

C

 Evaluate each Developer

D

 Approve the Increment for release to customers
2. Stakeholders leave every Review without giving feedback. What helps most?

A

 A working session on real results and next steps

B

 Emailing the slides

C

 More polished slides

D

 Hold Reviews only when there is a big release to show
3. A feature has not passed testing the DoD requires. May it be shown at the Review as part of the Increment?

A

 Yes, if the PO agrees

B

 Only to internal stakeholders, never to customers

C

 No; it returns to the Product Backlog

D

 Yes, if clearly labelled
4. A director signs off each feature at the Review like a contract inspection. What is the main concern?

A

 Only the SM may sign off

B

 Contracts are outside Scrum

C

 Sign-offs must happen at the Daily Scrum

D

 It turns the Review into a release gate

Answers 1: B Outcome and next steps. · 2: A Avoid limiting it to a presentation. · 3: C Definition of Done section. · 4: D The Review should never be a gate.

All 12 questions with full explanations →

Free PSM I questions →

LESSON 15 OF 26 · PART D · EVENTS

Sprint Retrospective

The Sprint Retrospective **concludes the Sprint**. The Scrum Team inspects **individuals, interactions, processes, tools and its Definition of Done**, and plans ways to **increase quality and effectiveness**. The most helpful changes are addressed as soon as possible.

🕒 About 25 minutes, including the practice questions online

What you will learn

1. State the purpose, timing and attendees of the Retrospective.
2. Name what the team inspects.
3. Turn discussion into a small number of real improvements.
4. Explain why improvements can go into the next Sprint Backlog.
5. Keep the Retrospective safe and blame-free.

Why it matters on the exam

Expect questions on who attends, what is inspected (including the Definition of Done), skipping Retrospectives, and how improvements are made real.

Simple explanation

The Retrospective is the last event of the Sprint, timeboxed to a maximum of three hours for a one-month Sprint (usually shorter for shorter Sprints). The whole Scrum Team attends.

The team inspects how the last Sprint went: individuals, interactions, processes, tools and its Definition of Done. It discusses what went well, what problems it met, and how those problems were or were not solved. It identifies the most helpful changes to improve its effectiveness.

"The most impactful improvements are addressed as soon as possible. They may even be added to the Sprint Backlog for the next Sprint."

The Scrum Guide, 2020, Sprint Retrospective

See it in one picture Lesson 15 · Sprint Retrospective

Sprint Review

the product and next steps
Scrum Team + key stakeholders
may adjust the Product Backlog
up to 4 h (1-month Sprint)
second to last

Sprint Retrospective

how the team works
the Scrum Team
people, process, tools, DoD
up to 3 h (1-month Sprint)
concludes the Sprint

Product versus process: the most common confusion between the two events.

Your memory card

1. The Retrospective concludes the Sprint; the whole Scrum Team attends.
2. It inspects individuals, interactions, processes, tools and the DoD.
3. It looks at what went well, what went wrong, and why.
4. Choose the most helpful changes and act soon, even in the next Sprint Backlog.
5. Up to three hours for a one-month Sprint.
6. Safety and openness make it work.

Official source: Scrum Guide 2020: **Sprint Retrospective** at scrumguides.org.

[Open the full lesson →](#)

Deeper professional explanation Lesson 15 · Sprint Retrospective

From complaints to change

A Retrospective that produces ten ideas and no change is a ritual. Choose one or two of the most helpful changes, make them visible (for example in the next Sprint Backlog), and check at the next Retrospective whether they worked.

Find the assumptions

The Guide says assumptions that led the team astray are identified and their origins explored. "We assumed customers wanted speed; they wanted reliability" is far more useful than "the feature was wrong".

The Definition of Done is on the table

The Retrospective is where the team can strengthen its Definition of Done, within any organizational standard. A stronger DoD may change future forecasts.

The whole Scrum Team, safely

The Product Owner is part of the Scrum Team and attends; interactions with the Product Owner are often worth improving. Openness depends on safety. If a manager's presence silences people, find other ways to share impediments with that manager.

Not optional for mature teams

Mature teams still improve. Skipping the Retrospective loses an opportunity to inspect and adapt.

Case study: Retrospectives that work Decide before you read the answer (cover the green part)

Decision 1: The same complaints

Sprint 25

For five Sprints, the Retrospective has listed the same problems: unclear items, late test environments, interruptions. Nothing changes.

What is the best change?

- ☐ A Hold Retrospectives less often until morale improves
- ☒ B Pick the one most helpful change, add it to the next Sprint Backlog, and check it next time
- ☐ C Let the Scrum Master write an improvement plan
- ☐ D Report the problems to management

Best decision: B. The team chooses to refine items with Maya every Wednesday. Two Sprints later, unclear items drop sharply and the team picks the next problem.

Why not the others? A removes improvement. C takes ownership away. D skips the team's own action.

Decision 2: The manager's visit

Sprint 26

A department manager asks to attend Retrospectives "to understand the team's problems". When he attends, nobody speaks openly.

What is the most appropriate response?

- ☐ A Let him attend, since transparency matters
- ☒ B Explain it is the Scrum Team's event, and share impediments with him another way
- ☐ C Hold two Retrospectives, with and without him
- ☐ D Ask him to attend silently

Best decision: B. Sam meets the manager after each Retrospective to share impediments that need his help. Openness returns, and the manager gets better information.

Why not the others? A, C and D keep the safety problem.

Common misunderstandings

- ✗ "Mature teams can skip Retrospectives."

✓ Every Sprint ends with one; there is always something to improve.
- ✗ "The Retrospective is for Developers only."

✓ The whole Scrum Team attends, including the PO.
- ✗ "Improvements happen outside Sprint work."

✓ They may be added to the next Sprint Backlog.
- ✗ "The Retrospective finds who caused problems."

✓ It finds what to improve, including the origins of wrong assumptions.

Exam traps

The trap	How to read it
"The Definition of Done is reviewed at the Sprint Review."	The Retrospective inspects the DoD.
"The Retrospective timebox is four hours."	Three hours for a one-month Sprint; four is the Review.
"Stakeholders attend the Retrospective."	It is the Scrum Team's event.

Exam-style questions 12 questions in the online lesson

1. What is the purpose of the Sprint Retrospective?

A Evaluate how well the Product Owner performed this Sprint

B Plan ways to increase quality and effectiveness

C Reorder the Product Backlog

D Demonstrate the Increment
2. What may the team inspect in the Retrospective?

A Its Definition of Done

B The release budget

C Competitors' products

D Stakeholder satisfaction scores
3. Ten ideas come up every Retrospective, and none are done. What helps most?

A Hold it less often

B Have the Scrum Master assign an owner to each idea

C Pick one or two and add them to the next Sprint

D Add even more ideas
4. A team strengthens its DoD in the Retrospective. What should it expect?

A Stakeholders must approve it

B Sprints must lengthen

C Past Increments become invalid

D Forecasts may need to change

Answers

1: B Better quality and effectiveness. · 2: A DoD is in scope. · 3: C The most helpful changes are addressed as soon as possible. · 4: D Forecasts depend on the DoD.

All 12 questions with full explanations →

Free PSM I questions →

Artifacts and commitments

Product Backlog, Sprint Backlog, Increment and the commitment of each.

Lesson 16 Product Backlog, refinement and Product Goal

Lesson 17 Sprint Backlog and Sprint Goal

Lesson 18 Increment, release and Definition of Done



LESSON 16 OF 26 · PART E · ARTIFACTS AND COMMITMENTS

Product Backlog, refinement and Product Goal

The Product Backlog is an **emergent, ordered list** of what is needed to improve the product, and the **single source of work** for the Scrum Team. **Refinement** keeps items ready. The **Product Goal** is its commitment: the long-term objective the team plans against.

🕒 About 35 minutes, including the practice questions online

What you will learn

1. Describe the Product Backlog and why it is a single source of work.
2. Explain refinement as an ongoing activity, and who sizes items.
3. Define when an item is ready for selection.
4. Write a Product Goal that describes a future state of the product.
5. Apply the rule of one Product Goal at a time.

Why it matters on the exam

Questions test the single source of work, refinement as an activity (not an event, no 10% rule), Developers sizing, and what a Product Goal is.

Simple explanation

The Product Backlog is an emergent, ordered list of what is needed to improve the product. It is the single source of work undertaken by the Scrum Team. The Product Owner is accountable for it.

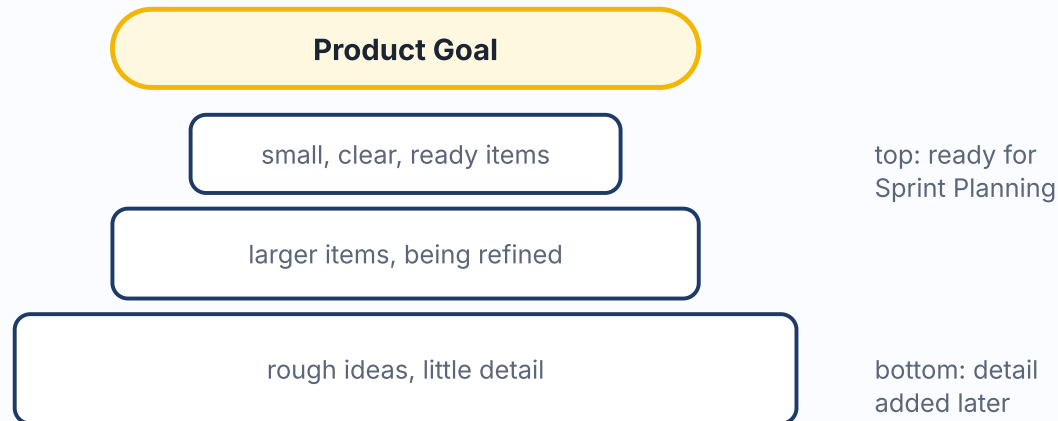
Refinement is the act of breaking down and further defining items into smaller, more precise items. It is an ongoing activity that adds details such as a description, order and size. Items that can be Done by the Scrum Team within one Sprint are ready for selection in Sprint Planning.

The **Product Goal** describes a future state of the product. It is in the Product Backlog, and the rest of the backlog emerges to define what will fulfil it. It is the long-term objective for the Scrum Team; they must fulfil or abandon one objective before taking on the next.

"The Developers who will be doing the work are responsible for the sizing. The Product Owner may influence the Developers by helping them understand and select trade-offs."

The Scrum Guide, 2020, Product Backlog

See it in one picture Lesson 16 · Product Backlog, refinement and Product Goal



ordered by the Product Owner · sized by the Developers · refined continuously

This shape is common practice, not a rule in the Guide.

Your memory card

1. The Product Backlog is an emergent, ordered list and the single source of work.
2. Refinement is an ongoing activity; there is no timebox or percentage.
3. Developers size items; the PO may help with trade-offs.
4. Ready means it can be Done within one Sprint.
5. The Product Goal is a long-term future state, in the Product Backlog.
6. One Product Goal at a time: fulfil or abandon before the next.

Official source: Scrum Guide 2020: **Product Backlog** and **Commitment: Product Goal** at scrumguides.org.

[Open the full lesson →](#)

Deeper professional explanation

Lesson 16 · Product Backlog, refinement and Product Goal

One list, one order

Separate lists for defects, technical work, sales requests or compliance hide value trade-offs. Everything the team works on belongs in the one Product Backlog, ordered by the Product Owner.

Refinement is not an event

The 2020 Guide sets no timebox and no percentage for refinement; the old "10% of capacity" rule is gone. Many teams hold regular refinement sessions as a practice. What matters is that top items are ready when Sprint Planning arrives.

Detail where it is needed

Items near the top are small and well understood; items further down can stay rough. Detailing work two years ahead is often waste, because needs will change. A huge backlog with hundreds of equal-priority items is hard to keep transparent.

What a product is

The Guide describes a product as a vehicle to deliver value, with a clear boundary, known stakeholders, and well-defined users or customers. It could be a service, a physical product or something more abstract.

A good Product Goal

"Deliver features 1 to 40 by June" is a list and a date. "Patients can book and change appointments online without calling" is a future state the team can plan against and measure progress toward.

Example: an equipment rental company Its Product Goal is "Customers can book, track and return equipment without phoning the depot". When a pilot shows customers mostly want real-time tracking, the Product Owner moves tracking items up and drops a planned loyalty feature.

Case study: Cleaning up the backlog Decide before you read the answer (cover the green part)

Decision 1: Two lists

Month 9

FreshCart keeps a feature backlog in one tool and a separate defect list owned by support. Developers pick from both, and nobody knows what matters most.

