

```
public class Test {  
    public static List data = new ArrayList();  
  
    // insert code here  
    {  
        for (String x : strs) {  
            data.add(x);  
        }  
        return data;  
    }  
  
    public static void main(String[] args) {  
        String[] d = {"a", "b", "c"};  
        update(d);  
        for (String s : d) {  
            System.out.print(s + " ");  
        }  
    }  
}
```

Which code fragment, when inserted at // insert code here, enables the code to compile and print a b c?

- A. List update (String[] strs)
- B. Static ArrayListupdate(String [] strs)
- C. Static List update (String [] strs)
- D. Static void update (String[] strs)
- E. ArrayList static update(String [] strs)

Answer: E

Question No : 115

Given:

```
public class TestField {  
  
    int x;  
  
    int y;  
  
    public void doStuff(int x, int y) {  
  
        this.x = x;  
  
        y = this.y;
```

```
}

public void display() {
    System.out.print(x + " " + y + ":");
}

public static void main(String[] args) {
    TestField m1 = new TestField();
    m1.x = 100;
    m1.y = 200;
    TestField m2 = new TestField();
    m2.doStuff(m1.x, m1.y);
    m1.display();
    m2.display();
}
}
```

What is the result?

- A. 100 200 : 100 200
- B. 100 0 : 100 0 :
- C. 100 200 : 100 0 :
- D. 100 0 : 100 200 :

Answer: C

Question No : 116

Given the code fragment:

```
int b = 3;
if ( !(b > 3)) {
```

```
System.out.println("square ");
}
System.out.println("circle ");
}
System.out.println("...");
```

What is the result?

- A. square...
- B. circle...
- C. squarecircle...
- D. Compilation fails.

Answer: C

Question No : 117

Given:

```
7.  StringBuilder sb1 = new StringBuilder("Duke");
8.  String str1 = sb1.toString();
9.  // insert code here
10. System.out.print(str1 == str2);
```

Which code fragment, when inserted at line 9, enables the code to print true?

- A. String str2 = str1;
- B. String str2 = new string (str1);
- C. String str2 = sb1.toString();
- D. String str2 = "Duke";

- 1. 值相等
- 2. 内存地址相等

Answer: B

A

Question No : 118

Given the code fragment:

```
int[] array = {1, 2, 3, 4, 5};
```

And given the requirements:

- ✓ Process all the elements of the array in the order of entry.
- ✓ Process all the elements of the array in the reverse order of entry.
- ✓ Process alternating elements of the array in the order of entry.

Which two statements are true?

- A. Requirements 1, 2, and 3 can be implemented by using the enhanced for loop.
- B. Requirements 1, 2, and 3 can be implemented by using the standard for loop.
- C. Requirements 2 and 3 CANNOT be implemented by using the standard for loop.
- D. Requirement 1 can be implemented by using the enhanced for loop.
- E. Requirement 3 CANNOT be implemented by using either the enhanced for loop or the standard for loop.

Answer: D,E

Question No : 119

Given the code fragment:

```
// insert code here
```

```
arr[0] = new int[3];
```

```
arr[0][0] = 1;
```

```
arr[0][1] = 2;
```

```
arr[0][2] = 3;
```

```
arr[1] = new int[4];
```

```
arr[1][0] = 10;
```

```
arr[1][1] = 20;
```

```
arr[1][2] = 30;
```

```
arr[1][3] = 40;
```

Which two statements, when inserted independently at line // insert code here, enable the code to compile?

- A. int [] [] arr = null;
- B. int [] [] arr = new int [2];
- C. int [] [] arr = new int [2] [];
- D. int [] [] arr = new int [] [4];
- E. int [] [] arr = new int [2] [0];
- F. int [] [] arr = new int [0] [4];

Answer: C,E

Question No : 120

Given:

```
public class CharToStr {  
    public static void main(String[] args) {  
        String str1 = "Java";  
        char str2[] = { 'J', 'a', 'v', 'a' };  
        String str3 = null;  
        for (char c : str2) {  
            str3 = str3 + c;  
        }  
        if (str1.equals(str3))  
            System.out.print("Successful");  
        else  
            System.out.print("Unsuccessful");  
    }  
}
```

What is result?

str3="nullJava"
str1="Java"

- A. Successful
- B. Unsuccessful
- C. Compilation fails
- D. An exception is thrown at runtime

Answer: C

B

Question No : 121

Which two statements are true for a two-dimensional array?

- A. It is implemented as an array of the specified element type.
- B. Using a row by column convention, each row of a two-dimensional array must be of the same size.
- C. At declaration time, the number of elements of the array in each dimension must be specified.
- D. All methods of the class Object may be invoked on the two-dimensional array.

Answer: A,D

Question No : 122

Given:

Given:

```
class Caller {
    private void init() {
        System.out.println("Initialized");
    }

    public void start() {
        init();
        System.out.println("Started");
    }
}

public class TestCall {
    public static void main(String[] args) {
        Caller c = new Caller();
        c.start();
        c.init();
    }
}
```

What is the result?

- A. Initialized
Started
- B. Initialized
Started
Initialized
- C. Compilation fails

-
- D. An exception is thrown at runtime

Answer: B

Question No : 123

Given:

```
1. import java.util.ArrayList;
2. import java.util.List;
3.
4. public class Whizlabs{
5.
6.     public static void main(String[] args){
7.         List<Integer> list = new ArrayList<>();
8.         list.add(21); list.add(13);
9.         list.add(30); list.add(11);
10.        list.add(2);
11.        //insert here
12.        System.out.println(list);
13.    }
14. }
```

Which inserted at line 11, will provide the following output?

[21, 15, 11]

- A. list.removeIf(e > e%2 != 0);
- B. list.removeIf(e -> e%2 != 0);
- C. Ust.removeIf(e -> e%2 = 0);
- D. list.remove(e -> e%2 = 0);
- E. None of the above.

