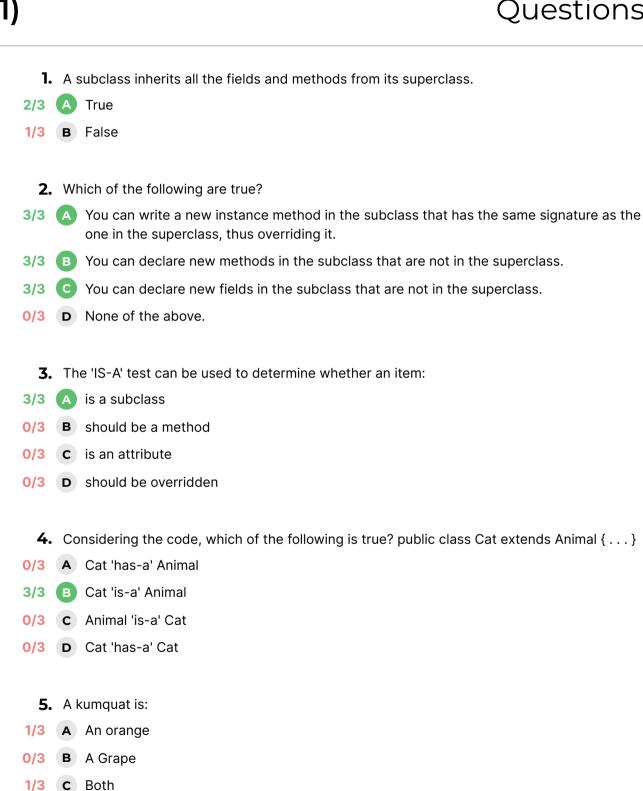


1/3 D Citrus

## Java 111 Chapter 7 (from version 131) Questions



- **6.** A subclass can use the "extends" keyword to extend multiple superclasses.
- 2/3 A True
- 1/3 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?
- 1/3 A super.display();
- 2/3 B Fruit.display();
- 0/3 C display(Fruit);
- 0/3 D Apple.display();
  - **8.** Which of the following are true with regard to inheritance?
- 0/3 A A subclass must have methods or instance variables in its source code
- 3/3 B A subclass is can be a superclass to another class
- 3/3 C A subclass is able to override methods of a superclass
- 3/3 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.
- 0/3 **A** True
- 3/3 **B** False
- **10.** You have two classes shown 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 {
    boslem ormatiospital;
    void treatPatient()
    }
    SetEmmont.orialn("Pop some pills");
}

public class Surgeon extends Doctor {
    void treatPatient() {
        System.out.printin("I'm going to gut you under and perform surgery");
    void makeIncision() {
        System.out.printin("Snip, Snip");
        System.out.printin("Snip, Snip");
    }
}
```

- 0/3 A Pop some pills
- 0/3 B Pop some pills I'm going to put you under and perform surgery
- 3/3 C I'm going to put you under and perform surgery
- 0/3 D I'm going to put you under and perform surgery Snip, Snip
  - **11.** Method "overloading" is when:
- 0/3 A Two or more methods have the same name, and the same number and type of parameters, but different return types
- 3/3 B Two or more methods have the same name but different numbers or types of parameters
- 0/3 C A method can take any number of arguments of the same type
- 0/3 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()) );
...
```

- 0/3 A Twice thing's value is: 40
- 2/3 B Twice thing's value is: 2020
- 0/3 C The code will not run
- 1/3 D The code will not compile
  - **13.** Write the method header for a method named "playRecord" that accepts three parameters: an int for the track number, an int for the speed (rpms), and a String for the track title. The method will not return anything.

## X Anon anondaf1ce42d8bf4011

1/3 public void playRecord(int trackNumber, int speedRPM, String trackTitle)

## Anon anone129b452a63741b4

1/3 void playRecord(int, rpms, String) {}

## X Anon anonf1ac7f49a13a48b1

public void playRecord(int trackNumber, int speed, String trackTitle) {
}