socrative
by MasteryConnect

12/05/2016

Java 111 Final Exam Review (from version 1)

Total Questions: 28

Most Correct Answers: **#28**

Least Correct Answers: **#19**

1. Instance variables are variables declared inside a method or method parameter.

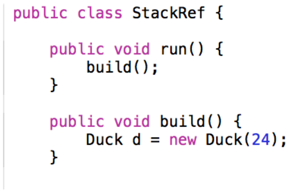
2/13 A True

0/13 B True only in an interface

0/13 C True only in an abstract class

11/13 [✓ D] False

2. In the code example below, which is the object reference variable?



```
public class StackRef {

    public void run() {
        build();
    }

    public void build() {
        Duck d = new Duck(24);
    }
```

0/13 A Duck()

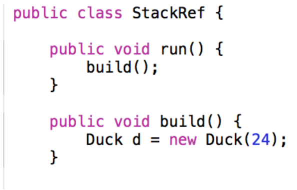
0/13 B 24

13/13 [✓ C] d

0/13 D new Duck()

0/13 E Duck

3. In the code example below, where will the Duck object live?



```
public class StackRef {

    public void run() {
        build();
    }

    public void build() {
        Duck d = new Duck(24);
    }
```

11/13 [✓ A] Heap

2/13 B Stack

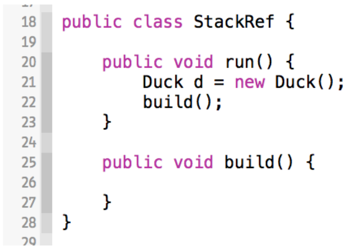
0/13 C Frame

4. The currently executing method is the one on the bottom of the stack.

1/13 A True

12/13 [✓ B] False

5. In the code example below, how long will "d" live on the stack?



```
public class StackRef {

    public void run() {
        Duck d = new Duck();
        build();
    }

    public void build() {

    }
}
```

9/13 [✓ A] Until run() pops off the stack

2/13 B Until build() is added to the top of the stack

2/13 C Until build() pops off the stack

0/13 D Until d is garbage collected

6. Which of the following are true?

12/13 [✓ A] A constructor is the code that runs when somebody says "new" on a class type, like this: Duck d = new Duck();

12/13 [✓ B] A constructor must have the same name as the class and no return type

0/13 C If you add a 2-argument constructor to a class, the compiler will create a default constructor

12/13 D The default constructor created by the compiler has no arguments.

7. Constructors cannot be overloaded.

2/13 A True

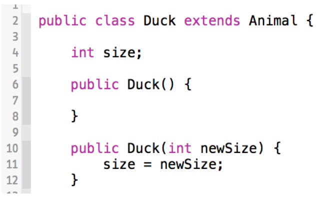
11/13 B False

8. A constructor on an object's parent must run when you make a new object.

9/13 A True

4/13 B False

9. Type the code that should appear on line 7 to call the Duck's single arg constructor with a parameter of 14.



```
 2  public class Duck extends Animal {
 3
 4      int size;
 5
 6      public Duck() {
 7
 8      }
 9
10      public Duck(int newSize) {
11          size = newSize;
12      }
```

Anon anon078c90873da74dd7

✗ 14

Anon anon1c513292729945da

✗ size(14);

Anon anon1efc3a600a10437e

✗ this.Duck(14);

Anon anon29632b75fdaa43d3

✓ this(14);

Anon anon66f2125bd8374481

✓ this(14);

Anon anon7c9f795df51a4e04

✗ this(14)

Anon anon88396baf770b4b30

✗ size.

Anon anon95166b4e77364ec0

✗ Duck size = new Duck(14);

Anon anone32f90af23cd4ba7

✗ super();

Anon anone8d48a36edf24991

✓ this(14);

Anon anonf0818d982dab48db

✗ Duck(14);

Anon anonfa61e8d1973b4c7b

✓ this(14);

Anon anonff7b84812f8f49b9

✗ this.duck(14);

10. A constructor can have a call to super() OR this(), but NEVER both.

11/13 ● A True

2/13 ○ B False

11. A class must be put into a directory structure that matches the package hierarchy.

12/13 ● A True

1/13 ○ B False

12. The keyword "static" lets a method run without any instance of the class.

11/13 ● A True

2/13 ○ B False

13. A static method cannot be dependent on any instance variable.

9/13 ● A True

4/13 ○ B False

14. Static final variables are also known as:

0/13 ○ A Constructor args

0/13 ○ B Primitives

0/13 ○ C Booleans

12/13 ● D Constants

0/13 ○ E None of the above

15. Which of the following are true?

12/13 ● A The naming convention for constants is to make the name all uppercase.