Answer: C

Explanation:

In output we can see that only odd numbers present, so we need to remove only even numbers to get expected output. From Java SE 8, there is new method call `removeIf` which takes predicate object and remove elements which satisfies predicate condition.

Predicate has functional method call `take` object and check if the given condition met or not, if met it returns true, otherwise false. Option C we have passed correct lambda expression to check whether the number is odd or even that matches to the functional method of predicate interface.

Option A is incorrect as it is invalid lambda expression. Option B is incorrect as it removes all odd numbers.

Option D is incorrect as there is no `remove` method that takes predicate as argument.

<https://docs.oracle.com/javase/8/docs/api/java/util/ArrayList.html>

Question No : 124

Given the following code:

```
int[] intArr = {15, 30, 45, 60, 75};  
intArr[2] = intArr[4];  
intArr[4] = 90;
```

What are the values of each element in `intArr` after this code has executed?

- A. 15, 60, 45, 90, 75
- B. 15, 90, 45, 90, 75
- C. 15, 30, 75, 60, 90
- D. 15, 30, 90, 60, 90
- E. 15, 4, 45, 60, 90

Answer: C

Question No : 125

Given the code from the `Greeting.java` file:

```
public class Greeting {  
    public static void main(String[] args) {  
        System.out.println("Hello " + args[0]);  
    }  
}
```

Which set of commands prints Hello Duke in the console?

- A) javac Greeting
java Greeting Duke
- B) javac Greeting.java Duke
java Greeting
- C) javac Greeting.java
java Greeting Duke
- D) javac Greeting.java
java Greeting.class Duke

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: C

| |
|--------------------------|
| Question No : 126 |
|--------------------------|

Given:

```

class SpecialException extends Exception {
    public SpecialException(String message) {
        super(message);
        System.out.println(message);
    }
}

public class ExceptionTest {
    public static void main(String[] args) {
        try {
            doSomething();
        } catch (SpecialException e) {
            System.out.println(e);
        }
    }
    static void doSomething() throws SpecialException {
        int[] ages = new int[4];
        ages[4] = 17;
        doSomethingElse();
    }
    static void doSomethingElse() throws SpecialException {
        throw new SpecialException("Thrown at end of doSomething() method");
    }
}

```

What will be the output?

- A) SpecialException: Thrown at end of doSomething() method
- B) Error in thread "main" java.lang.ArrayIndexOutOfBoundsException
- C) Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: 4
at ExceptionTest.doSomething(ExceptionTest.java:13)
at ExceptionTest.main(ExceptionTest.java:4)
- D) SpecialException: Thrown at end of doSomething() method
at ExceptionTest.doSomethingElse(ExceptionTest.java:16)
at ExceptionTest.doSomething(ExceptionTest.java:13)
at ExceptionTest.main(ExceptionTest.java:4)

- A. Option A
B. Option B
C. Option C
D. Option D

Answer: D

Question No : 127

Given the code fragment:

```
abstract class Planet {  
    protected void revolve() { //line n1  
    }  
  
    abstract void rotate(); //line n2  
}  
  
class Earth extends Planet {  
    void revolve() { //line n3  
    }  
  
    protected void rotate() { //line n4  
    }  
}
```

Which two modifications, made independently, enable the code to compile?

- A. Make the method at line n1 public.
- B. Make the method at line n2 public.
- C. Make the method at line n3 public.
- D. Make the method at line n3 protected.
- E. Make the method at line n4 public.

Answer: C,D

Question No : 128

```
Class StaticField {  
  
static int i = 7;  
  
public static void main(String[] args) {  
  
    StaticFied obj = new StaticField();  
  
    obj.i++;  
  
    StaticField.i++;  
  
    obj.i++;  
  
    System.out.println(StaticField.i + " " + obj.i);  
}  
}
```

What is the result?

- A. 10 10
- B. 8 9
- C. 9 8
- D. 7 10

Answer: A

Question No : 129

Given:

Given:

```
class Dog {
    Dog() {
        try {
            throw new Exception();
        } catch (Exception e) {
        }
    }
}

class Test {
    public static void main(String[] args) {
        Dog d1 = new Dog();
        Dog d2 = new Dog();
        Dog d3 = d2;
        // do complex stuff
    }
}
```

How many objects have been created when the line // do complex stuff is reached?

- A. Two
- B. Three
- C. Four
- D. Six

Answer: C

Question No : 130

Given:

MainTest.java:

```
public class MainTest {  
    public static void main(int[] args) {  
        System.out.println("int main " + args[0]);  
    }  
    public static void main(Object[] args) {  
        System.out.println("Object main " + args[0]);  
    }  
    public static void main(String[] args) {  
        System.out.println("String main " + args[0]);  
    }  
}
```

and commands:

```
javac MainTest.java  
java MainTest 1 2 3
```

What is the result?

- A. int main 1
- B. Object main 1
- C. String main 1
- D. Compilation fails
- E. An exception is thrown at runtime

Answer: C

Question No : 131

Which three statements are benefits of encapsulation?

- A. Allows a class implementation to change without changing the clients
- B. Protects confidential data from leaking out of the objects
- C. Prevents code from causing exceptions
- D. Enables the class implementation to protect its invariants
- E. Permits classes to be combined into the same package
- F. Enables multiple instances of the same class to be created safely

Answer: A,B,D

Question No : 132

Given the code in a file Traveler.java:

```
class Tours {
    public static void main(String[] args) {
        System.out.print("Happy Journey! " + args[1]);
    }
}

public class Traveler {
    public static void main(String[] args) {
        Tours.main(args);
    }
}
```

And the commands:

Javac Traveler.java

Java Traveler Java Duke

What is the result?