What is the best fix?

- ☐ A Keep both lists, synchronized weekly
- ☒ B Merge them into one Product Backlog ordered by the PO
- ☐ C Let support order the defects separately
- ☐ D Let the Developers alternate between lists

Best decision: B. Maya merges the lists. Two old defects turn out to matter more than a planned feature, and are fixed first.

Why not the others? A, C and D keep two sources of work.

Decision 2: The pivot

Month 10

A competitor's move makes FreshCart's Product Goal, "Become the fastest grocery delivery in the city", unrealistic. The team proposes a new goal focused on scheduled weekly deliveries.

What does the Guide allow?

- ☐ A Finish the old goal first, then change
- ☒ B Abandon the old goal and take on the new one
- ☐ C Pursue both goals in parallel
- ☐ D Let each Developer choose a goal

Best decision: B. Maya abandons the old goal and sets the new one. The backlog is reordered around scheduled deliveries.

Why not the others? A ignores the evidence. C and D break one objective at a time.

Common misunderstandings

- ✗ "Refinement is a Scrum event."

✓ It is an ongoing activity.
- ✗ "Refinement takes 10% of capacity."

✓ That came from older Guides; the 2020 Guide sets no percentage.
- ✗ "The Product Owner sizes items."

✓ The Developers who will do the work size them.
- ✗ "A team can pursue two Product Goals at once."

✓ One objective at a time: fulfil or abandon before the next.

Exam traps

The trap	How to read it
"Defects live in a separate list."	The Product Backlog is the single source of work.
"Items are ready once estimated in story points."	Ready means it can be Done within one Sprint.
"The Product Goal is finalized at each Sprint Planning."	That describes the Sprint Goal; the Product Goal is long-term.

Exam-style questions

12 questions in the online lesson

1. What is the Product Backlog?
- A A list of defects only

B An emergent, ordered list of what the product needs

C The task list for one Sprint

D A fixed list of requirements agreed with stakeholders at the start
2. Who sizes Product Backlog items?
- A The Developers who will do the work

B The Scrum Master

C The Product Owner, who knows the value

D Stakeholders
3. When is an item ready for Sprint Planning?
- A When the key stakeholders have formally approved it

B When written as a user story

C When it can be Done within one Sprint

D When it has story points
4. A PO overwrites the Developers' sizes to fit a plan. What is the issue?
- A Sizes need stakeholder approval

B The SM must arbitrate

C The PO must size everything to match the plan

D Sizing belongs to the Developers

Answers 1: B Product Backlog section. · 2: A Product Backlog section. · 3: C Product Backlog section. · 4: D Product Backlog section.

All 12 questions with full explanations →

Free PSM I questions →

LESSON 17 OF 26 · PART E · ARTIFACTS AND COMMITMENTS

Sprint Backlog and Sprint Goal

The Sprint Backlog is a **plan by and for the Developers**: the Sprint Goal (why), the selected items (what) and a plan (how). The **Sprint Goal is the commitment**; it stays fixed while the **exact work can change**.

⌚ About 30 minutes, including the practice questions online

What you will learn

1. Describe the three parts of the Sprint Backlog.
2. Explain who owns and updates the Sprint Backlog.
3. Explain the Sprint Goal as the commitment that gives focus and flexibility.
4. Handle work that turns out different than expected.
5. Separate the Sprint Goal from the Product Goal.

Why it matters on the exam

Questions test ownership (Developers), timing (updated throughout the Sprint), and the difference between committing to a goal and guaranteeing every item.

Simple explanation

The Sprint Backlog is composed of the Sprint Goal (why), the Product Backlog items selected for the Sprint (what), and an actionable plan for delivering the Increment (how). It is a highly visible, real-time picture of the work the Developers plan to do, and it is updated throughout the Sprint as more is learned. It should have enough detail that the Developers can inspect their progress at the Daily Scrum.

The Sprint Goal is the single objective for the Sprint. It is a commitment by the Developers, created during Sprint Planning and then added to the Sprint Backlog. It creates coherence and focus, encouraging the Scrum Team to work together rather than on separate initiatives.

"Although the Sprint Goal is a commitment by the Developers, it provides flexibility in terms of the exact work needed to achieve it."

The Scrum Guide, 2020, Commitment: Sprint Goal

"If the work turns out to be different than they expected, they collaborate with the Product Owner to negotiate the scope of the Sprint Backlog within the Sprint without affecting the Sprint Goal."

The Scrum Guide, 2020, Commitment: Sprint Goal

See it in one picture Lesson 17 · Sprint Backlog and Sprint Goal

Why

Sprint Goal
fixed for the Sprint

What

selected items
changes agreed with PO

How

the Developers' plan
changes any time

By and for the Developers · highly visible · updated throughout the Sprint

The further right, the more freely it changes.

Your memory card

1. The Sprint Backlog: Sprint Goal (why), selected items (what), plan (how).
2. A plan by and for the Developers, highly visible, updated throughout the Sprint.
3. The Sprint Goal is the Developers' commitment and the Sprint's single objective.
4. The goal is fixed; the exact work is flexible.
5. Scope changes are negotiated with the Product Owner.
6. Using the Sprint Backlog to police people destroys its transparency.

Official source: Scrum Guide 2020: **Sprint Backlog** and **Commitment: Sprint Goal** at scrumguides.org.

[Open the full lesson →](#)

Deeper professional explanation Lesson 17 · Sprint Backlog and Sprint Goal

Goal fixed, work flexible

The selected items are a forecast, not a promise. If the team learns a better path to the goal, the plan changes. If items must be added or removed, the Developers agree it with the Product Owner. A Sprint that meets its goal but leaves two items unfinished is a success; the items return to the Product Backlog.

Who changes what

The Developers change their plan whenever they learn something: tasks, order, who does what. Changing which items are in the Sprint is a scope change negotiated with the Product Owner. Nobody else, not a manager, a stakeholder or the Product Owner alone, edits the Developers' plan.

Visible, not policed

The Sprint Backlog should be highly visible. Using it to measure individual hours or rank people turns it into a control tool, and people start hiding the real state, which destroys the transparency it exists for.

Sprint Goal versus Product Goal

The Product Goal is the long-term objective, in the Product Backlog. The Sprint Goal is the single objective of one Sprint, in the Sprint Backlog. Each Sprint Goal should move the product toward the Product Goal.

Case study: When the plan meets reality Decide before you read the answer (cover the green part)

Decision 1: A better path

Sprint 27, day 4

The Sprint Goal is "Customers can pay with saved cards". The Developers find a library that removes the need for two selected items, but requires one new item.

What should they do?

- ☐ A Keep the original items, since they were selected in Planning
- ☒ B Agree the swap with Maya and update the Sprint Backlog
- ☐ C Make the swap quietly, since the plan is theirs
- ☐ D Wait for the next Sprint to use the library

Best decision: B. Maya agrees in five minutes. The goal is reached two days early, and the team uses the time on a small valuable item she adds.

Why not the others? A ignores learning. C changes scope without the PO. D delays a better approach.

Decision 2: The manager's spreadsheet

Sprint 28

A manager starts exporting the Sprint Backlog daily to compare how many tasks each Developer closes.

What is the best response from the Scrum Master?

- ☐ A Hide the Sprint Backlog from the manager
- ☒ B Explain the risk to transparency, and offer progress toward the Sprint Goal instead
- ☐ C Let it continue, since the data is visible anyway
- ☐ D Ask Developers to close tasks faster

Best decision: B. The manager switches to watching Sprint Goal progress and the Increment at Reviews. Developers stop splitting tasks artificially to look busy.

Why not the others? A destroys transparency. C lets misuse continue. D rewards gaming.

Common misunderstandings

✗ **"The Sprint Backlog is frozen after Planning."**
✓ It is updated throughout the Sprint as more is learned.

✗ **"The commitment is to finish every selected item."**
✓ The commitment is the Sprint Goal; items are a forecast.

✗ **"The Product Owner owns the Sprint Backlog."**
✓ It is a plan by and for the Developers.

✗ **"A Sprint that leaves items unfinished failed."**
✓ If the Sprint Goal was met, it succeeded.

Exam traps

The trap	How to read it
"The Product Owner may add items to the Sprint Backlog alone."	Scope changes are negotiated with the Developers, protecting the goal.
"The Sprint Goal may change daily."	The plan adapts daily; the goal is the Sprint's single objective.
"Sprint Goal and Product Goal are the same at different times."	One Sprint vs long-term; different artifacts.

Exam-style questions 12 questions in the online lesson

1. Which three parts make up the Sprint Backlog?

A Sprint Goal, selected items, and a plan

B Tasks, estimates, and owners

C Increment, Review notes, and Retro actions

D Product Goal, Product Backlog, and DoD
2. When is the Sprint Backlog updated?

A Only during Sprint Planning

B Only at the Daily Scrum each morning, after the team meets

C Only with the PO's approval

D Throughout the Sprint as more is learned
3. A Developer adds a refactoring task that supports the goal. Whose approval is needed?

A The Scrum Master's

B Nobody's, if scope is unchanged

C Management's

D The Product Owner's, since scope is involved
4. Developers find a better path that drops two items but keeps the goal. What is best?

A Change scope quietly

B Wait for the next Planning

C Agree the scope change with the PO

D Keep the originally selected items regardless of what was learned

Answers 1: A Sprint Backlog section. · 2: D Sprint Backlog section. · 3: B The plan belongs to the Developers. · 4: C Scope is negotiated with the PO.

All 12 questions with full explanations →

Free PSM I questions →

LESSON 18 OF 26 · PART E · ARTIFACTS AND COMMITMENTS

Increment, release and Definition of Done

An Increment is a **usable step toward the Product Goal**. It is born the moment an item **meets the Definition of Done**. Increments may be **released at any time**, even before the Sprint ends; usable and released are different things.

🕒 About 35 minutes, including the practice questions online

What you will learn

1. Define the Increment and when it is created.
2. Explain the Definition of Done and who creates and follows it.
3. Separate Increment from release, and the DoD from acceptance criteria.
4. Handle undone work correctly.
5. Apply the rules for organizational and multi-team Definitions of Done.

Why it matters on the exam

Expect questions on undone work at the Review, early releases, organizational DoD as a minimum, shared DoD across teams, and the difference between DoD and acceptance criteria.

Simple explanation

An Increment is a concrete stepping stone toward the Product Goal. Each Increment is additive to all prior Increments and thoroughly verified, so all Increments work together. To provide value, it must be usable. Multiple Increments may be created within a Sprint, and an Increment may be delivered to stakeholders before the Sprint ends.

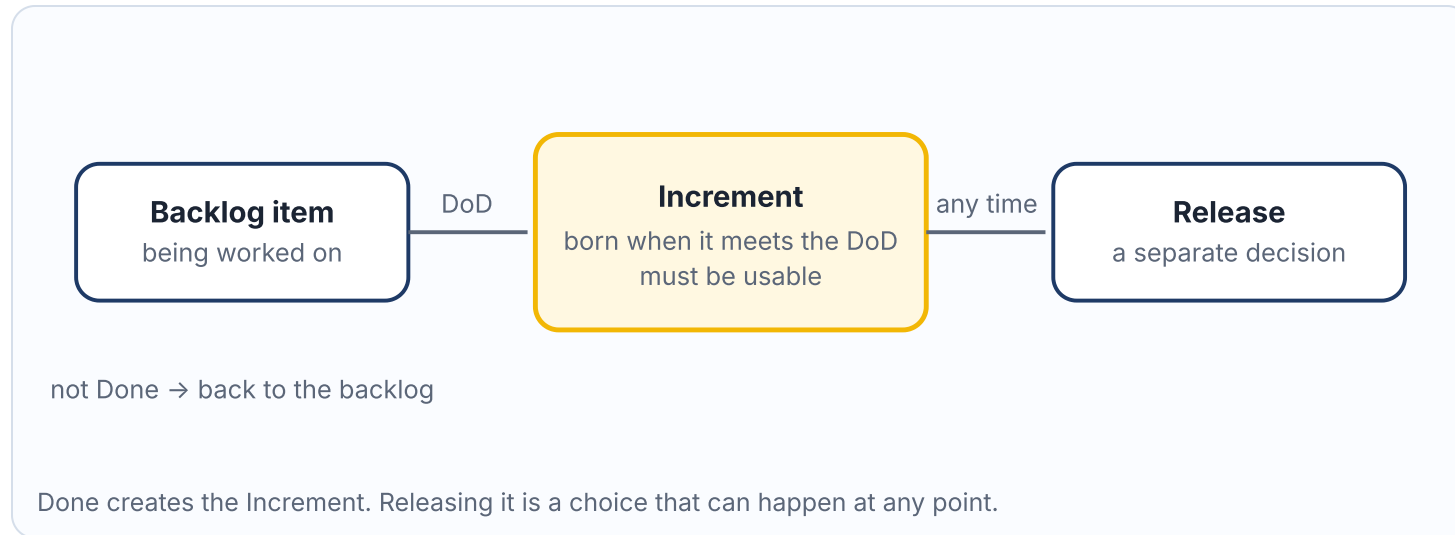
The Definition of Done is a formal description of the state of the Increment when it meets the quality measures required for the product. The moment a Product Backlog item meets it, an Increment is born.

