socrative by MasteryConnect

Java 111 Chapter 7 (from version 1)

Total Questions: 12

Most Correct Answers: **#6**

Least Correct Answers: **#11**

1. **A subclass inherits all the fields and methods from its superclass.**

9/11 (A) True

2/11 (B) False

2. **Which of the following are true?**

11/11 (A) You can write a new instance method in the subclass that has the same signature as the one in the superclass, thus overriding it.

9/11 (B) You can declare new methods in the subclass that are not in the superclass.

8/11 (C) You can declare new fields in the subclass that are not in the superclass.

0/11 (D) None of the above.

3. **The 'IS-A' test can be used to determine whether an item:**

7/11 (A) is a subclass

0/11 (B) should be a method

3/11 (C) is an attribute

1/11 (D) should be overridden

4. **Considering the code, which of the following is true? public class Cat extends Animal { . . . }**

0/11 (A) Cat 'has-a' Animal

10/11 (B) Cat 'is-a' Animal

1/11 (C) Animal 'is-a' Cat

0/11 (D) Cat 'has-a' Cat

5. **A kumquat is:**

1/11 (A) An orange

1/11 (B) A Grape

0/11 (C) Both

9/11 (D) Citrus

6. A subclass can use the "extends" keyword to extend multiple superclasses.

0/11 A True

11/11 B False

7. A superclass called Fruit contains a method called display() that outputs a message to the terminal. Apple is a subclass of Fruit. Which code segment IN THE SUBCLASS Apple will successfully call that method?

5/11 A super.display();

4/11 B Fruit.display();

0/11 C display(Fruit);

2/11 D Apple.display();

8. Which of the following are true with regard to inheritance?

2/11 A A subclass must have methods or instance variables in its source code

10/11 B A subclass is can be a superclass to another class

11/11 C A subclass is able to override methods of a superclass

10/11 D A subclass inherits instance variables and methods from a superclass

9. A superclass with a method that has the same header as a subclass will override the subclass' method.

1/11 A True

10/11 B False

10. You have two classes show below. If you run a test drive and create a Surgeon object and call the treatPatient method, what will the output be?

```
public class Doctor {

    boolean worksAtHospital;

    void treatPatient() {
        System.out.println("Pop some pills");
    }
}
public class Surgeon extends Doctor {

    void treatPatient() {
        System.out.println("I'm going to put you under and perform surgery");
    }

    void makeIncision() {
        System.out.println("Snip, Snip");
    }
}
```

0/11 A Pop some pills

2/11 B Pop some pills I'm going to put you under and perform surgery

8/11 C I'm going to put you under and perform surgery

1/11 D I'm going to put you under and perform surgery Snip, Snip

11. Method "overloading" is when:

3/11 A Two or more methods have the same name, and the same number and type of parameters, but different return types

4/11 B Two or more methods have the same name but different numbers or types of parameters

0/11 C A method can take any number of arguments of the same type

4/11 D A method calls the method of the same name of its superclass

12. Assuming the upper code is part of a Thing class and the lower code has an instance of the class called thing, what will the result be?

```
...
public String getValue() {
    return "20";
}
public int getValue() {
    return 20;
}
...
System.out.println("Twice thing's value is: "
        (thing.getValue() + thing.getValue()) );
...
```

1/11 (A) Twice thing's value is: 40

4/11 (B) Twice thing's value is: 2020

0/11 (C) The code will not run

6/11 (D) The code will not compile