- A. Happy Journey! Duke
- B. Happy Journey! Java
- C. An exception is thrown at runtime
- D. The program fails to execute due to a runtime error

Answer: D

Question No : 133

Given:

```
public class MarkList {  
    int num;  
    public static void graceMarks(MarkList obj4) {  
        obj4.num += 10;  
    }  
    public static void main(String[] args) {  
        MarkList obj1 = new MarkList();  
        MarkList obj2 = obj1;  
        MarkList obj3 = null;  
        obj2.num = 60;  
        graceMarks(obj2);  
    }  
}
```

How many MarkList instances are created in memory at runtime?

- A. 1
- B. 2
- C. 3
- D. 4

Answer: A

Question No : 134

Given:

```
public class Calculator {  
    public static void main(String[] args) {  
        int num = 5;  
        int sum;  
  
        do {  
            sum += num;  
        } while ((num--) > 1);  
  
        System.out.println("The sum is " + sum + ".");  
    }  
}
```

What is the result?

- A. The sum is 2
- B. The sum is 14

-
- C. The sum is 15
 - D. The loop executes infinite times
 - E. Compilation fails

Answer: E

Question No : 135

Given:

```
public class X {  
    static int i;  
    int j;  
    public static void main(String[] args) {  
        X x1 = new X();  
        X x2 = new X();  
        x1.i = 3;  
        x1.j = 4;  
        x2.i = 5;  
        x2.j = 6;  
        System.out.println(  
            x1.i + " "+  
            x1.j + " "+  
            x2.i + " "+  
            x2.j);  
    }  
}
```

What is the result?

-
- A. 3 4 5 6
 - B. 3 4 3 6
 - C. 5 4 5 6
 - D. 3 6 4 6

Answer: C

Question No : 136

Given:

```
public class Equal {  
    public static void main(String[] args) {  
        String str1 = "Java";  
        String[] str2 = {"J","a","v","a"};  
        String str3 = "";  
        for (String str : str2) {  
            str3 = str3+str;  
        }  
        boolean b1 = (str1 == str3);  
        boolean b2 = (str1.equals(str3));  
        System.out.print(b1+", "+b2);  
    }  
}
```

What is the result?

- A. true, false
- B. false, true
- C. true, true
- D. false, false

Answer: B

Explanation: == strict equality.
equals compare state, not identity.

Question No : 137

Given:

```
class X {
    int x1, x2, x3;
}
class Y extends X {
    int y1;
    Y() {
        x1 = 1;
        x2 = 2;
        y1 = 10;
    }
}

class Z extends Y {
    int z1;
    Z() {
        x1 = 3;
        y1 = 20;
        z1 = 100;
    }
}

And,
public class Test3 {
    public static void main(String[] args) {
        Z obj = new Z();
        System.out.println(obj.x3 + ", " + obj.y1 + ", " + obj.z1);
    }
}
```

Which constructor initializes the variable x3?

- A. Only the default constructor of class X
- B. Only the no-argument constructor of class Y
- C. Only the no-argument constructor of class Z
- D. Only the default constructor of object class

Answer: C

Question No : 138

Given the classes:

* `AssertionError`

* `ArithmeticException`

-
- * `ArrayIndexOutOfBoundsException`
 - * `FileNotFoundException`
 - * `IllegalArgumentException`
 - * `IOError`
 - * `IOException`
 - * `NumberFormatException`
 - * `SQLException`

Which option lists only those classes that belong to the unchecked exception category?

- A.** `AssertionError`, `ArrayIndexOutOfBoundsException`, `ArithmaticException`
- B.** `AssertionError`, `IOError`, `IOException`
- C.** `ArithmaticException`, `FileNotFoundException`, `NumberFormatException`
- D.** `FileNotFoundException`, `IOException`, `SQLException`
- E.** `ArrayIndexOutOfBoundsException`, `IllegalArgumentException`, `FileNotFoundException`

Answer: A

Explanation: Not B: `IOError` and `IOException` are both checked errors.

Not C, not D, not E: `FileNotFoundException` is a checked error.

Note:

Checked exceptions:

- * represent invalid conditions in areas outside the immediate control of the program (invalid user input, database problems, network outages, absent files)
- * are subclasses of `Exception`
- * a method is obliged to establish a policy for all checked exceptions thrown by its implementation (either pass the checked exception further up the stack, or handle it somehow)

Note:

Unchecked exceptions:

- * represent defects in the program (bugs) - often invalid arguments passed to a non-private method. To quote from *The Java Programming Language*, by Gosling, Arnold, and Holmes: "Unchecked runtime exceptions represent conditions that, generally speaking, reflect errors in your program's logic and cannot be reasonably recovered from at run time."
- * are subclasses of `RuntimeException`, and are usually implemented using `IllegalArgumentException`, `NullPointerException`, or `IllegalStateException`
- * method is not obliged to establish a policy for the unchecked exceptions thrown by its

implementation (and they almost always do not do so)

Question No : 139

Given the code fragment:

```
public class App {  
    public static void main(String[] args) {  
        String str1 = "Java";  
        String str2 = new String("java");  
        //line n1  
        {  
            System.out.println("Equal");  
        } else {  
            System.out.println("Not Equal");  
        }  
    }  
}
```

Which code fragment, when inserted at line n1, enables the App class to print Equal?

- A) String str3 = str2;
 if (str1 == str3)
 - B) if (str1.equalsIgnoreCase(str2))
 - C) String str3 = str2;
 if (str1.equals(str3))
 - D) if (str1.toLowerCase() == str2.toLowerCase())
- A.** Option A
B. Option B
C. Option C
D. Option D

Answer: B

Question No : 140

Consider

Integer number = Integer.valueOf("808.1");

Which is true about the above statement?

只能传int或者string

- A. The value of the variable number will be 808.1
- B. The value of the variable number will be 808
- C. The value of the variable number will be 0.
- D. A NumberFormatException will be thrown.
- E. It will not compile.