If an organizational Definition of Done exists, all Scrum Teams must follow it as a minimum. If not, the Scrum Team must create one appropriate for the product. The Developers are required to conform to it.

"If a Product Backlog item does not meet the Definition of Done, it cannot be released or even presented at the Sprint Review. Instead, it returns to the Product Backlog for future consideration."

The Scrum Guide, 2020, Commitment: Definition of Done

See it in one picture Lesson 18 · Increment, release and Definition of Done



Your memory card

1. An Increment is a usable, verified step toward the Product Goal.
2. It is born when an item meets the Definition of Done; many per Sprint are possible.
3. Release is a separate decision; the Review is never a gate.
4. An organizational DoD is a minimum; one product means one shared DoD.
5. Undone work is not presented and returns to the Product Backlog.
6. Acceptance criteria are optional and item-specific; the DoD covers all work.

Official source: Scrum Guide 2020: **Increment** and **Commitment: Definition of Done** at scrumguides.org.

[Open the full lesson →](#)

Deeper professional explanation Lesson 18 · Increment, release and Definition of Done

Usable is not the same as released

Every Increment must be usable. Whether and when to release it is a separate business decision: a team may release several times a Sprint, or hold an Increment for a marketing launch. The Sprint Review is never a gate to releasing value.

Definition of Done versus acceptance criteria

Acceptance criteria describe what one specific item must do. They are a common practice, not in the Guide. The Definition of Done applies to all work and describes the quality the Increment must have. An item needs both to be truly finished, but only the DoD is a Scrum commitment.

Organizational and shared Definitions of Done

An organizational DoD is a floor: teams may add stricter criteria but not remove any. When several Scrum Teams work on one product, they must mutually define and comply with the same Definition of Done, otherwise their work may not integrate.

Undone work

Work that does not meet the DoD is not part of the Increment, is not presented at the Review, and returns to the Product Backlog. "90% done" is not done. A separate QA phase after the Sprint usually signals that the DoD is too weak.

Example: a medical-device software team The company's DoD requires code review and unit tests. Because the product is regulated, the team adds traceability to requirements and a usability check. Their Increments are released quarterly, after regulatory approval, but every Sprint's Increment is usable and Done.

Case study: What counts as Done Decide before you read the answer (cover the green part)

Decision 1: 90% done

Last day of Sprint 29

A feature works, but performance testing required by the Definition of Done is not finished. Mr. Grant wants it counted in the Increment and shown at the Review.

What is correct?

- ☐ A Count it, since the remaining work is small
- ☒ B It is not part of the Increment; it returns to the Product Backlog
- ☐ C Show it as a partial Increment
- ☐ D Count it if Maya agrees

Best decision: B. The Review shows only Done work. The feature is finished early next Sprint and released mid-Sprint, without waiting for the Review.

Why not the others? A, C and D all treat undone work as Done.

Decision 2: Two teams, two Dones

Month 12

FreshCart now has two Scrum Teams on the app. One team's Done includes automated tests; the other's does not. Integration breaks every Sprint.

What does the Guide require?

- ☐ A Each team keeps its own DoD
- ☒ B Both teams mutually define and follow one DoD
- ☐ C The stricter team's DoD applies automatically
- ☐ D Maya writes the DoD for both teams

Best decision: B. The Developers of both teams agree a shared DoD that includes integration tests. Breakages stop within two Sprints.

Why not the others? A causes the problem. C and D are not the Guide's process.

Common misunderstandings

- ✗ "Increments are created only at the end of the Sprint."

✓ An Increment is born when an item meets the DoD; several per Sprint are possible.
- ✗ "Every Increment must be released."

✓ It must be usable; release is a separate decision.
- ✗ "Acceptance criteria are the Definition of Done."

✓ Criteria are item-specific (optional); the DoD covers all work.
- ✗ "An organizational DoD cannot be extended."

✓ It is a minimum; teams may add stricter criteria.

Exam traps

The trap	How to read it
"Undone work may be shown at the Review if labelled."	It cannot be presented as part of the Increment.
"Releases happen only after the Sprint Review."	Increments may be delivered before the Sprint ends.
"Each team on one product keeps its own DoD."	They mutually define and comply with one DoD.

Exam-style questions 12 questions in the online lesson

1. When is an Increment born?

A When an item meets the DoD

B When the Product Owner formally accepts it

C When it is released

D At the Sprint Review
2. An item is 90% finished at Sprint end. What happens to it?

A It counts as part of the Increment

B It is shown as a partial Increment

C It is released with a note

D It returns to the Product Backlog
3. May an Increment be released on day 4 of a 10-day Sprint?

A Only in the last Sprint

B Yes, at any time

C Only with SM approval

D No, only after the Review
4. Two teams on one product use different DoDs and integration keeps failing. What does the Guide require?

A The stricter DoD applies automatically

B The PO writes one DoD for both

C They define and follow one shared DoD

D Each keeps its own DoD

Answers 1: A Definition of Done section. · 2: D Not Done = not in the Increment. · 3: B Release when valuable. · 4: C Definition of Done section.

All 12 questions with full explanations →

Free PSM I questions →

Advanced practice

Value, quality, many teams, the organization and Scrum outside software.

Lesson 19 Product value, stakeholders and forecasting

Lesson 20 Quality and technical debt

Lesson 21 Many teams, one product

Lesson 22 Scrum in the organization

Lesson 23 Scrum beyond software

Lesson 24 One full Sprint, start to finish



LESSON 19 OF 26 · PART F · ADVANCED PRACTICE

Product value, stakeholders and forecasting

Scrum aims at **value**, not output. The Product Owner maximizes value by **learning from real results** and stakeholders, and the team **forecasts from what has already happened**, never from hope.

🕒 About 30 minutes, including the practice questions online

What you will learn

1. Explain the difference between output and value.
2. Use Sprint Reviews and evidence to adjust direction.
3. Forecast releases empirically and communicate uncertainty honestly.
4. Label forecasting tools as optional practices.
5. Collaborate with stakeholders without giving away the Product Owner's accountability.

Why it matters on the exam

Questions contrast delivering every requested feature with maximizing value, and fixed-date promises with empirical forecasts.

Simple explanation

A team can finish every planned item and still deliver little value, if customers do not use what was built. Value is the benefit the product delivers. The Product Owner is accountable for maximizing it.

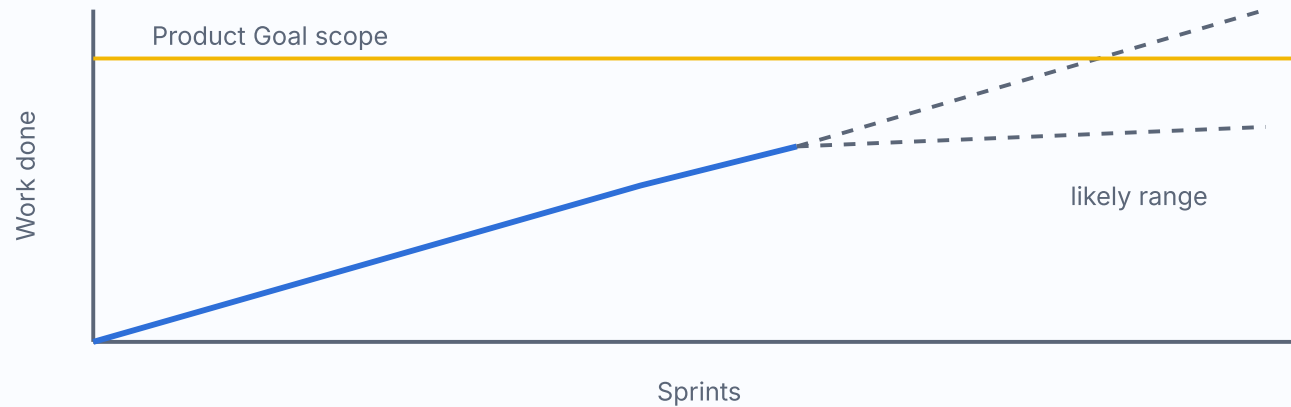
Stakeholders are anyone outside the Scrum Team with an interest in the product. They influence the Product Backlog by working with the Product Owner, especially at the Sprint Review.

Forecasts in Scrum are based on evidence. The Guide says that in complex environments, only what has already happened may be used for forward-looking decisions.

"Various practices exist to forecast progress, like burn-downs, burn-ups, or cumulative flows. While proven useful, these do not replace the importance of empiricism."

The Scrum Guide, 2020, The Sprint

See it in one picture Lesson 19 · Product value, stakeholders and forecasting



A burn-up shows a range from past results. The range narrows as evidence grows.

Your memory card

1. Output is what the team builds; value is the benefit it creates.
2. The PO maximizes value by ordering the backlog using evidence.
3. Stakeholders influence mainly through the Sprint Review and the PO.
4. Forecasts use what has already happened and are updated every Sprint.
5. Burn-downs and similar tools are optional and never replace empiricism.
6. The Product Goal is a long-term objective, not a fixed-scope contract.

Official source: Scrum Guide 2020: **Product Owner**, **The Sprint** and **Sprint Review** at scrumguides.org.

[Open the full lesson →](#)

Deeper professional explanation Lesson 19 · Product value, stakeholders and forecasting

Output versus value

Output is what the team produces; value is what it changes for users and the organization. Two warning signs: a backlog that never changes after Reviews, and success measured only by the number of items completed.

Evidence for value

Useful evidence includes usage data, customer feedback, support requests, revenue or cost effects. The Guide does not prescribe metrics; it requires decisions based on what is observed.

Forecasting honestly

Burn-up charts and similar tools can show likely ranges based on past Sprints. A good forecast is a range that updates every Sprint, not a fixed date promised months ahead. Product Goals are long-term objectives, not fixed-scope contracts.

Stakeholder collaboration

The Scrum Team is responsible for stakeholder collaboration as a product activity; the Scrum Master facilitates it when needed; the Product Owner decides the order. Stakeholders who go around the Product Owner, or committees that try to order the backlog, undermine this.

Example: a public library app The team planned a reading-challenge feature. Sprint Review data showed most users only renewed loans. The Product Owner moved renewal improvements up; visits to branches for renewals fell sharply within two months.

Case study: Busy or valuable? Decide before you read the answer (cover the green part)

Decision 1: Every feature delivered

Month 14

FreshCart's team delivered every planned feature for six months. Usage data shows most of them are rarely used.

What should Maya conclude?

- ☐ A The plan worked, since everything was delivered
- ☒ B Output is not value; use the evidence to reorder the backlog
- ☐ C The Developers built features badly
- ☐ D Usage data is irrelevant to Scrum

Best decision: B. Maya interviews heavy users, drops four planned features and invests in faster checkout. Weekly orders rise.

Why not the others? A confuses output with value. C blames without evidence. D ignores empiricism.

Decision 2: "Give me a date"

Month 15

The board demands a fixed date for the whole Product Goal. The team has six Sprints of results.

What is the best response?

- ☐ A Promise a date with a 30% buffer
- ☒ B Give a forecast range from recent results, updated every Sprint
- ☐ C Refuse to forecast at all
- ☐ D Let the Scrum Master guarantee a date

Best decision: B. Maya shows a likely range of five to seven Sprints and updates it each Review. The board trusts the honest range more than past fixed dates.

Why not the others? A pads a guess. C ignores a reasonable need. D is not a Scrum accountability.

Common misunderstandings

- ✗ "Delivering everything requested maximizes value."

✓ Value comes from what users benefit from, learned through evidence.
- ✗ "Good teams give fixed dates months ahead."

✓ In complex work, forecasts are ranges based on what has happened.
- ✗ "Burn-down charts are required."

✓ They are optional practices that do not replace empiricism.
- ✗ "Stakeholders order the backlog together."

