

Quiz-4 Advance Programming

Time:30 minutes

Name	
Roll number	

Q1) What will happen to a working code when a parameterized constructor is added to a class that didn't have one earlier? Assume an object of that class was being created in the code.

- a) compilation error b) no compilation error
c) no runtime error **d) runtime error**

Q2) Which of the following is/are TRUE about abstract classes in Java (more than one correct)?

- a) If a child class does not implement all the abstract methods of the parent abstract class, then the child class should also be marked as abstract using 'abstract' keyword**
b) Abstract classes can have constructors
c) A class can be made abstract without any abstract method
d) A class can inherit from multiple abstract classes.

Q3) Mark the correct answer.

```
abstract class demo{
    public int a;
    demo(){ a = 10; }
    abstract public void set();
    abstract final public void get();
}
class Test extends demo{
    public void set(int a){ this.a = a;}
    final public void get() {
        System.out.println("a = " + a);
    }
    public static void main(String[] args){
        Test obj = new Test();
        obj.set(20);
        obj.get();
    }
}
```

- a) a=10 b) a=20
c) Compilation error d) None of these

Q4) Predict the output of the below code.

```
class Point {
    int m_x, m_y;
    public Point(int x, int y) { m_x = x; m_y = y; }
    public int getX() { return m_x; }
    public int getY() { return m_y; }

    public static void main(String args[]) {
        Point p = new Point(10,20);
        System.out.println(getX());
    }
}
```

- a)10 b)0
c)Compilation Error d)None of the above

Q5) What's the output?

```
class Outer {
    public static int temp1 = 1;
    private static int temp2 = 2;
    public static int temp3 = 3;
    private int temp4 = 4;
    public class Inner{
        private static int temp5 = 5;
        private static int getSum(){
            return (temp1 + temp2 + temp3 + temp4 +temp5);
        }
    }
    public static void main(String[] args)
    {
        Outer.Inner obj = (new Outer()).new Inner();
        System.out.println(obj.getSum());
    }
}
```

- a)15 b) 9
c) 5 **d)Compilation Error**

Q6) What will be the output?

```
interface A{
    public void method1();
}
class One implements A{
    public void method1(){
        System.out.println("Class One Method1");
    }
}
class Two extends One{
    public void method1(){
        System.out.println("Class Two Method1");
    }
}
public class Test {
    public static void main(String[] args) {
        A a = new Two();
        a.method1();
    }
}
```

- a.) Class One method1
- b.) Class Two method1**
- c.) Nothing will be printed
- d.) Compilation Error

Q7) Given the following definitions, which assignments are legal?

```
class Box<T> { }
class SuperBox<T> extends Box<T> { }
```

- a.) `Box<Object>b = new Box<String>();`
- b.) `Box<String>b = new SuperBox<String>();`**
- c.) `Box<Object>b = new SuperBox<String>();`
- d) None of the above

Q8) Given the following definition of Bird and Chicken, which of the given statements will not compile?

```
abstract class Bird implements Livestock{ }
class Chicken extends Bird { }
```

- (a) `Bird bird = new Chicken();`
- (b) `Livestock livestock = new Chicken();`
- (c) `Bird bird = new Bird();`**
- (d) None of these will compile

Q9) Suppose the class Undergraduate extends the class Student which extends the class Person. Given the following variable declaration:
`Person p = new Person();`
`Student s = new Student();`
`Undergraduate ug = new Undergraduate();`

- 1) `p = ug;` 2) `p = new Undergraduate();`
- 3) `ug = new Student();`
- 4) `ug = p;` 5) `s = new Person();`

Which of the above assignments are legal?

- a) 1 and 2**
- b) Only 2
- c) 3, 4 and 5
- d) None of the above

Q10) Which statement is/are false in java language:

- (a) A public member of a class can be accessed in all the packages.
- (b) A private member of a class cannot be accessed by the methods of the same class.**
- (c) A private member of a class cannot be accessed from its derived class.
- (d) A protected member of a class can be accessed from its derived class.

Q11) Number of threads required to run the below java program is:

```
public class ThreadExtended extends Thread {
    public void run()
    {System.out.println("\nThread is running now\n"); }
    public static void main(String[] args) {
        ThreadExtended threadE = new ThreadExtended();
        threadE.start();
    }
}
```

- a) 0
- c) 2**
- b) 1
- d) 3

Q12) Predict the output.

```
class Test{
    int count = 0;
    void A() throws Exception {
        try{
            count++;
            try{
                count++;
                try{
                    count++;
                    throw new Exception();
                }
                catch(Exception ex){
                    count++;
                    throw new Exception();
                }
            }
            catch(Exception ex){
                count++;
            }
        }
        catch(Exception ex){
            count++;
        }
    }
}

void display(){
    System.out.println(count);
}

public static void main(String[] args) throws
Exception{
    Test obj = new Test();
    obj.A();
    obj.display();
}
}
```

- a) 4
c) 6
- b) 5**
d) Compilation error

Q13) Which of these different types of checked exceptions could get generated during the deserialization process of an object?