Answer: D

Explanation:

The Integer class valueOf() method returns an Integer from given string. But we need to pass string which has correct format for integer otherwise it will throw a NumberFormatException. In this case we have passed string which is not an integer value (since what we passed is fractional number), so option D is correct.

Question No : 141

Given:

```
package p1;
public interface DoInterface {
    void m1(int n);                      // line n1
    public void m2(int n);
}

package p3;
import p1.DoInterface;
public class DoClass implements DoInterface{
    int x1,x2;
    DoClass(){
        this.x1 = 0;
        this.x2 = 10;
    }
    public void m1(int p1) { x1+=p1; System.out.println(x1); } // line n2
    public void m2(int p1) { x2+=p1; System.out.println(x2); }
}

package p2;
import p1.*;
import p3.*;
class Test {
    public static void main(String[] args){           // line n3
        DoInterface doi= new DoClass();
        doi.method1(100);
        doi.method2(200);
    }
}
```

What is the result?

- A. 100
- 210
- B. Compilation fails due to an error in line n1
- C. Compilation fails due to an error at line n2
- D. Compilation fails due to an error at line n3

Answer: C

Question No : 142

Which two are benefits of polymorphism?

- A. Faster code at runtime
- B. More efficient code at runtime
- C. More dynamic code at runtime
- D. More flexible and reusable code
- E. Code that is protected from extension by other classes

Answer: C,D

Question No : 143

Given:

```
class Product {  
    double price;  
}  
  
public class Test {  
    public void updatePrice(Product product, double price) {  
        price = price * 2;  
        product.price = product.price + price;  
    }  
    public static void main(String[] args) {  
        Product prt = new Product();  
        prt.price = 200;  
        double newPrice = 100;  
  
        Test t = new Test();  
        t.updatePrice(prt, newPrice);  
        System.out.println(prt.price + " : " + newPrice);  
    }  
}
```

What is the result?

- A. 200.0 : 100.0
- B. 400.0 : 200.0
- C. 400.0 : 100.0
- D. Compilation fails.

Answer: C

Question No : 144

Given:

```
class Base {  
    // insert code here  
}  
  
public class Derived extends Base{  
    public static void main(String[] args) {  
        Derived obj = new Derived();  
        obj.setNum(3);  
        System.out.println("Square = " + obj.getNum() * obj.getNum());  
    }  
}
```

Which two options, when inserted independently inside class Base, ensure that the class is being properly encapsulated and allow the program to execute and print the square of the number?

- A. private int num; public int getNum() { return num; }public void setNum(int num) { this.num = num;}
- B. public int num; protected public int getNum() { return num; }protected public void setNum(int num) { this.num = num;}
- C. private int num;public int getNum() {return num;} private void setNum(int num) { this.num = num;}

D. protected int num; public int getNum() { return num; } public void setNum(int num) { this.num = num;}

E. protected int num; private int getNum() { return num; } public void setNum(int num) { this.num = num;}

Answer: A,D

Explanation:

Incorrect:

Not B: illegal combination of modifiers: protected and public

not C: setNum method cannot be private.

not E: getNum method cannot be private.

Question No : 145

You are asked to develop a program for a shopping application, and you are given the following information:

- ↗ The application must contain the classes Toy, EduToy, and consToy. The Toy class is the superclass of the other two classes.
- ↗ The int calculatePrice (Toy t) method calculates the price of a toy.
- ↗ The void printToy (Toy t) method prints the details of a toy.

Which definition of the Toy class adds a valid layer of abstraction to the class hierarchy?

- A)

```
public abstract class Toy{
    public abstract int calculatePrice(Toy t);
    public void printToy(Toy t) { /* code goes here */ }
}
```
- B)

```
public abstract class Toy {
    public int calculatePrice(Toy t) ;
    public void printToy(Toy t) ;
}
```
- C)

```
public abstract class Toy {
    public int calculatePrice(Toy t);
    public final void printToy(Toy t){ /* code goes here */ }
}
```
- D)

```
public abstract class Toy {
    public abstract int calculatePrice(Toy t) { /* code goes here */ }
    public abstract void printToy(Toy t) { /* code goes here */ }
}
```

A. Option A

B. Option B

C. Option C

D. Option D

Answer: A

Question No : 146

Given the code fragment:

```
public static void main(String[] args) {  
    List<String> names = new ArrayList<>();  
    names.add("Robb");  
    names.add("Bran");  
    names.add("Rick");  
    names.add("Bran");  
  
    if (names.remove("Bran")) {  
        names.remove("Jon");  
    }  
    System.out.println(names);  
}
```

What is the result?

- A. [Robb, Rick, Bran]
- B. [Robb, Rick]
- C. [Robb, Bran, Rick, Bran]
- D. An exception is thrown at runtime.

Answer: A

Question No : 147

Given the code fragment:

```
for (int ii = 0; ii < 3; ii++) {  
  
    int count = 0;  
  
    for (int jj = 3; jj > 0; jj--) {
```

```
if (ii == jj) {  
    ++count;  
    break;  
}  
}  
  
System.out.print(count);  
  
continue;  
}
```

What is the result?

- A. 011
- B. 012
- C. 123
- D. 000

Answer: A

Question No : 148

Which statement will empty the contents of a StringBuilder variable named sb?

- A. sb.deleteAll();
- B. sb.delete(0, sb.size());
- C. sb.delete(0, sb.length());
- D. sb.removeAll();

Answer: C

Question No : 149

Given:

```
class Base {
```

```
public static void main(String[] args) {  
    System.out.println("Base " + args[2]);  
}  
}  
  
public class Sub extends Base{  
  
    public static void main(String[] args) {  
        System.out.println("Overriden " + args[1]);  
    }  
}
```

And the commands:

```
javac Sub.java  
java Sub 10 20 30
```

What is the result?

