Congratulations! You passed!

TO PASS 80% or higher

Keep Learning

GRADE
100%

Object-Oriented Programming with Java

LATEST SUBMISSION GRADE
100%

1. Which of the following statements about object-oriented programming is **INCORRECT**? **1 / 1 point**

- ○ Objects can contain data, called "**properties**", and functions that manipulate the data, called "**methods**".
- ○ Objects often interact with each other much like they do in real life.
- ○ Objects can hide certain details from other objects
- ◉ In object-oriented programming. Data and functions that manipulate them are usually separate entities that interact with each other.

✓ **Correct**
Well done! This statement is incorrect. In object-oriented programming, data and functions that manipulate them are **often** bundled into objects. This is called "**encapsulation**".

2. Which of the following keyword is used in Java to define the blueprint of an object? **1 / 1 point**

- ○ private
- ○ method
- ◉ class
- ○ constructor

✓ **Correct**
Well done. In Java and many other languages, classes are blueprints of objects.

3. In Java, which of these is a correct statement to call the constructor of the ancestor class named "**Animal**" with no parameters? **1 / 1 point**

- ○ **Animal**.constructor( );
- ○ super.constructor( );
- ○ ancestor( ).
- ◉ super( );

✓ **Correct**
Well done! The constructor of the ancestor class is simply super( ).

4. If we want to extend a base class called "**Flower**" and call it "**Rose**", which of these is the correct syntax to do that in Java? **1 / 1 point**

- ◉ public class **Rose** extends **Flower** {

  }
- ○ public class **Rose** as **Flower** {

  }

- public class **Flower** extends **Rose** {

  }
- public class **Flower** descendent **Rose** {

  }

✓ **Correct**
Correct!

5. Which of the follow statement is **CORRECT** about abstraction? **1 / 1 point**

- Abstraction is the practice of making code more obscure to read and understand so other people who look at the code have no idea what it does.
- Abstraction is a way for base objects to behave more generically when compared to their descendents.
- (•) Abstraction is the practice of hiding complex or proprietary processes from users of a class.
- Abstraction is a principle of object-oriented programming where abstract variable names are used to make the code more compact.

✓ **Correct**
Correct!

6. In Java, if we have a property **String secretCode**, which of these declarations hides it from other users of the objects? **1 / 1 point**

- nonpublic **String secretCode**;
- (•) private **String secretCode**;
- abstract **String secretCode**;
- hidden **String secretCode**;

✓ **Correct**
Correct! The private keyword means that a property or method is only accessible from within the object.

7. Which of these statements about polymorphism is **correct**? **1 / 1 point**

- (•) Polymorphism makes it possible for methods of objects in the same object hierarchy to have different behaviors.
- Polymorphism allows methods of an object to have different names in their descendents.
- Polymorphism refers to the principle of object-oriented programmer where one definition of an object can have multiple names.
- Polymorphism allows objects to become other types of unrelated objects during program execution.

✓ **Correct**
Correct!

8. If we want a method **public double offerBestPrice( )** in our object to have a completely different algorithm than the one in the ancestor class, how would we redefine it in our object? **1 / 1 point**

- #OVERRIDE **public double offerBestPrice( )**
- (•) @Override **public double offerBestPrice( )**
- **public double offerBestPrice( )**
- @Redefine **public double offerBestPrice( )**

✓ **Correct**

Correct!

9. What does the Java keyword "**protected**" do? **1 / 1 point**

- ○ It is used in front of a class declaration to make sure that the class **CANNOT** be inherited by any descendent classes.
- ○ When it is in front of a property or method definition, it makes that property or method accessible from within the class itself **ONLY**
- ◉ When it is in front of a property or method definition, it makes that property or method accessible from within the class itself and **ALSO** from any descendent classes.
- ○ It makes everything in the class protected and **inaccessible** to any code outside of the class.

✓ **Correct**

Correct!

10. Which of the following statements about constructors in Java is **CORRECT**? **1 / 1 point**

- ○ All the parameters in a constructor definition are optional at runtime.
- ○ An object **CAN** have multiple constructors as long as they have different names.
- ◉ An object **CAN** have multiple constructors but they must have different parameter lists. The one matching the parameters provided by the caller at runtime will be called.
- ○ An object **CANNOT** have multiple constructors.

✓ **Correct**

Correct!