Quiz Answers for the pytest course

Chapter 1

1. What command prints the current Python version?
- ◉ python --version
- ○ python version
- ○ pyversion
- ○ python --v

2. Conventionally, tests in a Python project belong under a directory named "tests".
- ◉ true
- ○ false

3. What command does pytest used for making assertions?
- ○ pytest.assert
- ○ verify
- ◉ assert
- ○ check

4. What is the best command for running pytest tests?
- ○ pytest run
- ○ python <path-to-test>
- ○ pytest
- ◉ python -m pytest

5. pytest treats all functions in modules under the "tests" directory as test cases.
- ○ true
- ◉ false

Note: 100 credits is for successful completion on the first try; 50 credits for the second try, and 25 credits thereafter

Chapter 2

1. One Python test module can contain more than one test case function.

- ⦿ true
- ○ false

2. What symbol does pytest use in its reports to denote a passing test?

- ○ !
- ○ P
- ○ p
- ⦿ .

3. What symbol does pytest use in its reports to denote a failing test?

- ○ -
- ⦿ F
- ○ .
- ○ x

4. What types of exceptions will make a pytest test case fail?

- ⦿ any type of exception
- ○ only AssertionError
- ○ any exception other than AssertionError

5. By default, pytest will print test code snippets, failure reasons, and test result tallies for failed test cases.

- ⦿ true
- ○ false

Note: 100 credits is for successful completion on the first try; 50 credits for the second try, and 25 credits thereafter

Chapter 3

✕ 1. What is the purpose of "pytest.raises"?

- ◯ To verify that the code under test raises expected exceptions.
- ⬤ To catch exceptions raised by the pytest framework.
- ◯ To raise new exceptions when test failures happen.
- ◯ To change the order of test cases.

2. "pytest.raises" must be imported from the pytest module.

- ⬤ true
- ◯ false

3. "pytest.raises" does not look for exceptions based on type.

- ◯ true
- ⬤ false

4. Why does "pytest.raises" need to be used with a "with" statement?

- ◯ The "with" statement adds additional logs to the test report.
- ◯ False – "pytest.raises" can be used without a "with" statement.
- ⬤ The "with" statement executes additional "enter" and "exit" logic for handling raised exceptions.
- ◯ The "with" statement enables the test case to run functions from other modules.

5. How can "pytest.raises" store the raised exception object for further assertions?

- ◯ This is not possible.
- ◯ Include a variable name as the second argument in the "pytest.raises" call.
- ⬤ Add an "as" clause to the end of the "with" statement to store the exception into a variable.
- ◯ Use the "exception" keyword to reference the most recent exception.

Submit

Chapter 4

1. What is an "equivalence class" for test case inputs?

- ○ A range of possible output values.
- ○ Duplicate test parameters.
- ○ A hard-coded set of input values.
- ● A unique kind of input that yields a unique kind of output.

2. What is the name of the pytest decorator for parametrized inputs?

- ○ @pytest.parametrize
- ● @pytest.mark.parametrize
- ○ @pytest.mark.parameterize
- ○ @pytest.parameters

3. What is the data format for passing pytest parameters?

- ○ a list of Parameter objects
- ○ a one-dimensional list of parameter values
- ○ a dictionary of parameter names to values
- ● a list of parameter value tuples

4. pytest parameters may be strings as well as numbers.

- ● true
- ○ false

5. pytest parameters cannot be read from external files.

- ○ true
- ● false

Chapter 5

1. What are "unit tests"?

- ○ Small tests that cover variable assignments
- ○ Large tests that cover Web UIs and REST APIs
- ● Small tests that directly cover "units of work" like functions and methods
- ○ Manual tests that interact with the product under test like a user

2. What file turns a regular directory into a package in a Python project?

- ○ _init_.py
- ○ init.py
- ● __init__.py
- ○ __init.py

3. pytest can run tests from multiple modules in the same Python project.

- ● true
- ○ false

4. Which of the following lines represents an "Arrange" step?

- ○ accum.add()
- ○ accum.add(3)
- ● accum = Accumulator()
- ○ assert accum.count == 0

5. Which of the following lines represents an "Assert" step?

- ○ accum.add()
- ○ accum = Accumulator()
- ● assert accum.count == 0
- ○ accum.add(3)

Note: 100 credits is for successful completion on the first try; 50 credits for the second try, and 25 credits thereafter

Chapter 6

1. In pytest, what is a fixture?

- ◯ Any function in a test module that is not a test case.
- ⦿ A function that handles setup and cleanup operations for a test case.
- ◯ A term that refers to the overall structure of a test case module.
- ◯ Another name for a test case function.

✖ 2. What is the name of the decorator used for defining functions as fixtures?