- A.** Base 30
- B.** Overridden 20
- C.** Overridden 20
Base 30
- D.** Base 30
Overridden 20

Answer: B

Question No : 150

Which of the following can fill in the blank in this code to make it compile?

```
interface CanFly{  
    String type = "A";  
    void fly();  
  
    _____ String getType(){  
        return type;  
    }  
}
```

- A. abstract
- B. public
- C. default
- D. It will not compile with any as interfaces cannot have non abstract methods.
- E. It will compile without filling the blank.

Answer: C

Explanation:

From Java SE 8, we can use static and/or default methods in interfaces, but they should be non abstract methods. SO in this case using default in blank is completely legal. Hence option C is correct.

Option A is incorrect as given method is not abstract, so can't use abstract there.

Options B and E are incorrect as we can't have non abstract method interface if they are not default or static.

<https://docs.oracle.com/javase/tutorial/java/lambda/defaultmethods.html>

Question No : 151

Given the code fragment:

```
float x = 22.00f % 3.00f;
```

```
int y = 22 % 3;  
System.out.print(x + ", " + y);
```

What is the result?

- A. 1.0, 1
- B. 1.0f, 1
- C. 7.33, 7
- D. Compilation fails
- E. An exception is thrown at runtime

Answer: A

Question No : 152

Given:

```
public class Access {  
    private int x = 0;  
    private int y = 0;  
  
    public static void main(String[] args) {  
        Access accApp = new Access();  
        accApp.printThis(1, 2);  
        accApp.printThat(3, 4);  
    }  
  
    public void printThis(int x, int y) {  
        x = x;  
        y = y;  
        System.out.println("x:" + this.x + " y:" + this.y);  
    }  
  
    public void printThat(int x, int y) {  
        this.x = x;  
        this.y = y;  
        System.out.println("x:" + this.x + " y:" + this.y);  
    }  
}
```

What is the result?

- A. x: 1 y: 2
- B. 3 y: 4
- C. x: 0 y: 0
- D. 3 y: 4
- E. x: 1 y: 2
- F. 0 y: 0

G. x: 0 y: 0

H. 0 y: 0

Answer: C

Question No : 153

Given the following class declarations:

- public abstract class Animal
- public interface Hunter
- public class Cat extends Animal implements Hunter
- public class Tiger extends Cat

Which answer fails to compile?

- A) ArrayList<Animal> myList = new ArrayList<>();
myList.add(new Tiger());
- B) ArrayList<Hunter> myList = new ArrayList<>();
myList.add(new Cat());
- C) ArrayList<Hunter> myList = new ArrayList<>();
myList.add(new Tiger());
- D) ArrayList<Tiger> myList = new ArrayList<>();
myList.add(new Cat());
- E) ArrayList<Animal> myList = new ArrayList<>();
myList.add(new Cat());

- A.** Option A
- B.** Option B
- C.** Option C
- D.** Option D
- E.** Option E

Answer: E

Explanation: Look at the right side of the declaration ArrayList() rather than ArrayList

Question No : 154

Given:

```
1. public class TestLoop {  
2.     public static void main(String[] args) {  
3.         float myarray[] = {10.20f, 20.30f, 30.40f, 50.60f};  
4.         int index = 0;  
5.         boolean isFound = false;  
6.         float key = 30.40f;  
7.         // insert code here  
8.         System.out.println(isFound);  
9.     }  
10. }
```

Which code fragment, when inserted at line 7, enables the code print true?

C A) while (key == myarray[index++]) {
 isFound = true;
}

C B) while (index <= 4) {
 if (key == myarray[index]) {
 index++;
 isFound = true;
 break;
 }
}

C C) while (index++ < 5) {
 if (key == myarray[index]) {
 isFound = true;
 }
}

C D) while (index < 5) {
 if (key == myarray[index]) {
 isFound = true;
 break;
 }
 index++;
}

- A. Option A
B. Option B
C. Option C
D. Option D

Answer: A

Question No : 155

Given:

```
public class Test {  
    public static void main(String[] args) {  
        int ax = 10, az = 30;  
        int aw = 1, ay = 1;  
        try {  
            aw = ax % 2;  
            ay = az / aw;  
        } catch (ArithmetricException e1) {  
            System.out.println("Invalid Divisor");  
        } catch (Exception e2) {  
            aw = 1;  
            System.out.println("Divisor Changed");  
        }  
        ay = az /aw; // Line 14  
        System.out.println("Succesful Division " + ay);  
    }  
}
```

What is the result?

- A. Invalid Divisor
Divisor Changed
Successful Division 30
- B. Invalid Divisor
Successful Division 30

C. Invalid Divisor

Exception in thread "main" java.lang.ArithmetricException: / by zero
at test.Teagle.main(Teagle.java:14)

D. Invalid Divisor

Exception in thread "main" java.lang.ArithmetricException: / by zero
at test.Teagle.main(Teagle.java:14)

Successful Division 1

Answer: C

Question No : 156

Given:

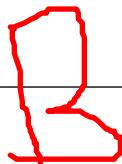
```
1. public class Whizlabs {  
2.  
3.     public static void main(String[] args) {  
4.         String s = "A";  
5.  
6.         switch (s) {  
7.             case "a":  
8.                 System.out.print("simaple A ");  
9.             default:  
10.                 System.out.print("default ");  
11.             case 'A':  
12.                 System.out.print("Capital A ");  
13.             }  
14.     }  
15. }
```



What is the result?