✓ They influence; the Product Owner decides.

Exam traps

The trap	How to read it
"Velocity proves value."	Velocity measures output at best; value is about benefit.
"Add a buffer to make the forecast reliable."	A buffer pads a guess; evidence improves forecasts.
"The Product Goal is a fixed-date contract."	It is a long-term objective guiding the team.

Exam-style questions

12 questions in the online lesson

1. What does the Guide say about burn-down charts?
- A Required only for large products

B Required for every Scrum Team to report progress

C Useful, but no replacement for empiricism

D Forbidden in Scrum
2. A team delivered every planned item, but customers rarely use them. What is most accurate?
- A Output was high; value may be low

B The Developers must have built the features badly

C The team maximized value

D Usage does not matter in Scrum
3. Which basis for a release forecast best fits the Guide?
- A The date sales promised

B A large safety buffer on estimates

C The team's actual recent results

D An expert's best estimate
4. The board demands a fixed date for the whole Product Goal. What is the best response?
- A A refusal to forecast anything

B A guarantee from the Scrum Master

C A fixed date with a 30% buffer added to every estimate

D A range from recent results, updated each Sprint

Answers

1: C Tools help, evidence decides. · 2: A Value is benefit, not output. · 3: C Only what has happened informs forward decisions. · 4: D Honest ranges build trust.

All 12 questions with full explanations →

Free PSM I questions →

LESSON 20 OF 26 · PART F · ADVANCED PRACTICE

Quality and technical debt

In Scrum, **quality is built in every Sprint** through the Definition of Done. Shortcuts that lower quality create **hidden debt** that slows every future Sprint. Quality does not decrease to meet a date.

⌚ About 25 minutes, including the practice questions online

What you will learn

1. Explain why quality does not decrease during a Sprint.
2. Describe technical debt and how it hides undone work.
3. Make quality work visible in the Product Backlog.
4. Use the Retrospective to strengthen the Definition of Done.
5. Respond to deadline pressure without cutting quality.

Why it matters on the exam

"Technical debt" is not a Guide term, but exam questions test the underlying rules: quality does not decrease, undone work is not in the Increment, and all work is in the Product Backlog.

Simple explanation

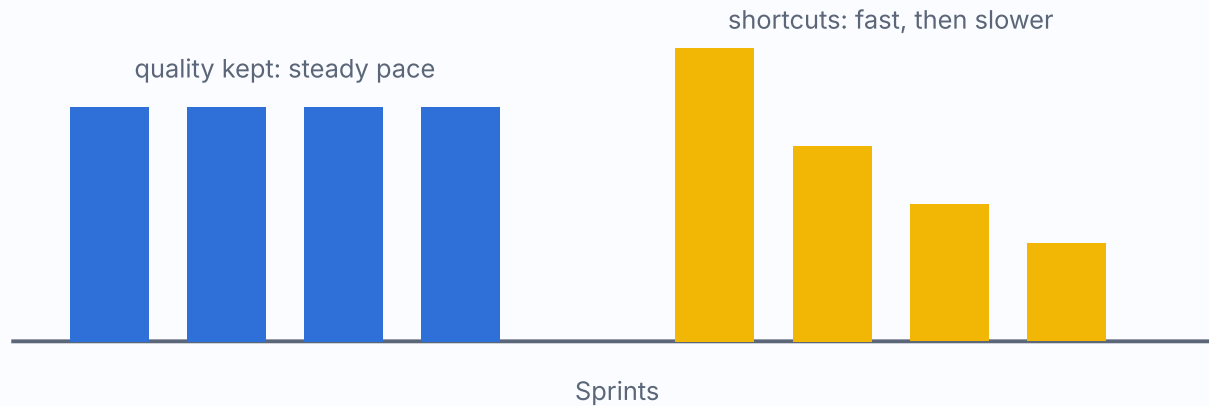
The Guide says quality does not decrease during the Sprint. The Developers instill quality by adhering to the Definition of Done, and work that does not meet it is not part of the Increment.

Technical debt is a common name for shortcuts that make future work harder: missing tests, rushed design, copy-pasted code. It is not a Guide term, but it describes what happens when quality is traded for speed.

"Quality does not decrease."

The Scrum Guide, 2020, The Sprint

See it in one picture Lesson 20 · Quality and technical debt



An illustration of a common pattern, not data from the Guide.

Your memory card

1. Quality does not decrease during the Sprint.
2. The Developers build quality in through the Definition of Done.
3. Shortcuts create hidden debt that slows future Sprints and harms transparency.
4. Quality work belongs in the Product Backlog, ordered by the PO.
5. Strengthen the DoD in the Retrospective; forecasts may change.
6. Under pressure, change scope, not quality.

Official source: Scrum Guide 2020: **The Sprint, Developers** and **Definition of Done** at scrumguides.org.

[Open the full lesson →](#)

Deeper professional explanation Lesson 20 · Quality and technical debt

Why shortcuts are expensive

A shortcut makes this Sprint look faster and every later Sprint slower. Worse, it is often invisible: the Increment looks Done, but carries hidden work. That breaks transparency, because stakeholders believe the product is in better shape than it is.

Make quality work visible

Improvements such as refactoring or test automation belong in the Product Backlog, where the Product Owner orders them with input from the Developers about value and risk. Secret side lists hide work and break the single source of work.

Strengthen the Definition of Done over time

The Retrospective inspects the Definition of Done. Teams often begin with a modest DoD and raise it as their skills and tools grow. A stronger DoD may reduce forecasts at first; that is honest, not a failure.

Under deadline pressure

The answer to pressure is scope, not quality: renegotiate what is delivered with the Product Owner, protect the Sprint Goal where possible, and keep the Definition of Done.

Example: a bridge-inspection data system A civil engineering firm's team skipped data-validation checks to meet a deadline. Two months later, wrong load readings reached reports and every Sprint slowed for cleanup. The team added validation to its DoD and paid down the backlog of fixes over three Sprints.

Case study: The deadline shortcut Decide before you read the answer (cover the green part)

Decision 1: Skip the tests?

Sprint 31, day 8

The team can meet the Sprint Goal only by skipping the automated tests its DoD requires. Mr. Grant says: "Just this once."

What is the best course?

- ☐ A Skip the tests once and add them later
- ☒ B Keep the DoD and renegotiate scope with Maya
- ☐ C Extend the Sprint to finish the tests
- ☐ D Mark the tests as optional in the DoD

Best decision: B. One low-priority part of the feature moves to the next Sprint. The rest ships Done, with no hidden debt.

Why not the others? A and D lower quality. C breaks the timebox.

Decision 2: The secret list

Sprint 32

The Developers keep a private list of refactoring tasks and work on it when they have spare time. Maya does not know it exists.

What is the best change?

- ☐ A Keep the list private to avoid debate
- ☒ B Bring the items into the Product Backlog so Maya can order them
- ☐ C Ask the Scrum Master to own the list
- ☐ D Stop all refactoring work

Best decision: B. Maya sees that two refactoring items will speed up the next big feature, and orders them first.

Why not the others? A hides work. C gives it the wrong owner. D lets debt grow.

Common misunderstandings

- ✗ "Quality can be relaxed once to meet a date."

✓ Quality does not decrease during the Sprint; renegotiate scope instead.
- ✗ "Technical work should be kept off the Product Backlog."

✓ All work belongs in the single Product Backlog.
- ✗ "A strict DoD from day one is required."

✓ Teams improve their DoD over time, inspected in the Retrospective.
- ✗ "Technical debt is a Scrum Guide term."

✓ It is a common practical term, not in the Guide.

Exam traps

The trap	How to read it
"The PO may waive the DoD for urgent work."	The DoD is not waived; scope changes instead.
"A hardening Sprint fixes quality later."	Every Sprint should produce a Done, usable Increment.
"Hidden shortcuts don't affect transparency."	They make the Increment look better than it is.

Exam-style questions 12 questions in the online lesson

1. What does the Guide say about quality during a Sprint?

A

 Quality is checked only at release

B

 Quality does not decrease at all

C

 Quality is set by the PO each Sprint

D

 Quality may drop near deadlines
2. To meet the Sprint Goal, the team must skip tests the DoD requires. What is best?

A

 Keep the DoD; renegotiate scope with the PO

B

 Extend the Sprint by two days

C

 Skip the tests just this once and add them in the next Sprint

D

 Remove the tests from the DoD
3. Developers keep a private refactoring list. What is the main issue?

A

 The Scrum Master should own and maintain the list

B

 It must be done only in Retros

C

 It hides work outside the Product Backlog

D

 Refactoring is not allowed in Scrum
4. A consultant proposes a hardening Sprint every quarter. What is the best response?

A

 Accept it, as long as the hardening Sprint is shorter than a month

B

 Ask the PO to schedule it

C

 Accept it as a normal Scrum event

D

 Each Sprint should yield a Done, usable Increment

Answers 1: B Quality holds. · 2: A Quality does not decrease; scope can move. · 3: C All work, one list. · 4: D Hardening Sprints signal a weak DoD.

All 12 questions with full explanations →

Free PSM I questions →

LESSON 21 OF 26 · PART F · ADVANCED PRACTICE

Many teams, one product

When a product needs several Scrum Teams, they share **one Product Goal, one Product Backlog and one Product Owner**, and they **mutually define and comply with one Definition of Done**. Each team still has its own Developers, Scrum Master and Sprint Backlog.

⌚ About 25 minutes, including the practice questions online

What you will learn

1. State what multiple teams on one product share.
2. Explain why separate backlogs or Product Owners cause problems.
3. Keep Increments integrated across teams.
4. Coordinate across teams without adding hierarchy.
5. Label scaling frameworks as outside the Scrum Guide.

Why it matters on the exam

The Guide's rules for multiple teams are short; questions test them precisely: shared Product Goal, Product Backlog, Product Owner and Definition of Done.

Simple explanation

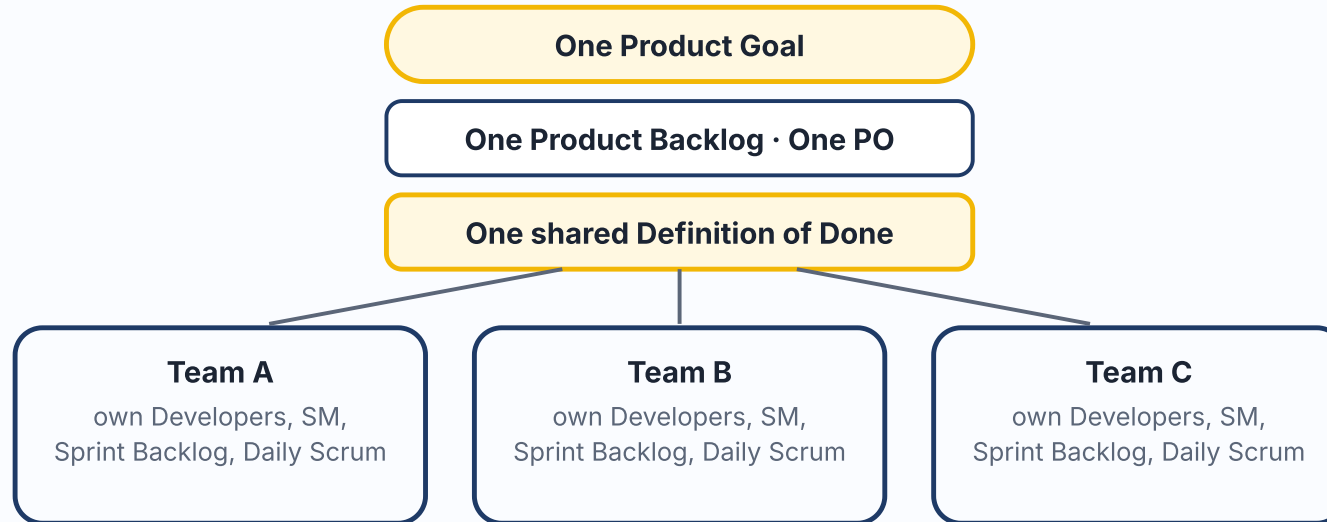
If a Scrum Team becomes too large, the Guide says it should consider reorganizing into multiple cohesive Scrum Teams, each focused on the same product. Those teams share the same Product Goal, Product Backlog and Product Owner.

If multiple Scrum Teams are working together on a product, they must mutually define and comply with the same Definition of Done.

"If Scrum Teams become too large, they should consider reorganizing into multiple cohesive Scrum Teams, each focused on the same product. Therefore, they should share the same Product Goal, Product Backlog, and Product Owner."