- 1) IOException
- 2) ClassCastException
- 3) ClassNotFoundException
- 4) NullPointerException

- a) 1 and 3**
b) Only 3
c) 1, 3 and 4
d) 1,2,3 and 4

Q14) What will be the output?

```
public class Test {
    public static void main(String [] args) {
        try {
            badMethod();
            System.out.print("A");
        }
        catch (Exception ex) {
            System.out.print("B");
        }
        finally {
            System.out.print("C");
        }
        System.out.print("D");
    }
    public static void badMethod() {
        throw new Error();
    }
}
```

- a.) ABCD
b.) Compilation fails.
c.) C is printed before exiting with an error message.
d.) BC is printed before exiting with an error message.

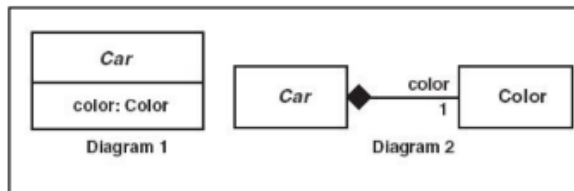
Q15) RectangleComparator is a class that implements Comparator interface to compare two rectangles based on its area. What kind of design pattern can be implemented for creating objects of the RectangleComparator class?

- a. **Singleton**
c. Facade
- b. Flyweight
d. Proxy

Q16) Which design pattern provides a single class that provides simplified methods required by the client and delegates calls to those methods?

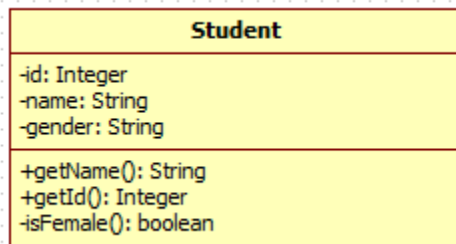
- a. Adapter pattern
b. Iterator pattern
c. Facade pattern
d. Prototype pattern

Q17) With reference to the following figure , what do Diagrams 1 and 2 illustrate?
Choose only one option.



- A. 1: An aggregation, 2: A composition.
- B. 1: An attribute, 2: An aggregation.
- C. 1: An aggregation, 2: An attribute.
- D. 1: An attribute, 2: A composition.**

Q18) Which of the following is/are true about the Student class as depicted in UML



- a) the attribute "id" is privately accessed**
- b) the attribute "id" is publicly accessed
- c) getName, getId and isFemale are all attributes in class Student
- d) getName accepts a parameter of type String
- e) isFemale() is a protected method

Q19) How many methods does Serializable have?

- a)0**
- b)1
- c)2
- d)3

Q20) You have thread T1, T2, and T3. How will you use it to ensure that thread T2 is run after T1 and thread T3 after T2?

- a) notify()
- b) sleep()
- c) wait()
- d) join()**

Q21) What will be the output of below code?

```
class Main {
    public static void main(String args[]) {
        try {
            throw 10;
        }
        catch(int e) {
            System.out.println("Got the Exception " + e);
        }
    }
}
```

- a)Got the Exception 10
- b)Got the Exception 0
- c)Compiler Error**
- d)None of the above

Q22) What will be the output of below code?

```
class BadThreads {
    static String message;

    class CorrectorThread extends Thread {
        public void run() {
            try {
                sleep(1000);
            } catch (InterruptedException e) {}
            message = "Mares do eat oats.";
        }
    }

    public static void main(String args[])throws
    InterruptedException {
        ((new BadThreads()).new CorrectorThread()).start();
        message = "Mares do not eat oats.";
        Thread.sleep(2000);
        System.out.println(message);
    }
}
```

- a. Mares do eat oats.**
- b. Mares do not eat oats.
- c. Compilation Error
- d. none of the above

Q23)What will be the Output?

```
class Animal {
    String name = "animal";
    String makeNoise() { return "generic
noise"; }
}
class Dog extends Animal {
    String name = "dog";
    String makeNoise() {
        return "bark";
    }
}
public class Test {
    public static void main(String[] args){
        Animal an = new Dog();
        System.out.println(an.name+"
"+an.makeNoise());
    }
}
```

- a.) animal generic noise
- b.) animal bark**
- c.) dog bark
- d.) dog generic noise

Q24) Which of the following is correct about JUnit execution procedure?

- a. Method annotated as @After executes for each test case but after the execution of test case
- b. In between method annotated as @Before and method annotated as @After each test case executes
- c. Both of the above**
- d. None of the above

Q25) Name the keyword that makes a variable belong to a class, rather than being defined for each instance of the class.

- a. final
- b. static**
- c. volatile
- d. Native