- A. simaple A**
 - B. Capital A**
 - C. simaple A default Capital A**
 - D. simaple A default**
 - E. Compilation fails.**
-

Answer: C



Explanation:

Here we have to use two ternary operators combined. SO first we can use to check first condition which is $x > 10$, as follows;

$x > 10 ? > :$ (when condition false) Now we have to use another to check if $x < 10$ as follows;

$x < 10 ? V : =$ We can combine these two by putting last ternary statement in the false position of first ternary statement as follows;

$x > 10 ? > : x < 10 ? < : =$

<https://docs.oracle.com/javase/tutorial/java/nutsandbolts/if.html>

Question No : 157

Given:

```
1. public class Whizlabs{  
2.  
3.     public static void main(String[] args){  
4.         try{  
5.             Double number = Double.valueOf("120D");  
6.         }catch(NumberFormatException ex){  
7.         }  
8.         System.out.println(number);  
9.     }  
10. }
```

What is the result?

- A. 120
- B. 120D
- C. A NumberFormatException will be thrown.
- D. Compilation fails due to error at line 5.
- E. Compilation fails due to error at line 8.

number未定义

Answer: E

Explanation:

At line 5, we have created a wrapper object of double by passing 120D, which is

convertible to a Double, so there won't be any exception there. But if you check carefully, you can see the variable number is declared inside try block, so the scope of the variable number is limited to that block, so trying to access it outside causes a compile time error.
<https://docs.oracle.com/javase/tutorial/java/nutsandbolts/variables.html>

Question No : 158

Given the code fragment:

```
class Student {  
    int rollnumber;  
    String name;  
    List courses = new ArrayList();  
    // insert code here  
    public String toString() {  
        return rollnumber + " : " + name + " : " + courses;  
    }  
}
```

And,

```
public class Test {  
    public static void main(String[] args) {  
        List cs = newArrayList();  
        cs.add("Java");  
        cs.add("C");  
        Student s = new Student(123,"Fred", cs);  
        System.out.println(s);
```

```
}
```

```
}
```

Which code fragment, when inserted at line // insert code here, enables class Test to print 123 : Fred : [Java, C]?

- A. private Student(int i, String name, List cs) {
/* initialization code goes here */
}
- B. public void Student(int i, String name, List cs) {
/* initialization code goes here */
}
- C. Student(int i, String name, List cs) {
/* initialization code goes here */
}
- D. Student(int i, String name, ArrayList cs) {
/* initialization code goes here */
}

Answer: C

Explanation:

Incorrect:

Not A: Student has private access line: Student s = new Student(123,"Fred", cs);

Not D: Cannot be applied to given types. Line: Student s = new Student(123,"Fred", cs);

Question No : 159

Given:

```
public class SampleClass {  
  
    public static void main(String[] args) {  
  
        AnotherSampleClass asc = new AnotherSampleClass(); SampleClass sc = new  
        SampleClass();  
  
        sc = asc;  
  
        System.out.println("sc: " + sc.getClass());
```

```
System.out.println("asc: " + asc.getClass());  
}  
  
class AnotherSampleClass extends SampleClass {  
}
```

What is the result?

- A. sc: class Object
asc: class AnotherSampleClass
- B. sc: class SampleClass
asc: class AnotherSampleClass
- C. sc: class AnotherSampleClass
asc: class SampleClass
- D. sc: class AnotherSampleClass
asc: class AnotherSampleClass

Answer: D

Question No : 160

Given the code fragment:

```
9. int a = -10;  
10. int b = 17;  
11. int c = expression1;  
12. int d = expression2;  
13. c++;  
14. d--;  
15. System.out.print(c + ", " + d);
```

What could **expression1** and **expression2** be, respectively, in order to produce output –8, 16?

- A. + +a, - -b
- B. + +a, b- -
- C. A+ +, -- b
- D. A + +, b - -

Answer: D

Question No : 161

Given the code fragment:

```
public static void main(String[] args) {  
    Short s1 = 200;  
    Integer s2 = 400;  
    Long s3 = (long) s1 + s2;           //line n1  
    String s4 = (String) (s3 * s2);     //line n2  
    System.out.println("Sum is " + s4);  
}
```

What is the result?

- A. Sum is 600
- B. Compilation fails at line n1.
- C. Compilation fails at line n2.
- D. A ClassCastException is thrown at line n1.
- E. A ClassCastException is thrown at line n2.

Answer: C

Question No : 162

Given the following four Java file definitions:

```
// Foo.java  
  
package facades;  
  
public interface Foo {}  
  
// Boo.java  
  
package facades;  
  
public interface Boo extends Foo {}
```

```
// Woofy.java  
  
package org.domain  
  
// line n1  
  
public class Woofy implements Boo, Foo { }
```

```
// Test.java  
  
package.org;  
  
public class Test {  
  
    public static void main(String[] args) {  
  
        Foo obj=new Woofy();
```

Which set modifications enable the code to compile and run?

- A. At line n1, Insert: import facades;At line n2, insert:import facades;import org.domain;
- B. At line n1, Insert: import facades.*;At line n2, insert:import facades;import org.*;
- C. At line n1, Insert: import facades.*;At line n2, insert:import facades.Boo;import org.*;
- D. At line n1, Insert: import facades.Foo, Boo;At line n2, insert:import org.domain.Woofy;
- E. At line n1, Insert: import facades.*;At line n2, insert:import facades;import org.domain.Woofy;

Answer: E

Question No : 163

Given:

```
public class MyFor3 {  
    public static void main(String[] args) {  
        int[] xx = null;  
        for (int ii : xx) {  
            System.out.println(ii);  
        }  
    }  
}
```

What is the result?

- A. Null
- B. Compilation fails
- C. An exception is thrown at runtime
- D. 0

Answer: C

Question No : 164

What is the proper way to define a method that takes two int values and returns their sum as an int value?

- A. int sum(int first, int second) { first + second; }
- B. int sum(int first, second) { return first + second; }
- C. sum(int first, int second) { return first + second; }
- D. int sum(int first, int second) { return first + second; }
- E. void sum (int first, int second) { return first + second; }

Answer: D

Question No : 165

Given:

```
1. public class Whizlabs {
2.     public static void main(String[] args) {
3.         int sum = 0;
4.
5.         for(int x = 0;x<=10;x++)
6.             sum += x;
7.         System.out.print("Sum for 0 to " + x);
8.         System.out.println(" = " + sum);
9.     }
10. }
```

Which is true?