The Scrum Guide, 2020, Scrum Team

See it in one picture Lesson 21 · Many teams, one product



Shared on top, separate at team level.

Your memory card

1. Several teams on one product share the Product Goal, Product Backlog and Product Owner.
2. They mutually define and comply with one Definition of Done.
3. Each team keeps its own Developers, Sprint Backlog and Daily Scrum.
4. Developers coordinate directly across teams, without new hierarchy.
5. Increments must work together, verified within the Sprint.
6. Scaling frameworks are outside the Scrum Guide.

Official source: Scrum Guide 2020: **Scrum Team** and **Definition of Done** at scrumguides.org.

[Open the full lesson →](#)

Deeper professional explanation

Lesson 21 · Many teams, one product

Why share so much

Separate backlogs mean competing orders of value and no single view of the product. Separate Product Owners mean conflicting decisions. Separate Definitions of Done mean Increments that do not fit together. Sharing these keeps one product with one direction and one quality standard.

What each team keeps

Each Scrum Team has its own Developers, its own Scrum Master accountability (one person may serve several teams), its own Sprint Backlog and its own Daily Scrum. Teams select work from the shared Product Backlog.

Coordination without hierarchy

Developers from different teams can coordinate directly whenever needed, for example on shared interfaces. Many groups plan in the same session or review the integrated Increment together. Adding a "coordinator" who directs teams creates hierarchy.

Scaling frameworks

Frameworks such as Nexus, LeSS and SAFe offer patterns for many teams. They are not part of the Scrum Guide, and PSM I tests only the Guide's rules.

Case study: FreshCart grows to three teams Decide before you read the answer (cover the green part)

Decision 1: Three backlogs?

Month 16

FreshCart now has three Scrum Teams. To avoid conflicts, each team proposes its own Product Backlog, and two teams want their own Product Owners.

What is the best response?

- ☐ A Allow it, since self-managing teams choose their structure
- ☒ B Keep one Product Goal, Product Backlog and PO, and coordinate openly
- ☐ C Keep one PO but give each team its own backlog
- ☐ D Merge all three teams into one large team

Best decision: B. The teams select from Maya's single backlog and meet briefly before each Sprint Planning to avoid collisions.

Why not the others? A and C split the product. D creates a team too large to work well.

Decision 2: The integration crash

Month 17

Each team's work passes its own tests, but combined, the app crashes before every Sprint Review.

What fits the Guide best?

- ☐ A Add a separate integration team after each Sprint
- ☒ B One shared DoD that includes integration, verified within the Sprint
- ☐ C Integrate only before major releases
- ☐ D Let each team keep its own DoD

Best decision: B. The shared DoD now requires the integrated build to pass. Crashes stop, and the Review shows one working Increment.

Why not the others? A and C delay integration. D keeps the cause.

Common misunderstandings

- ✗ "Each team on a product needs its own Product Owner."

✓ Teams on one product share one Product Owner.
- ✗ "Each team keeps its own Definition of Done."

✓ Teams on one product mutually define and follow one DoD.
- ✗ "All teams must hold one joint Daily Scrum."

✓ Each team holds its own; Developers coordinate as needed.
- ✗ "Scaling frameworks are part of Scrum."

✓ They are outside the Scrum Guide.

Exam traps

The trap	How to read it
"A chief Product Owner coordinates team POs."	One product, one Product Owner.
"Each team needs its own Product Goal."	They share one Product Goal.
"An integration team combines work after the Sprint."	Increments should already work together, verified within the Sprint.

Exam-style questions 12 questions in the online lesson

1. What do multiple Scrum Teams on one product share?

A Sprint Goal and Sprint length

B Product Goal, Product Backlog and PO

C Scrum Master and Developers

D Sprint Backlog, Daily Scrum and Sprint Goal
2. Two teams on one product want separate backlogs to avoid conflicts. What is best?

A Keep one backlog and coordinate openly

B Allow it if the PO approves

C Allow it, to give each team more independence

D Let each Scrum Master decide
3. Does each team keep its own Daily Scrum?

A Only the largest team does

B Only if the PO attends

C Yes, for its own Developers

D No, all teams hold one together
4. A company adds a "chief Product Owner" over three team Product Owners. What is the main issue?

A The SM should be the chief PO

B Three POs need a fourth to decide

C Chief Product Owners are a Scrum requirement for scale

D The product should have one Product Owner

Answers 1: B Goal, backlog, owner. · 2: A One product, one Product Backlog. · 3: C Each team's Developers inspect their own Sprint Goal. · 4: D One owner.

All 12 questions with full explanations →

Free PSM I questions →

LESSON 22 OF 26 · PART F · ADVANCED PRACTICE

Scrum in the organization

Scrum Teams work inside organizations. Scrum makes **visible how well management and processes work**, and the Scrum Master helps the organization **remove barriers**. Leaders support Scrum by **empowering teams** and **respecting the Product Owner**, not by directing daily work.

🕒 About 30 minutes, including the practice questions online

What you will learn

1. Explain how Scrum exposes organizational problems.
2. Handle management interference constructively.
3. Compare a Scrum Master and a traditional project manager fairly.
4. Recognize policies that undermine Scrum, such as individual rankings.
5. Describe the Scrum Master's service to the organization.

Why it matters on the exam

Questions describe managers, PMOs, HR systems or approval chains clashing with Scrum. The best answer usually makes the impact visible and works with the organization to change it.

Simple explanation

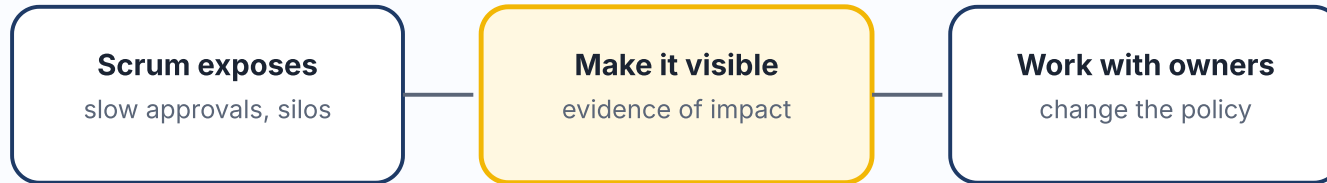
The Guide says Scrum makes visible the relative efficacy of current management, environment and work techniques, so improvements can be made. That can be uncomfortable: slow approvals, conflicting priorities and siloed departments become obvious.

The Scrum Master serves the organization by leading, training and coaching its Scrum adoption; planning and advising implementations; helping people understand an empirical approach; and removing barriers between stakeholders and Scrum Teams.

"For Product Owners to succeed, the entire organization must respect their decisions."

The Scrum Guide, 2020, Product Owner

See it in one picture Lesson 22 · Scrum in the organization



Respect legitimate needs; remove the waste around them

The Scrum Master's path from friction to improvement.

Your memory card

1. Scrum exposes how well management, processes and techniques work.
2. The Scrum Master serves the organization: training, advising, removing barriers.
3. Respect legitimate needs; remove avoidable delay around them.
4. Policies like individual rankings and constant reassignment undermine Scrum.
5. The organization must respect the Product Owner's decisions.
6. Scrum Master and traditional PM roles fit different kinds of work.

Official source: Scrum Guide 2020: **Scrum Definition**, **Scrum Master** and **Product Owner** at scrumguides.org.

[Open the full lesson →](#)

Deeper professional explanation Lesson 22 · Scrum in the organization

Common clashes

Managers assigning tasks to Developers. Approval boards that delay every change. Individual performance rankings that discourage helping. Fixed-scope, fixed-date contracts for complex work. Constant reassignment of people between teams. Each undermines self-management, focus or adaptation.

How to respond

Make the impact transparent with evidence: how long changes wait, what a ranking does to teamwork. Then work with the people who own the policy. Respect legitimate needs, such as safety rules or legal approvals, while removing the delay around them.

Scrum Master and traditional project manager

A traditional project manager often plans and assigns work and reports status, which fits stable, well-understood work. A Scrum Master builds the team's ability to plan and assign its own work and helps the organization adopt Scrum. Neither role is "bad"; they suit different kinds of work.

What managers can do

Managers support Scrum Teams by setting clear objectives and constraints, keeping teams stable, funding tools and training, and removing barriers. They influence the product through the Product Owner.

Example: a government licensing office Every form change needed three signatures and took six weeks. The Scrum Master showed the director how many improvements were waiting. The director delegated approval of minor changes to the Product Owner, and the backlog of waiting changes cleared in two months.

Case study: The organization pushes back Decide before you read the answer (cover the green part)

Decision 1: The PMO report

Month 18

The PMO asks every Scrum Team for a weekly 10-page status report with individual task hours.

What is the best response from the Scrum Master?

- ☐ A Write the report as requested
- ☒ B Understand the PMO's real need and offer transparency through Scrum's artifacts and events
- ☐ C Refuse to share any information
- ☐ D Ask Developers to write it in turns

Best decision: B. The PMO mainly needed release forecasts and risks. It now joins Sprint Reviews and reads a one-page forecast. The 10-page report is dropped.

Why not the others? A wastes effort and invites micromanagement. C is unhelpful. D spreads the waste.

Decision 2: The ranking

Month 19

HR plans to rank Developers by individual tickets closed for bonuses.

What should the Scrum Master do?

- ☐ A Accept it, since HR is outside Scrum
- ☒ B Explain how it undermines teamwork and suggest team-based recognition
- ☐ C Tell Developers to ignore the bonuses
- ☐ D Help HR build the ranking

Best decision: B. HR adds a team-outcome bonus instead. Developers keep helping each other.

Why not the others? A ignores a barrier. C is unrealistic. D reinforces the harm.

Common misunderstandings

- ✗ "A Scrum Master is a project manager with a new title."

✓ The roles differ; the Scrum Master builds team self-management.
- ✗ "Scrum works best if the organization doesn't change."

✓ Scrum exposes organizational problems so they can be improved.
- ✗ "HR systems are none of the Scrum Master's business."

✓ Removing barriers to Scrum is part of serving the organization.
- ✗ "Traditional project management is always wrong."

✓ It fits stable, well-understood work.

Exam traps

The trap	How to read it
"The Scrum Master should write status reports for management."	Offer transparency through artifacts and events instead.
"Remove every approval as an impediment."	Respect legitimate needs; remove avoidable delay.
"Managers decide the backlog order."	The organization must respect the Product Owner's decisions.

Exam-style questions 12 questions in the online lesson

1. According to the Guide, what does Scrum make visible?

A Which managers are performing well or badly

B How effective management and practices are

C The cost of every Scrum event

D Each Developer's individual output
2. A PMO demands a weekly 10-page report with individual hours. What is best?

A Understand the need; offer Scrum's transparency

B Refuse to share information

C Write the report as requested

D Rotate writing the report among the Developers each week
3. HR plans individual rankings by tickets closed. What should the SM do?

A Help design the ranking

B Tell the Developers to simply ignore the bonus scheme

C Explain the harm; suggest team-based recognition

D Accept it as outside Scrum
4. Which organizational practices undermine Scrum? (Choose 2)

A Keeping teams stable

B Delegating minor approvals to the PO

C Moving people between teams often

D Managers assigning daily tasks

Answers 1: B Scrum is a mirror. · 2: A SM serves the organization. · 3: C Systems should reinforce the values. · 4: C and D They erode self-management and focus.

All 12 questions with full explanations →

Free PSM I questions →

LESSON 23 OF 26 · PART F · ADVANCED PRACTICE

Scrum beyond software

Scrum works wherever work is **complex**: hardware, healthcare, construction design, marketing, research, government. The rules stay the same; what changes is **what a usable Increment looks like** and what the Definition of Done must cover.

⌚ About 25 minutes, including the practice questions online

What you will learn

1. Explain why Scrum applies to complex work in any domain.
2. Define a usable Increment for physical or service work.
3. Write a Definition of Done for non-software products.
4. Handle long lead times without breaking the Sprint length.
5. Recognize that "Developers" includes every specialist.

Why it matters on the exam

Questions often set a scenario in a factory, hospital or marketing team to test whether you still apply the Guide's rules exactly.