- ◯ @pytest.fixture
- ⦿ @pytest.mark.fixture
- ◯ @pytest.fix
- ◯ @fixture

3. Fixtures cover the "Act" phase of "Arrange-Act-Assert".

- ◯ true
- ⦿ false

4. To share fixtures between test modules, where should the fixtures be located?

- ⦿ In a "conftest.py" module under the "tests" directory.
- ◯ In any module under the "tests" directory.
- ◯ In any module in the project.
- ◯ In a "fixtures.py" module under the "tests" directory.

5. A test case cannot use multiple fixtures.

- ◯ true
- ⦿ false

✖ 6. A fixture can provide both setup and cleanup logic by using a yield statement.

- ◯ true
- ⦿ false

Chapter 7

1. How many different command line options does pytest have?

- ● Many more than we can cover in this course!
- ○ 12
- ○ 54
- ○ none

2. What is the option for listing all available command line options with descriptions?

- ● "-h" or "--help"
- ○ "-help"
- ○ "-H"
- ○ "--h"

3. Which of the following is NOT printed by pytest when the "--verbose" option is used?

- ○ failure introspection
- ○ metadata
- ● dots for passing tests and F's for failing tests
- ○ the name and path of each test case with its result

4. pytest does not print failure introspection when the "--quiet" option is used.

- ○ true
- ● false

5. pytest provides options for ending a test suite run after a certain limit of tests fail.

- ● true
- ○ false

6. pytest can generate JUnit XML reports.

- ● true
- ○ false

7. Which of the following is NOT a supported pytest configuration file type? (Hint: Check the "Configuration" page in the pytest docs online!)

- ● pytest.cfg
- ○ pytest.ini
- ○ tox.ini
- ○ setup.cfg

Note: 100 credits is for successful completion on the first try; 50 credits for the second try, and 25 credits thereafter

Chapter 8

1. Which command would run all test cases under the "tests/unit" directory?

- ○ python -m pytest -m tests/unit
- ○ python -m pytest --path tests/unit
- ⦿ python -m pytest tests/unit
- ○ python -m pytest unit

2. Which command would run only a test named "test_login" in the "tests/test_page.py" module?

- ○ python -m pytest test_login
- ○ python -m pytest tests/test_page::test_login
- ○ python -m pytest --test test_login
- ⦿ python -m pytest tests/test_page.py::test_login

3. Which command would run all tests that contain "upload" in the test function name?

- ⦿ python -m pytest -k upload
- ○ python -m pytest -k *upload*
- ○ python -m pytest --grep upload
- ○ python -m pytest *upload*

4. Filtering expressions for "-k" and "-m" can use Boolean conditions with multiple tokens.

- ⦿ true
- ○ false

5. Custom marker names should be added to pytest configuration files.

- ⦿ true
- ○ false

6. What marker can be used to skip a test case?

- ○ @pytest.skip
- ⦿ @pytest.mark.skip
- ○ @pytest.mark.ignore
- ○ @skip

Note: 100 credits is for successful completion on the first try; 50 credits for the second try, and 25 credits thereafter

Chapter 9

1. Why is it important to test code directly with unit tests?

- ○ Unit tests run quickly, and many can be run
- ○ Unit tests are typically small and simple to write
- ○ Unit tests catch low-level problems at the source
- ● All these reasons

2. REST API tests and Web UI tests are types of feature tests.

- ● true
- ○ false

3. Feature tests are typically faster than unit tests.

- ○ true
- ● false

4. What is the name of the most popular Python module for making REST API requests?

- ○ rest_api
- ● requests
- ○ http
- ○ requester

5. What is the name of the Python package that extends pytest for writing REST API tests using YAML?

- ● tavern
- ○ pytest-bdd
- ○ services
- ○ requests

Note: 100 credits is for successful completion on the first try; 50 credits for the second try, and 25 credits thereafter

Chapter 10

1. What is a plugin?

- ● An optional package that adds new capabilities to a framework
- ○ A package that is part of a framework's standard library
- ○ An asset that lets pytest run integration tests
- ○ A special test case that tests aspects of the framework

2. What is the name of the package that generates HTML reports for pytest results?

- ○ pytest-html-report
- ● pytest-html
- ○ pytest-report
- ○ pytest-report-html

3. Using pytest-cov, what types of code should be checked for code coverage?

- ○ all types of code should have code coverage
- ● only product code
- ○ product code and test code
- ○ only test code

4. Tests that share resources or data can always run safely in parallel using pytest-xdist.

- ○ true
- ● false

5. What does "BDD" for the "pytest-bdd" plugin stand for?

- ● Behavior-Driven Development
- ○ Buzzword-Driven Development
- ○ Bad Data Discrepancy
- ○ Behavior-Driven Data

6. Writing a pytest plugin requires a deep understanding of pytest's inner workings, such as hooks, fixtures, and command line options.

- ● true
- ○ false

Note: 100 credits is for successful completion on the first try; 50 credits for the second try, and 25 credits thereafter