- A. Sum for 0 to 0 = 55
- B. Sum for 0 to 10 = 55
- C. Compilation fails due to error on line 6.
- D. Compilation fails due to error on line 7.
- E. An Exception is thrown at the runtime.

Answer: D

Explanation:

Loop variables scope limited to that enclosing loop. So in this case, the scope of the loop variable x declared at line 5, limited to that for loop. Trying to access that variable at line 7, which is out of scope of the variable x, causes a compile time error. So compilation fails due to error at line 7. Hence option D is correct.

Options A and B are incorrect, since code fails to compile.

Reference: <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/variables.html>

Question No : 166

Given:

```
public class Test {  
    public static void main(String[] args) {  
        Test ts = new Test();  
        System.out.print(isAvailable + " ");  
        isAvailable= ts.doStuff();  
        System.out.println(isAvailable);  
    }  
    public static boolean doStuff() {  
        return !isAvailable;  
    }  
    static boolean isAvailable = false;  
}
```

What is the result?

- A. true true
- B. true false
- C. false true

-
- D. false false
 - E. Compilation fails

Answer: E

Question No : 167

Given the following main method:

```
public static void main(String[] args) {  
    int num = 5;  
    do {  
        System.out.print(num-- + " ");  
    } while (num == 0);  
}
```

What is the result?

- A. 5 4 3 2 1 0
- B. 5 4 3 2 1
- C. 4 2 1
- D. 5
- E. Nothing is printed

当while中条件符合才会继续循环，否则将会跳出循环

Answer: D

Explanation:

Loop will run only once and after that num == 0 will break it

After first cycle of the loop.

Question No : 168

Given the definitions of the MyString class and the Test class:

MyString.java:

```
package p1;
class MyString {
    String msg;
    MyString(String msg) {
        this.msg = msg;
    }
}
```

Test.java:

```
package p1;
public class Test {
    public static void main(String[] args) {
        System.out.println("Hello " + new StringBuilder("Java SE 8"));
        System.out.println("Hello " + new MyString("Java SE 8"));
    }
}
```

What is the result?

- A) Hello Java SE 8
Hello Java SE 8
- B) Hello java.lang.StringBuilder@<<hashcode1>>
Hello p1.MyString@<<hashcode2>>
- C) Hello Java SE 8
Hello p1.MyString@<<hashcode>>
- D) Compilation fails at the Test class.

- A. Option A
B. Option B
C. Option C
D. Option D

Answer: C

Question No : 169

Given the code fragment:

```
System.out.println( 28 + 5 <= 4 + 29 );
System.out.println( ( 28 + 5 ) <= ( 4 + 29 ) );
```

What is the result?

- A. 28false29
true
- B. 285 < 429
true
- C. true
true
- D. compilation fails

Answer: C

Question No : 170

Given:

```
public class MyClass {  
    public static void main(String[] args) {  
        String s = " Java Duke ";  
        int len = s.trim().length();  
        System.out.print(len);  
    }  
}
```

What is the result?

- A. 8
- B. 9
- C. 11
- D. 10
- E. Compilation fails

Answer: B

Explanation: Java - String trim() Method

This method returns a copy of the string, with leading and trailing whitespace omitted.

Question No : 171

Given:

```
public class App {  
    public static void main(String[] args) {  
        Boolean[] bool = new Boolean[2];  
  
        bool[0] = new Boolean(Boolean.parseBoolean("true"));  
        bool[1] = new Boolean(null);  
  
        System.out.println(bool[0] + " " + bool[1]);  
    }  
}
```

What is the result?

- A. True false
- B. True null
- C. Compilation fails
- D. A NullPointerException is thrown at runtime

Answer: A

Question No : 172

Which statement is/are true?

- I. Default constructor only contains "super();" call.
- II. We can't use any access modifier with a constructor.
- III. A constructor should not have a return type.

- A. Only I.
- B. Only II.
- C. Only I and II.
- D. Only I and III.

E. AII

Answer: D

Explanation:

Statement I is correct as the default constructor only contains super() call

Statement II is incorrect as we can use any access modifier with a constructor.

Statement III is correct as constructor can't have return type, even void.

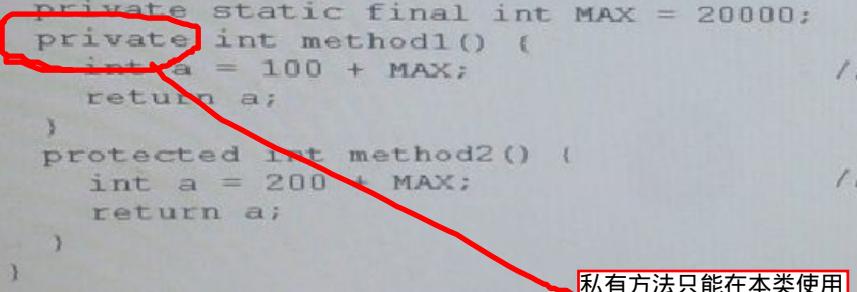
So option D is correct.

<https://docs.oracle.com/javase/tutorial/java/javaOO/constructors.html>

Question No : 173

Given the fragments:

```
public class TestA extends Root {
    public static void main(String[] args) {
        Root r = new TestA();
        System.out.println(r.method1());          // line n1
        System.out.println(r.method2());          // line n2
    }
}
class Root {
    private static final int MAX = 20000;
    private int method1() {
        int a = 100 + MAX;                      // line n3
        return a;
    }
    protected int method2() {                  // line n4
        int a = 200 + MAX;
        return a;
    }
}
```



私有方法只能在本类使用

Which line causes a compilation error?