Simple explanation

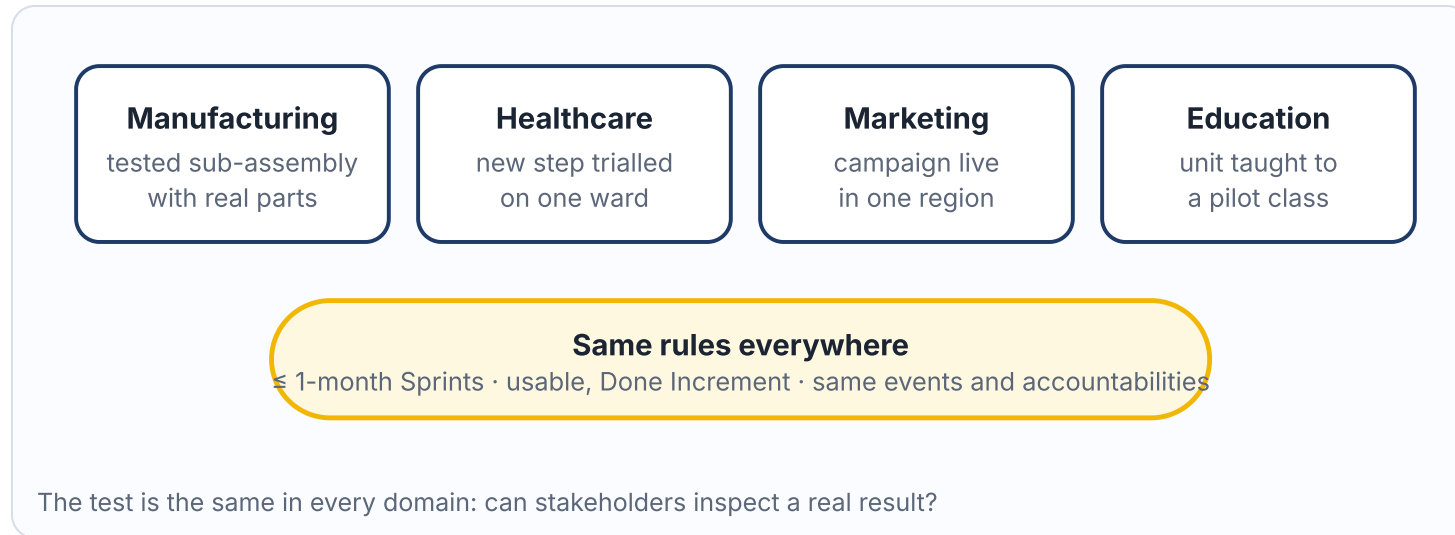
The Guide says Scrum was initially developed for products, and is now used for software, hardware, embedded software, networks of interacting functions, autonomous vehicles, schools, government, marketing, managing the operation of organizations and much more. It uses the word "Developers" to simplify, not to exclude: researchers, analysts, scientists and other specialists are Developers too.

The rules do not change outside software. Sprints are still one month or less, each Sprint still aims for a usable Increment that meets the Definition of Done, and the same accountabilities, events and artifacts apply.

"We use the word "developers" in Scrum not to exclude, but to simplify. If you get value from Scrum, consider yourself included."

The Scrum Guide, 2020, Purpose of the Scrum Guide

See it in one picture Lesson 23 · Scrum beyond software



Your memory card

1. Scrum fits complex work in any domain.
2. "Developers" means everyone who creates the Increment.
3. The rules do not change outside software.
4. A usable Increment is a real, inspectable result.
5. The DoD reflects the quality the product requires.
6. Long lead times call for slicing work, not longer Sprints.

Official source: Scrum Guide 2020: **Purpose of the Scrum Guide** and **Increment** at scrumguides.org.

[Open the full lesson →](#)

Deeper professional explanation Lesson 23 · Scrum beyond software

What "usable" means outside software

A usable Increment is something stakeholders can actually inspect or use: a tested gripper station running real parts, a new patient-intake step trialled on one ward, a campaign landing page live for one region, a verified structural design for one floor. Drawings or reports alone rarely count; they show intent, not a working result.

Definition of Done for physical work

A DoD outside software may include safety checks, measurements against tolerances, regulatory documentation, sign-off tests with real materials, or user trials. Organizational standards, such as safety rules, are a minimum.

Long lead times

Parts that take eight weeks to arrive do not justify eight-week Sprints. Teams slice work so each Sprint still produces something verifiable: simulate, prototype, test sub-assemblies, or order early while building what is possible now.

Cross-functional across disciplines

Handoffs between departments (design, then electrical, then programming, then test) recreate a sequential process inside Sprints. Putting the disciplines in one Scrum Team, working on one goal, surfaces integration problems in weeks instead of at the end.

Example: a school curriculum team Teachers redesigning a science course work in three-week Sprints. Each Sprint, one new unit is taught to a pilot class and student results are inspected at the Sprint Review with the department head and parents' representatives. Weak units are reworked before the full rollout.

Case study: A conveyor manufacturer tries Scrum Decide before you read the answer (cover the green part)

Decision 1: The eight-week motors

Project start

A conveyor manufacturer adopts Scrum. Motors take eight weeks to arrive, so the engineers propose eight-week Sprints.

What is the best approach?

- ☐ A Use eight-week Sprints to match the lead time
- ☒ B Keep Sprints to a month or less and slice work so each Sprint ends with something verifiable
- ☐ C Skip Sprint Reviews until the motors arrive
- ☐ D Run Scrum only for the software part of the machine

Best decision: B. Motors are ordered in Sprint 1. Meanwhile the team builds and tests the frame and controls with a rental motor. When the real motors arrive, integration takes days, not weeks.

Why not the others? A breaks the limit. C removes inspection. D splits the product.

Decision 2: The drawings Increment

Sprint 2 Review

The team proposes to present a complete set of drawings as its Increment for Sprint 2.

What is the most useful view?

- ☐ A Drawings are always a valid Increment
- ☒ B Show something stakeholders can inspect working, such as a tested sub-assembly
- ☐ C Skip the Review until a full machine exists
- ☐ D Let the PO decide alone, without a DoD

Best decision: B. The team shows a sub-assembly running real packages. The customer spots a jam point that the drawings had hidden.

Why not the others? A shows intent, not results. C removes inspection. D ignores the DoD.

Common misunderstandings

- ✗ "Scrum is only for software."

✓ The Guide lists many other domains.
- ✗ "Hardware teams need longer Sprints."

✓ Sprints stay one month or less; slice the work.
- ✗ "Only programmers are Developers."

✓ The word includes all specialists creating the Increment.
- ✗ "Drawings and reports are always good Increments."

✓ They show intent; a usable result is better evidence.

Exam traps

The trap	How to read it
A factory scenario with an extended Sprint.	The one-month limit applies in every domain.
"Non-software teams may skip the Definition of Done."	Every Increment must meet a DoD, whatever the domain.
"Researchers are stakeholders, not Developers."	If they create the Increment, they are Developers.

Exam-style questions 12 questions in the online lesson

1. Which domains does the Guide mention as using Scrum?

A Only products that are sold to outside customers

B Hardware, schools, government and more customers

C Software and IT operations only

D Software only
2. Parts take eight weeks to arrive. A team wants eight-week Sprints. What is best?

A Keep a month or less; slice the work

B Skip Reviews until parts arrive

C Allow eight-week Sprints for hardware teams only

D Use Scrum only for the software
3. Which is the most useful Increment for a machine builder in Sprint 2?

A A detailed project schedule

B A list of purchased components

C A sub-assembly tested with real parts

D A complete, detailed set of design drawings
4. Which make a good Increment for a marketing team? (Choose 2)

A A slide deck describing the campaign

B A plan approved by the director

C A tested email sent to a pilot group

D A campaign page live for one region

Answers 1: B Purpose of the Scrum Guide. · 2: A The limit applies in every domain. · 3: C Usable and inspectable. · 4: C and D Usable and inspectable with real users.

All 12 questions with full explanations →

Free PSM I questions →

LESSON 24 OF 26 · PART F · ADVANCED PRACTICE

One full Sprint, start to finish

This lesson follows **one complete FreshCart Sprint**, from Planning to Retrospective, and asks you to make the key decisions. It ties together everything from the course: **accountabilities, events, artifacts and commitments working as one system**.

⌚ About 40 minutes, including the practice questions online

What you will learn

1. Apply the rules of every event in sequence.
2. Make realistic decisions across a whole Sprint.
3. See how artifacts and commitments connect day by day.
4. Recognize how one bad decision affects later events.
5. Prepare for integrated exam scenarios.

Why it matters on the exam

Many hard PSM I questions combine two or three rules. Working through a whole Sprint builds the habit of seeing how they interact.

Simple explanation

The setting. FreshCart, Sprint 40. Two-week Sprint. Product Goal: "Customers can schedule weekly grocery deliveries without phoning support." Maya is Product Owner, Sam is Scrum Master, and seven Developers are on the team.

- **Day 1, Sprint Planning:** Maya proposes making scheduling self-service. The team agrees the Sprint Goal "Customers can change their delivery day themselves." The Developers select six items and plan the first days.
- **Days 2–9:** Daily Scrums, a mid-Sprint surprise, a stakeholder request and a quality decision.
- **Day 10:** Sprint Review with stakeholders, then the Retrospective.

"The Sprint Goal must be finalized prior to the end of Sprint Planning."

The Scrum Guide, 2020, Sprint Planning

See it in one picture

Lesson 24 · One full Sprint, start to finish



Your memory card

1. The Sprint Goal guides every decision during the Sprint.
2. Transparency early makes adaptation cheap.
3. Scope changes are agreed with the PO; quality never drops.
4. New requests go to the Product Owner and the Product Backlog.
5. The Review uses outcomes to decide next steps.
6. The Retrospective traces causes and picks a concrete improvement.

Official source: Read the full Scrum Guide 2020 once more after this lesson, start to finish.

[Open the full lesson →](#)

Deeper professional explanation

Lesson 24 · One full Sprint, start to finish

Day 4: the surprise

The Developers discover that the delivery-slot service cannot handle changes inside 24 hours. One selected item cannot be finished. The Sprint Goal is still reachable with a simpler rule. They adapt the plan and agree the scope change with Maya the same day.

Day 6: the stakeholder request

The sales director asks a Developer to add a promotional banner "today". It is unrelated to the Sprint Goal. The Developer brings it to Maya, who adds it to the Product Backlog for the next Sprint.

Day 8: the quality decision

Performance tests required by the Definition of Done reveal slow loading. The team considers skipping them; instead they fix the slowest query and keep the DoD. One low-value item moves back to the Product Backlog.

Day 10: Review and Retrospective

At the Review, stakeholders try the feature live. Support reports a 30% drop in calls during the pilot. They suggest letting customers skip a week, and Maya moves that up the backlog. At the Retrospective, the team notices that the day-4 surprise came from an untested assumption about the slot service. They add a one-hour technical check to refinement for risky items.

What connected it all

The Sprint Goal guided every choice. Transparency (the Day 4 problem raised at once) made adaptation possible. The Definition of Done kept quality. The Product Owner kept one ordered backlog. The Scrum Master mostly stayed in the background, facilitating and protecting the process.

Case study: Your decisions in Sprint 40

Decide before you read the answer (cover the green part)

Decision 1: Day 4 surprise

Sprint 40, day 4

One selected item cannot be finished because of a technical limit. The Sprint Goal can still be met with a simpler approach.

What should the Developers do?

- ☐ A Cancel the Sprint because the plan failed
- ☒ B Adapt the plan and agree the scope change with Maya
- ☐ C Keep working on the impossible item
- ☐ D Extend the Sprint by three days

Best decision: B. The goal stays intact, the plan changes, and Maya knows why.

Why not the others? A: the goal is not obsolete, and only the PO may cancel. C wastes effort. D breaks the timebox.

Decision 2: Day 8 quality

Sprint 40, day 8

Performance tests fail. Fixing them may cost one of the remaining items.

What is the best course?

- ☐ A Skip the tests to finish every item
- ☒ B Keep the DoD and move one item back to the Product Backlog
- ☐ C Ask Sam to waive the tests
- ☐ D Release without the tests and fix later

Best decision: B. The Increment is genuinely Done, and the goal is met.

Why not the others? A, C and D lower quality or release undone work.

Common misunderstandings

- ✗ "A surprise mid-Sprint means the Sprint failed."

✓ Adapting to surprises is how Scrum is meant to work.
- ✗ "The Scrum Master runs the Sprint."

✓ The Scrum Master facilitates and protects the process; the team runs the work.
- ✗ "Stakeholder requests go straight into the Sprint."

✓ They go to the Product Owner and the Product Backlog.
- ✗ "Retrospectives only list problems."

✓ They trace causes and choose a concrete improvement.

