**Test Driven Development in C++**

**Chapter 2: Overview of Test Driven Development Quiz**

1. At what level of the software is unit testing normally performed?

A. System Level

B. Component Level

C. Class and Function Level

2. Who typically implements and runs unit tests?

A. System Test Engineers

B. Integration Engineers

C. Software Developers

D. Product Owners

3. In test driven development, you write all the unit tests first and then write all the production code.

A. True

B. False

4. What are the three phases of Test Driven Development?

A. Red, Green, Refactor

B. Test, Code, Release

C. User Stories, Tests, and Coding

5. What are one of the benefits of Test Driven Development?

A. Gives you immediate feedback in how your changes have affected things.

B. Documents the code.

C. Drives good object-oriented design.

D. All of the above

**ANSWER KEY**

1. At what level of the software is unit testing normally performed?

A. System Level

* Incorrect. Unit tests are generally the lowest level of testing performed by the developer usually at the class or function level.

B. Component Level

* Incorrect. Unit tests are generally the lowest level of testing performed by the developer usually at the class or function level.

**C. Class and Function Level**

* Correct! Unit tests are generally the lowest level of testing performed by the developer usually at the class or function level.

2. Who typically implements and runs unit tests?

A. System Test Engineers

* Incorrect. Unit tests are generally the lowest level of testing performed by the developer usually at the class or function level.

B. Integration Engineers

* Incorrect. Unit tests are generally the lowest level of testing performed by the developer usually at the class or function level.

**C. Software Developers**

* Correct! Unit tests are generally the lowest level of testing performed by the developer usually at the class or function level.

D. Product Owners

* Incorrect. Unit tests are generally the lowest level of testing performed by the developer usually at the class or function level.

3. In test driven development, you write all the unit tests first and then write all the production code.

A. True

* Incorrect. In Test Drive Development, you write one failing unit test and then you write the production code to make it pass.

**B. False**

* Correct! In Test Drive Development, you write one failing unit test and then you write the production code to make it pass.

4. What are the three phases of Test Driven Development?

**A. Red, Green, Refactor**

* Correct! In the Red phase, you implement a failing unit test. Then in the Green phase, you make it pass. Then in the refactor phase, you modify the production code and the test to make remove duplication and make it clean.

B. Test, Code, Release

* Incorrect. The three phases are Red, Green, Refactor. In the Red phase, you implement a failing unit test. Then in the Green phase, you make it pass. Then in the refactor phase, you modify the production code and the test to make remove duplication and make it clean.

C. User Stories, Tests, and Coding

* Incorrect. The three phases are Red, Green, Refactor. In the Red phase, you implement a failing unit test. Then in the Green phase, you make it pass. Then in the refactor phase, you modify the production code and the test to make remove duplication and make it clean.

5. What are one of the benefits of Test Driven Development?

A. Gives you immediate feedback in how your changes have affected things.

* This isn't the best answer! The correct answer is "All of these" as each of these examples are a benefit of TDD.

B. Documents the code.

* This isn't the best answer! The correct answer is "All of these" as each of these examples are a benefit of TDD.

C. Drives good object-oriented design.

* This isn't the best answer! The correct answer is "All of these" as each of these examples are a benefit of TDD.

**D. All of the above**

* Correct! This is the best answer! The correct answer is "All of the above" as each of these examples are a benefit of TDD.