

1. What is unit testing?

1 / 1 point

- ☒ Unit testing revolves around the idea of having separate, small pieces of code that are easy to test.
- ☐ Unit testing tries to imitate how a user might interact with your app.
- ☐ Unit testing gives you an efficiency rating for your code measured in units.
- ☐ Unit testing is testing how parts of your system interact with other parts of your system.



Correct

That's correct! Unit testing is focused on testing small, specific "units" of code.

2. When the following test executes, what will the test result be?

1 / 1 point

```
function subtract(a, b) { return a - b; } expect(subtract(10, 4)).toBe(6);
```

- ☒ Success.
- ☐ Fail.



Correct

That's correct! The function will return **6** and the expectation will succeed.

3. What is End-to-end testing (e2e)?

1 / 1 point

- ☐ End-to-end testing revolves around the idea of having separate, small pieces of code that are easy to test.
- ☐ End-to-end testing is testing how parts of your system interact with other parts of your system.
- ☐ End-to-end testing is a comprehensive test that starts at the beginning of your code and runs to the end.
- ☒ End-to-end testing tries to imitate how a user might interact with your application.

✓ **Correct**

That's correct! End-to-end testing mimics how users will interact with your application.

4. What reason would you want to know about your code coverage?

1 / 1 point

- ☐ It lets you know if your code is in Javascript.
- ☒ It lets you know where more testing may be required.
- ☐ It lets you know if you are testing correctly.
- ☐ It lets you know if you are done writing code.

✓ **Correct**

That's correct! Code coverage shows what percentage of your code is covered by tests and where more testing may be required.

5. Which of the following runtimes are typically used for building backends?

1 / 1 point

- ☒ **Node.js**
- ☐ Protractor
- ☐ Jest
- ☐ Cypress



Correct

That's correct! Node.js is a popular JavaScript runtime for building backends.

6. Which of the following are **NOT** one of the three types of testing?

1 / 1 point

- ☐ Integration testing
- ☐ End-to-end testing
- ☒ Post-hoc testing
- ☐ Unit testing



Correct

That's correct! Unit testing, integration testing, and end-to-end testing are the three types of testing.

7. True or False: You can install packages from the **npm** repository using the node command.

1 / 1 point

- ☐ True
- ☒ False

✓ Correct

That's correct! The **npm** command is used to install packages.

8. **package.json** is used to do which of the following?

1 / 1 point

- ☒ Store all the dependencies required for application.
- ☐ Turn your code into an application.
- ☐ Download **npm** packages.
- ☐ Store all the testing code.

✓ Correct

That's correct! **package.json** contains human-readable metadata about the project (like the project name and description) as well as functional metadata like the package version number and a list of dependencies required by the application.

9. True or False: End-to-end tests can be performed in a web browser without writing code.

1 / 1 point

- ☒ True
- ☐ False



Correct

That's correct! This is because the tests mimic how a user will interact with the application.

10. What is the recommended way to separate the code that you are testing from its related dependencies?

1 / 1 point

- ☐ End-to-end testing
- ☒ Mocking
- ☐ Fakes
- ☐ module.exports



Correct

That's correct! Mocking allows developers to simulate the behavior of dependent code during tests.