Exam traps

The trap	How to read it
An integrated scenario with one rule hidden in the details.	Check every rule involved: goal, scope, quality, accountability, timebox.
"Cancel the Sprint because the plan changed."	Only an obsolete goal justifies cancellation.
"Fix the root cause at the Review."	Process causes belong in the Retrospective.

Exam-style questions 12 questions in the online lesson

1. When must the Sprint Goal be finalized?

A Whenever Maya decides

B Before Sprint Planning ends

C Before the Sprint Review

D By the first Daily Scrum of the Sprint
2. On day 4 a selected item proves impossible, but the goal is reachable. What is best?

A Adapt and agree the scope change with the PO

B Keep working on the item regardless

C Cancel the Sprint, because the original plan has failed

D Extend the Sprint by three days
3. On day 6 the sales director asks a Developer for an unrelated banner today. Where should it go?

A To the Scrum Master to approve

B Into a private Developer list

C To the PO and the Product Backlog

D Straight into the current Sprint, as it is quick
4. Which rules guided the day-4 and day-8 decisions? (Choose 2)

A Sprints may be extended when needed

B The SM decides scope changes

C Quality does not decrease

D Scope may be renegotiated with the PO

Answers 1: B Sprint Planning section. · 2: A Scope renegotiation protects the goal. · 3: C Work enters through the PO. · 4: C and D Scope moves; quality holds.

All 12 questions with full explanations →

Free PSM I questions →

Exam readiness

How PSM I questions are written, and the distinctions that decide a pass.

Lesson 25 Reading PSM I questions; 2017 vs 2020

Lesson 26 Final review: the hardest distinctions



LESSON 25 OF 26 · PART G · EXAM READINESS

Reading PSM I questions; 2017 vs 2020

Many wrong answers come from **misreading the question** or from **old study material**. Learn a simple reading method, and know exactly **what changed in the 2020 Scrum Guide**.

🕒 About 30 minutes, including the practice questions online

What you will learn

1. Apply a five-step method to read any PSM I question.
2. Spot invented rules and outdated rules in answer options.
3. Handle "choose 2" and "choose 3" questions safely.
4. Name the main changes from the 2017 to the 2020 Scrum Guide.
5. Use time well across 80 questions.

Why it matters on the exam

This lesson is pure exam technique. It applies to every topic in the course.

Simple explanation

A five-step reading method:

1. **Read the last line first.** Is it asking for the best action, an accurate statement, or something NOT true? How many answers?
2. **Find who is involved.** Product Owner, Scrum Master, Developers, the whole Scrum Team, or someone outside?
3. **Find the rule at stake.** Accountability, event purpose, timebox, artifact ownership, commitment.
4. **Remove invented and outdated options.** Rules the Guide never states, or rules from older versions.
5. **Choose the option closest to the Guide,** at the strength the Guide states it: required, allowed or not mentioned.

See it in one picture Lesson 25 · Reading PSM I questions; 2017 vs 2020

1 Last line

what, how many

2 Who

which accountability

3 Rule

what is tested

4 Remove

invented, outdated

5 Choose

closest to Guide

Practise this on every question until it becomes automatic.

Your memory card

1. Read the last line first: what, and how many.
2. Identify who is involved and which rule is tested.
3. Remove invented and outdated options.
4. Choose the option closest to the Guide, at its stated strength.
5. Know the 2017-to-2020 changes.
6. About 45 seconds per question; never leave blanks.

Official source: Compare the Scrum Guide 2020 with older study material you may own.

[Open the full lesson →](#)

Deeper professional explanation

Lesson 25 · Reading PSM I questions; 2017 vs 2020

Main changes from 2017 to 2020

2017 Guide	2020 Guide
Development Team (a team within the team)	Developers, inside one Scrum Team
Roles	Accountabilities
Self-organizing	Self-managing
3 to 9 Development Team members	Scrum Team typically 10 or fewer
No Product Goal	Product Goal added as a commitment
No named commitments	Product Goal, Sprint Goal, Definition of Done
Sprint Planning: what and how	Sprint Planning adds why (the Sprint Goal)
Three Daily Scrum questions	Developers choose any structure
Refinement usually no more than 10% of capacity	Refinement is an ongoing activity; no percentage
Detailed steps for cancelled Sprints	Only: PO may cancel if the goal is obsolete
Scrum Master as "servant-leader"	"True leaders who serve"

Invented rules

Wrong options often state rules the Guide never makes: "Scrum forbids part-time members", "each Scrum Master may serve only one team", "every Increment must be released", "Sprints must be two weeks". If you cannot recall the rule in the Guide, be suspicious.

Multi-answer questions

Each selection must be right. Pick the options you are sure of first, then judge the rest. Partly right is fully wrong.

Time

About 45 seconds per question. Answer everything, mark doubtful ones, return at the end. A blank answer is always wrong.

Case study: Reading under pressure Decide before you read the answer (cover the green part)

Decision 1: Read the last line

Practice exam, question 12

A long scenario describes a frustrated Product Owner, a busy Scrum Master and delayed testing. The last line asks: "Which TWO actions should the Developers take?"

What should you focus on first?

- ☐ A What the Product Owner should do
- ☒ B What the Developers can do, choosing exactly two
- ☐ C What the Scrum Master is accountable for
- ☐ D Which option is longest

Best approach: B. The question asks about the Developers, and for two answers. Options about the PO or SM may be true but answer a different question.

Why not the others? A and C answer a different question. D is not a reliable signal.

Decision 2: The old rule

Practice exam, question 47

An option says: "The Development Team should have between 3 and 9 members." It sounds familiar.

How should you judge it?

- ☐ A Choose it; it is a well-known rule
- ☒ B Recognize it as a 2017 rule and test the claim against 2020
- ☐ C Ignore all options with old terms
- ☐ D Choose it only if no other option fits

Best approach: B. The 2020 Guide says the Scrum Team is typically 10 or fewer, with no 3-to-9 rule.

Why not the others? A trusts outdated memory. C is too blunt. D still risks an outdated answer.

Common misunderstandings

✗ "Answers with 'always' or 'never' are wrong."

✓ Judge the claim; the Guide uses strong words too.

✗ "The longest answer is usually right."

✓ Length is not a reliable signal.

✗ "Older study guides are still accurate."

✓ Several rules changed in 2020.

✗ "Partial credit exists for multi-answer questions."

✓ Every selection must be right.

Exam traps

The trap	How to read it
"3 to 9 Developers", "10% refinement", "three questions".	2017 rules; not in the 2020 Guide.
"Scrum forbids X" when the Guide is silent.	An invented rule; be suspicious.
Answering about the wrong person.	Re-read who the question asks about.

Exam-style questions

12 questions in the online lesson

1. Which term did the 2020 Guide introduce as a commitment for the Product Backlog?

A Roadmap

B Product Goal

C Product Vision

D Release Goal
2. A question ends "Which TWO actions should the Developers take?" What should you check first?

A Which options are Developer actions

B Which options are longest

C Which options mention the Product Owner

D Which option sounds most agile by name
3. An option says "refinement should take no more than 10% of capacity". How should you judge it?

A It applies only to large teams

B It is a Scrum Master rule

C It is from older Guides, not 2020

D It is a current Scrum rule for all teams
4. Which are 2017 rules no longer in the 2020 Guide? (Choose 2)

A Sprints of one month or less

B One Product Owner per product

C Three Daily Scrum questions

D 3 to 9 Developers

Answers 1: B Product Goal is new. · 2: A Read the last line first. · 3: C 2020 sets no percentage. · 4: C and D Know what was removed.

All 12 questions with full explanations →

Free PSM I questions →

LESSON 26 OF 26 · PART G · EXAM READINESS

Final review: the hardest distinctions

This final lesson reviews the **distinctions that cause most wrong answers**. If you can explain each pair below in one sentence, you are close to ready for the exam.

🕒 About 30 minutes, including the practice questions online

What you will learn

1. Explain each of the key distinctions in one sentence.
2. Recognize them inside realistic scenarios.
3. Identify your weakest distinctions for final review.
4. Connect each distinction to its Scrum Guide section.
5. Decide when you are ready to book the exam.

Why it matters on the exam

These distinctions appear again and again on PSM I in different disguises.

Simple explanation

The key distinctions:

- **Accountable vs responsible:** answer for the result vs do the work.
- **Self-managing vs manager-directed:** the team decides who, what, when, how.
- **Product Goal vs Sprint Goal:** long-term objective vs one Sprint's objective.
- **Sprint Review vs Retrospective:** the product vs the way of working.
- **Increment vs release:** usable and Done vs delivered to users.
- **Refinement vs events:** an ongoing activity vs the five events.
- **Definition of Done vs acceptance criteria:** quality for all work vs item-specific (optional).
- **Scrum Master vs project manager:** enables self-management vs plans and assigns.
- **Empiricism vs predictive planning:** decide from observation vs fix the plan upfront.
- **Commitment vs guarantee:** the Sprint Goal vs every selected item.

See it in one picture Lesson 26 · Final review: the hardest distinctions

Accountable ↔ responsible

Self-managing ↔ manager-directed

Product Goal ↔ Sprint Goal

Review ↔ Retrospective

Increment ↔ release

Refinement ↔ events

DoD ↔ acceptance criteria

Scrum Master ↔ project manager

Empiricism ↔ predictive planning

Commitment ↔ guarantee

Cover the right side of the list above and test yourself on each pair.

Your memory card

1. Most PSM I questions test distinctions inside scenarios.
2. Know all ten pairs and the three extra ones by heart.
3. Find which distinction the last line asks about.
4. Book when two timed mocks in a row reach 90% or more.
5. Reread the Guide calmly in the final week.
6. Good luck on your exam!

Official source: Your final read: the full Scrum Guide 2020, slowly, one more time.

[Open the full lesson →](#)

Deeper professional explanation

Lesson 26 · Final review: the hardest distinctions

Three more that catch people

Obsolete vs difficult Sprint Goal: only an obsolete goal justifies cancellation, and only by the Product Owner. **Scope vs quality:** scope can be renegotiated during the Sprint; quality cannot decrease. **Required vs allowed vs not mentioned:** many wrong options turn an allowed practice into a rule, or invent one.

A readiness check

You are likely ready when you can explain every distinction above without notes, you score 90% or more on two timed 80-question mock exams in a row, and you no longer get questions wrong because of outdated rules.

Your last week

Read the Scrum Guide once more, slowly. Take two full mock exams under time. Review every wrong answer and link it to a distinction in this lesson. Rest the day before the exam.

Case study: Ready or not? Decide before you read the answer (cover the green part)

Decision 1: Final mocks

One week before the exam

A student scores 88%, 91% and 93% on three timed mocks. Most errors are on Review vs Retrospective questions.

What is the best plan?

- ☐ A Book now and stop studying
- ☒ B Review that one distinction, take one more mock, then book
- ☐ C Restart the whole course from Lesson 0
- ☐ D Delay the exam by three months

Best decision: B. Targeted review fixes the weak spot. The next mock is 95%, and the student passes.

Why not the others? A leaves a known gap. C and D waste time.

Decision 2: The familiar wrong answer

Exam day

A question asks who is accountable for the Product Backlog when the PO delegates writing items to an analyst. One option names the analyst.

Which distinction decides it?

- ☒ A Accountable vs responsible
- ☐ B Review vs Retrospective
- ☐ C Increment vs release
- ☐ D Refinement vs events

Best answer: A. The analyst is responsible for the work; the Product Owner remains accountable.

Why not the others? They are real distinctions, but not the one tested here.

Common misunderstandings

- ✗ "If I know the definitions, I will pass."

✓ Most questions test distinctions inside scenarios.
- ✗ "One good mock score means I'm ready."

✓ Aim for two in a row at 90% or more.
- ✗ "The night before is for cramming."

✓ A calm, rested mind reads tricky wording better.
- ✗ "Experience replaces the Guide."

✓ The exam follows the 2020 Guide exactly.

Exam traps

The trap	How to read it
A question that mixes two distinctions.	Find the one the last line actually asks about.
An obsolete vs difficult Sprint Goal.	Only obsolete justifies cancellation.
Scope vs quality under pressure.	Scope may move; quality may not.

Exam-style questions 12 questions in the online lesson

1. Who is accountable for the Product Backlog when the PO delegates writing items?

A

 The Developers jointly

B

 The Product Owner

C

 The Scrum Master

D

 The person writing the items
2. A Sprint met its goal but left one item unfinished. What does this show?

A

 The goal is the commitment; items are a forecast

B

 The Sprint should have been extended

C

 The team broke its commitment to stakeholders

D

 The DoD should be relaxed for the next Sprint
3. A usable Increment is held for a launch event. Which distinction applies?

