

DUMPSBOSS.

Professional Scrum Developer I (PSD 1)

Scrum PSD

Version Demo

Total Demo Questions: 10

Total Premium Questions: 120

Buy Premium PDF

<https://dumpsboss.co>

support@dumpsboss.co

support@dumpsboss.co
dumpsboss.co

QUESTION NO: 1

Which are obligatory scrum events?

- A. sprint retrospective
- B. sprint goal meeting
- C. backlog refinement
- D. sprint planning
- E. sprint review
- F. daily scrum

ANSWER: A D E F

Explanation:

:

If you are not sure, please review the Scrum events section in the scrum guide

QUESTION NO: 2

What are some disadvantages of code coverage as a measure for how well a system or product is tested? (three answers)

- A. It does not measure usability
- B. It is too complicated to explain to management
- C. It only provides insights to programmers
- D. It does not ensure that the most important or highest risk areas of the code are being exercised by tests
- E. It could create incentives to write tests that simply increase code coverage, rather than tests that find bugs without increasing code coverage

ANSWER: A D E

Explanation:

:

Not being able to explain to management should never be the reason to not do it. It doesn't only provide some data to programmers, the problem is that code coverage alone is not meaning a lot.

QUESTION NO: 3

Why does a test written using TDD always fail?

- A. Because testers are terrible coders
- B. Because the tests are checked in before the code exists
- C. Because the product code to satisfy the test doesn't exist yet
- D. Because the test needs to be refactored

ANSWER: C

Explanation:

:

TDD stands for test driven development. It makes developers think before starting to code. This leads to better organized code. It requires the developer to write the code for the test first, the test fails because the code it's testing doesn't exist yet.

QUESTION NO: 4

Which three of the following are code quality metrics?

- A. Unit tests per class
- B. Cycle time per product backlog item
- C. Number of check-ins per day
- D. Depth of inheritance
- E. Cyclomatic complexity
- F. Class coupling

ANSWER: D E F

Explanation:

:

Number of checkins per day says something about whether the developers are committing often enough. Depth of inheritance says some about complexiity, just like class coupling and cyclomatic complexity. Unit tests per class don't say

anything about quality, though no unit tests might indicate a maintenance problem. Cycle time per product backlog item says something about the scrum process.

QUESTION NO: 5

Why might a scrum team choose to do pair programming?

- A. Improving code quality
- B. All of these
- C. It is a nice way to work
- D. Information sharing and learning
- E. Efficiency

ANSWER: B

Explanation:

:

QUESTION NO: 6

Choose two common cross-cutting concerns in application architecture

- A. Release documentation
- B. Build automation
- C. Security
- D. Design pattern choices
- E. Scalability

ANSWER: C E

Explanation:

:

The others are either not cross-cutting or not related to architecture.

QUESTION NO: 7

The practice of decomposing a requirement into failing tests is called:

- A. Behavior driven development
- B. Scrum testing
- C. Regression testing
- D. Acceptance test driven development

ANSWER: D

Explanation:

:

In ATDD you start with writing tests to represent the requirement. Since you start with this, there is no production code to satisfy the test which is why the tests will fail.

QUESTION NO: 8

What is a unit test?

- A. A technique for ensuring that units of co-dependent systems integrate correctly
- B. A way to ensure product code satisfies the user requirements
- C. A test that isolates and verifies individual units of functionality
- D. A test that ensures a specific method integrates properly with the code calling it

ANSWER: C

Explanation:

:

A unit test is a test that isolates and verifies the functionality of one unit.

QUESTION NO: 9

What is an inappropriate activity for a developer with special QA skills?

- A. Create a plan for testing the application
- B. Identify tools and techniques to be used in testing the product
- C. Define acceptance criteria for features

D. Delay test activities until a feature has been coded

ANSWER: D

Explanation:

:

Test activities can start directly with the sprint.

QUESTION NO: 10

What are two reasons to use mock objects in unit tests?

- A. The behavior of a given object can be emulated to the system under test
- B. They are machine-generated and eliminate the need to write code by hand
- C. To increase tight coupling
- D. TDD is impossible without them
- E. Isolating a particular system component for controlled testing

ANSWER: A E

Explanation:

:

Mock objects are simulating the behavior of dependencies such as persistence layers or APIs. By mocking these you can be sure about what you are testing, and not accidentally testing the external dependency.