12/13 ● B A final class cannot be extended (subclassed).

10/13 ● C A static method can access a static variable.

1/13 ○ D A final method can be overridden.

0/13 ○ E None of the above

16. An abstract class can only have abstract methods.

4/13 ○ A True

9/13 ● B False

17. Marking a class with the "abstract" keyword prevents a developer from instantiating that class.

12/13 [✓ A] True

1/13 B False

18. Which of the following are true?

2/13 A An interface must be created using the keyword "abstract".

11/13 [✓ B] An interface defines only abstract methods.

12/13 [✓ C] A class can implement multiple interfaces.

10/13 [✓ D] All interface methods are implicitly public.

0/13 E None of the above.

19. Write an abstract method called calculatePremium that two parameters: a double for rate and an int for term. The method has returns a double.

Anon anon078c90873da74dd7

✗ public abstract double calculatePremium();

Anon anon1c513292729945da

✗ public abstract (double rate, int term);

Anon anon1efc3a600a10437e

✗ public abstract double calculatePremium(double rate , int term);

Anon anon29632b75fdaa43d3

✗ abstract double calculatePremium(double rate, int term);

Anon anon66f2125bd8374481

✗ public abstract calculatePremium(double rate, int term);

Anon anon7c9f795df51a4e04

✗ abstract double calculatePremium(double rate, int term);

Anon anon95166b4e77364ec0

public abstract double calculatePremium(double rate int term) {

✗ }

Anon anone8d48a36edf24991

✓ public abstract double calculatePremium(double rate, int term);

Anon anonf0818d982dab48db

✗ public double abstract calculatePremium();

Anon anonfa61e8d1973b4c7b

✗ public abstract double calculaePremium(double rate, int term);

20. If you override a superclass method in a subclass, you cannot invoke (call) the superclass method.

0/13 A True

13/13 [✓ B] False

21. Given the following: SuperStarCoder rockstar = new JavaRockStar(); what is the object reference type?

2/13 A JavaRockStar

2/13 B rockstar

9/13 [✓ C] SuperStarCoder

0/13 D new

0/13 E None of the above

22. Given the following: SuperStarCoder rockstar = new JavaRockStar(); what is the actual object type?

12/13 [✓ A] JavaRockStar

1/13 B rockstar

0/13 C SuperStarCoder

0/13 D new

0/13 E None of the above

23. What is the proper way to create an interface called Payable?

0/13 A public abstract interface class Payable {}

0/13 B public abstract Payable {}

0/13 C public abstract interface class Payable extends Payable {}

11/13 [✓ D] public interface Payable {}

1/13 E None of the above

24. Which of the following are true?

9/13 [✓ 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.

11/13 [✓ B] You can declare new methods in the subclass that are not in the superclass.

11/13 [✓ C] You can declare new fields in the subclass that are not in the superclass.

9/13 [✓ D] A subclass cannot extend multiple superclasses.

8/13 E A subclass inherits the private instance variables and private methods of its superclass.

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

12/13 [✓ A] super.display();

0/13 B Fruit.display();

0/13 C display(Fruit);

0/13 D Apple.display;

0/13 E this.display();

26. Which of the following are true:

6/13 [✓ A] The arguments and return types of an overriding method must look to the outside world exactly like the overridden method in the superclass.

11/13 [✓ B] Overloaded methods have the same name, but different argument lists.

8/13 [✓ C] Overloaded methods can have different return types as long as the arguments lists are different.

1/13 D When overriding a public method, you can declare the method as private.

3/13 E All of the above are true.

27. An import statement saves you from having to type out the full name of classes.

11/13 [✓ A] True

2/13 B False

28. What is the proper way to add a Kumquat object to an ArrayList of Kumquats called myKumquatList? Assume I have created a Kumquat object like this: Kumquat myKumquat = new Kumquat();

0/13 A myKumquatList.kumquat = myKumquat;

0/13 B myKumquatList[0] = myKumquat;

13/13 [✓ C] myKumquatList.add(myKumquat);

0/13 D myKumquatList[1].add(myKumquat)