A

 Product Goal vs Sprint Goal choice

B

 Refinement vs events

C

 Increment vs release decision

D

 Review vs Retrospective
4. Which statements are accurate? (Choose 2)

A

 The Review inspects the team's processes

B

 Acceptance criteria replace the DoD

C

 The Product Goal is long-term; the Sprint Goal is for one Sprint

D

 Refinement is an ongoing activity, not an event

Answers 1: B Accountable vs responsible. · 2: A Commitment vs guarantee. · 3: C Usable and released differ. · 4: C and D Know the pairs.

All 12 questions with full explanations →

Free PSM I questions →

Toolkit 1 · Who decides what

the most tested distinction on PSM I

Decision	Product Owner	Scrum Master	Developers	Others
Order of the Product Backlog	accountable			influence
Sprint Goal (the whole Scrum Team defines it)	proposes value	takes part	takes part	
Which items are selected for the Sprint	collaborates		accountable	
How the work gets done; who does which task			accountable	
Sizing Product Backlog items	may influence		accountable	
Cancelling a Sprint	accountable			
Creating the Definition of Done	with team	with team	with team	org standard
Format of the Daily Scrum			accountable	
Scope change during the Sprint	negotiates		negotiates	
Removing impediments		causes removal		help
Effectiveness of the Scrum Team		accountable		
Adapting the Sprint plan daily			accountable	
Quality of the Increment			accountable	

Read every question for WHO

Many options describe a good action done by the wrong person. "The Scrum Master assigns tasks" or "the stakeholders order the backlog" are wrong even if the result looks sensible.

Accountable vs responsible

The PO can ask others to write backlog items, but the PO stays **accountable**. Delegating work never moves accountability.

Managers and stakeholders

They influence through the Product Owner and give feedback at the Sprint Review. They do not assign work to Developers.

Toolkit 2 · Glossary in simple English the exact words of the 2020 Guide

Accountability · What a person must answer for. The Guide names three: Product Owner, Scrum Master, Developers.

Adaptation · Adjusting the process or product as soon as possible when inspection shows a problem.

Artifact · Represents work or value: Product Backlog, Sprint Backlog, Increment.

Commitment · What each artifact is measured against: Product Goal, Sprint Goal, Definition of Done.

Cross-functional · The team has all the skills needed to create value each Sprint.

Daily Scrum · 15-minute event for the Developers to inspect progress toward the Sprint Goal.

Definition of Done · Formal description of the quality an Increment must meet. Not Done = not in the Increment.

Developers · People in the Scrum Team who create any aspect of a usable Increment each Sprint.

Empiricism · Knowledge comes from experience and decisions are based on what is observed.

Impediment · Anything that blocks the team's progress. The Scrum Master causes its removal.

Increment · A concrete stepping stone toward the Product Goal; must be usable and meet the DoD.

Inspection · Checking artifacts and progress frequently and diligently to detect problems.

Lean thinking · Reducing waste and focusing on the essentials.

Product Backlog · Emergent, ordered list of what is needed to improve the product. One per product.

Product Goal · The long-term objective for the Scrum Team; commitment for the Product Backlog.

Product Owner · Accountable for maximizing the value of the product. One person, not a committee.

Toolkit 2 · Glossary in simple English (continued)

the exact words of the 2020 Guide

Refinement · Ongoing activity to break down and add detail to Product Backlog items. Not an event.

Release · Delivering an Increment to users. Can happen any time, also before the Sprint Review.

Scrum Master · Accountable for establishing Scrum and for the Scrum Team's effectiveness.

Scrum Team · One Scrum Master, one Product Owner and Developers; typically 10 or fewer people.

Self-managing · The team decides internally who does what, when and how.

Sprint · Fixed-length event of one month or less; the container for all other events.

Sprint Backlog · Sprint Goal (why) + selected items (what) + plan (how). Owned by the Developers.

Sprint Goal · The single objective for the Sprint; commitment for the Sprint Backlog.

Sprint Planning · Starts the Sprint: why it is valuable, what can be Done, how it will be done.

Sprint Retrospective · Ends the Sprint: plan ways to increase quality and effectiveness.

Sprint Review · Working session to inspect the outcome of the Sprint and decide future adaptations.

Stakeholder · Anyone with an interest in the product outside the Scrum Team.

Timebox · The maximum time an event may take.

Transparency · The work and the process are visible to those doing and receiving the work.

Values · Commitment, Focus, Openness, Respect and Courage.

Velocity · Not in the Scrum Guide. A team practice some teams use for forecasting.

Toolkit 3 · The top exam traps

one from every lesson

Lesson	The trap	How to read it
L0	Answering from your workplace's practice.	Ask what the Scrum Guide says, even if your company does it differently.
L1	An option that adds buffer or detail to a long-range prediction.	More detail on a guess is still a guess. Look for the option that decides from observed results.
L2	"The Development Team is accountable for..."	An older term. Check whether the claim still matches the 2020 Guide.
L3	"Scrum Lite" or any variant with a missing part.	Implementing only parts of Scrum is possible, but the result is not Scrum.
L4	"Inspection without transparency is pointless."	The Guide calls it misleading and wasteful. "Pointless" is inspection without adaptation.
L5	A list mixing values and pillars.	Values: commitment, focus, openness, respect, courage. Pillars: transparency, inspection, adaptation.
L6	Confusing "the Scrum Team" with "the Developers".	Check which group the question names. The whole Scrum Team is accountable for the Increment; specific accountabilities belong to Developers, Product Owner or Scrum Master.
L7	"The Product Owner decides the technical approach because of value."	How is at the Developers' sole discretion.
L8	"Each team on the same product may have its own PO."	Teams on one product share one Product Owner and one Product Backlog.
L9	An option where the Scrum Master decides for the team.	Usually wrong; look for the option that helps the team decide.
L10	"Self-managing teams may set aside organizational standards."	An organizational DoD is a minimum; standards still apply.
L11	"Quality may be reduced to protect the Sprint Goal."	Quality does not decrease during the Sprint.
L12	"Each item must be broken into tasks of exactly one day."	The Guide says this is often done, not required.
L13	"The timebox can grow for larger teams."	It is 15 minutes regardless.

Toolkit 3 · The top exam traps (continued)

one from every lesson

Lesson	The trap	How to read it
L14	"The Review is the last event of the Sprint."	It is second to last; the Retrospective concludes the Sprint.
L15	"The Definition of Done is reviewed at the Sprint Review."	The Retrospective inspects the DoD.
L16	"Defects live in a separate list."	The Product Backlog is the single source of work.
L17	"The Product Owner may add items to the Sprint Backlog alone."	Scope changes are negotiated with the Developers, protecting the goal.
L18	"Undone work may be shown at the Review if labelled."	It cannot be presented as part of the Increment.
L19	"Velocity proves value."	Velocity measures output at best; value is about benefit.
L20	"The PO may waive the DoD for urgent work."	The DoD is not waived; scope changes instead.
L21	"A chief Product Owner coordinates team POs."	One product, one Product Owner.
L22	"The Scrum Master should write status reports for management."	Offer transparency through artifacts and events instead.
L23	A factory scenario with an extended Sprint.	The one-month limit applies in every domain.
L24	An integrated scenario with one rule hidden in the details.	Check every rule involved: goal, scope, quality, accountability, timebox.
L25	"3 to 9 Developers", "10% refinement", "three questions".	2017 rules; not in the 2020 Guide.
L26	A question that mixes two distinctions.	Find the one the last line actually asks about.

Toolkit 4 · Checklists

Before you book

- ☐ I have read the 2020 Scrum Guide at least twice
- ☐ I can explain all 27 memory cards without looking
- ☐ I scored 90%+ on two full timed practice exams in a row
- ☐ I know the timeboxes and who attends each event
- ☐ I can list what changed from 2017 to 2020
- ☐ I bought the password from Scrum.org (it does not expire)

On exam day

- ☐ Quiet room, stable internet, charged laptop
- ☐ Close other programs and notifications
- ☐ Water and a clock in view: 45 seconds per question
- ☐ First pass: answer every question, mark the hard ones
- ☐ Read the last sentence first; check for "choose 2"
- ☐ Keep about 10 minutes to review marked questions
- ☐ Change an answer only with a clear reason from the Guide

Toolkit 5 · My progress tracker

#	Lesson	Read	Questions	Can explain
0	The PSM I exam and how to study	☐	__ / 4	☐
1	Complex work and why Scrum exists	☐	__ / 4	☐
2	The language of Scrum	☐	__ / 4	☐
3	The framework in one picture	☐	__ / 4	☐
4	Empiricism, lean and the three pillars	☐	__ / 4	☐
5	Scrum values in real decisions	☐	__ / 4	☐
6	The Scrum Team	☐	__ / 4	☐
7	Developers	☐	__ / 4	☐
8	Product Owner	☐	__ / 4	☐
9	Scrum Master	☐	__ / 4	☐
10	Self-management and leadership	☐	__ / 4	☐
11	The Sprint	☐	__ / 4	☐
12	Sprint Planning	☐	__ / 4	☐
13	Daily Scrum	☐	__ / 4	☐

#	Lesson	Read	Questions	Can explain
14	Sprint Review	☐	__ / 4	☐
15	Sprint Retrospective	☐	__ / 4	☐
16	Product Backlog, refinement and Product Goal	☐	__ / 4	☐
17	Sprint Backlog and Sprint Goal	☐	__ / 4	☐
18	Increment, release and Definition of Done	☐	__ / 4	☐
19	Product value, stakeholders and forecasting	☐	__ / 4	☐
20	Quality and technical debt	☐	__ / 4	☐
21	Many teams, one product	☐	__ / 4	☐
22	Scrum in the organization	☐	__ / 4	☐
23	Scrum beyond software	☐	__ / 4	☐
24	One full Sprint, start to finish	☐	__ / 4	☐
25	Reading PSM I questions; 2017 vs 2020	☐	__ / 4	☐
26	Final review: the hardest distinctions	☐	__ / 4	☐

What next? Free PMCLEAR resources

Full lessons online →

All 27 lessons with 12 to 15 questions each and full explanations.

pmclean.com/psm/course/

Free PSM I sample questions →

Try exam-level questions with no account.

pmclean.com/psm/#sample

Practical Scrum labs →

Be the Scrum Master in a real team situation, step by step.

pmclean.com/psm/labs/

PSM I preparation guide →

How to prepare, in simple English.

pmclean.com/guides/psm-i-preparation.html

PSM I vs CSM →

Which Scrum Master certificate fits you.

pmclean.com/guides/psm-i-vs-csm.html

Free account: 3 days full access →

1,000 practice questions and 80-question timed exams.

pmclean.com/psm/app/#signup

Also free on PMCLEAR: [96 free PMP lessons \(PDF\)](#) and [free project management tools](#).

The small print

Independent study material. PMCLEAR is an independent educational resource. We are **not Scrum.org**, we are **not PMI**, and we are not affiliated with, sponsored by or endorsed by Scrum.org, Scrum Inc., the Project Management Institute (PMI) or the authors of the Scrum Guide.

Trademarks. Professional Scrum Master, PSM, PSM I and Scrum.org are trademarks of Scrum.org. PMI and PMP are registered marks of the Project Management Institute, Inc. All other trademarks belong to their owners. They are used here only to describe the exam this book helps you prepare for.

The Scrum Guide. Short quotations from The Scrum Guide™ (November 2020) by Ken Schwaber and Jeff Sutherland are used with attribution. The Scrum Guide is © 2020 Ken Schwaber and Jeff Sutherland and is offered under the Creative Commons Attribution-ShareAlike 4.0 International license (CC BY-SA 4.0), available at scrumguides.org. The official Guide is the only authoritative source; if this book and the Guide ever differ, the Guide is right.

Exam details can change. The exam format, number of questions, time limit, passing score, fees and policies described in this book were correct to the best of our knowledge at the time of writing (2026). Always check the official Scrum.org PSM I page before you buy a password or take the assessment.

No guarantee. This book helps you learn. It does not guarantee that you will pass any exam. The case studies, companies and people in this book are fictional; any resemblance to real organizations or persons is coincidental. The practice questions are original PMCLEAR questions written for study; they are not real exam questions.

Copyright. © 2026 PMCLEAR. All rights reserved, except the quoted Scrum Guide text, which keeps its own license. You may download, print and share this PDF free of charge, unchanged and complete, with this notice. You may not sell it, change it, or present it as your own.

Contact. Questions, corrections or feedback: pmclean.com/contact. We are grateful for every correction.