- A. Line n1
- B. Line n2
- C. Line n3
- D. Line n4

Answer: A

Question No : 174

Consider following method

```
default void print(){
```

```
}
```

Which statement is true?

- A. This method is invalid.
- B. This method can be used only in an interface.
- C. This method can return anything.
- D. This method can be used only in an interface or an abstract class.
- E. None of above.

Answer: B

Explanation:

Given method is declared as default method so we can use it only inside an interface.

Hence option B is correct and option D is incorrect.

Option A is incorrect as it is valid method. Option C is incorrect as return type is void, which means we can't return anything.

Question No : 175

Given:

```
package p1;  
  
public class Test {  
  
    static double dvalue;  
  
    static Test ref;  
  
    public static void main(String[] args) {  
  
        System.out.println(ref);  
    }  
}
```

```
System.out.println(dvalue);
}
}
```

What is the result?

- A. p1.Test.class
- 0.0
- B. <the summary address referenced by ref>
- 0.000000
- C. Null
- 0.0
- D. Compilation fails
- E. A NullPointerException is thrown at runtime

Answer: C

Question No : 176

Given:

```
public class App {
    String myStr = "7007";
    public void doStuff(String str) {
        int myNum = 0;
        try {
            String myStr = str;
            myNum = Integer.parseInt(myStr);
        } catch (NumberFormatException ne) {
            System.err.println("Error");
        }
        System.out.println(
            "myStr: " + myStr + ", myNum: " + myNum);
    }
}
public static void main(String[] args) {
    App obj = new App();
    obj.doStuff("9009");
}
```

注意作用域

What is the result?

- A. myStr: 9009, myNum: 9009
- B. myStr: 7007, myNum: 7007
- C. myStr: 7007, myNum: 9009
- D. Compilation fails

Answer: C

Question No : 177

```
int i, j=0;  
i = (3* 2 +4 +5 ) ;  
j = (3 * ((2+4) + 5));  
System.out.println("i:"+ i + "\nj":+j);
```

What is the result?

- A. i: 16
 j: 33
- B. i: 15
 j: 33
- C. i: 33
 j: 23
- D. i: 15
 j: 23

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: B

Question No : 178

Given:

```
interface Pet { }

class Dog implements Pet { }

public class Beagle extends Dog{ }
```

Which three are valid?

- A. Pet a = new Dog();
- B. Pet b = new Pet();
- C. Dog f = new Pet();
- D. Dog d = new Beagle();
- E. Pet e = new Beagle();
- F. Beagle c = new Dog();

Answer: A,D,E

Explanation:

Incorrect:

Not B, not C: Pet is abstract, cannot be instantiated.

Not F: incompatible type. Required Beagle, found Dog.

Question No : 179

Given:

```
Test.java

public class Test {
    public static void main(String[] args) {
        Integer num = Integer.parseInt(args[1]);
        System.out.println("Number is : " + num);
    }
}
```

And the commands:

Javac Test.java

Java Test 12345

What is the result?

- A. Number us : 12345
- B. A NullPointerException is thrown at runtime
- C. A NumberFormatException is thrown at runtime
- D. AnArrayIndexOutOfBoundsException is thrown at runtime.

Answer: A

Question No : 180

Given:

```
public class ColorTest {  
    public static void main(String[] args) {  
        String[] colors = {"red", "blue", "green", "yellow", "maroon", "cyan"};  
        int count = 0;  
        for (String c : colors) {  
            if (count >= 4) {  
                break;  
            }  
            else {  
                continue;  
            }  
            if (c.length() >= 4) {  
                colors[count] = c.substring(0,3);  
            }  
        }  
    }  
}
```

```
}

count++;

}

System.out.println(colors[count]);

}

}
```

What is the result?

- A. Yellow
- B. Maroon
- C. Compilation fails
- D. A StringIndexOutOfBoundsException is thrown at runtime.

Answer: C

Explanation: The line, if (c.length() >= 4) {}, is never reached. This causes a compilation error.

Note: The continue statement skips the current iteration of a for, while , or do-while loop. An unlabeled break statement terminates the innermost switch, for, while, or do-while statement, but a labeled break terminates an outer statement.

Question No : 181

Given:

```
class Jump {
    static String args[] = {"lazy", "lion", "is", "always"};
    public static void main(String[] args) {
        System.out.println(
            args[1] + " " + args[2] + " " + args[3] + " jumping");
    }
}
```

And the commands:

跟定义的这个args[]没有毛关系

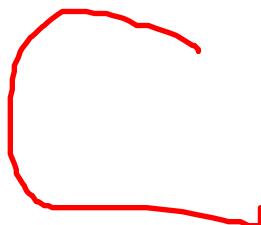
Javac Jump.java

Java Jump crazy elephant is always

What is the result?

- A. Lazy lion is jumping
- B. Lion is always jumping
- C. Crazy elephant is jumping
- D. Elephant is always jumping
- E. Compilation fails

Answer: B



Question No : 182

Given:

```
public class Vowel {  
    private char var;  
    public static void main(String[] args) {  
        char var1 = 'a';  
        char var2 = var1;  
        var2 = 'e';  
  
        Vowel obj1 = new Vowel();  
        Vowel obj2 = obj1;  
        obj1.var = 'i';  
        obj2.var = 'o';  
  
        System.out.println(var1 + ", " + var2);  
        System.out.print(obj1.var + ", " + obj2.var);  
    }  
}
```

- A. a, e
i, o
- B. a, e
o, o
- C. e, e
l, o
- D. e, e
o, o

Answer: B

Question No : 183

Which code fragment cause a compilation error?

- A. float flt = 100F;
- B. float flt = (float) 1_11.00;
- C. float flt = 100;
- D. double y1 = 203.22;
floatflt = y1
- E. int y2 = 100;
floatflt = (float) y2;

Answer: B

Question No : 184

Given:

```
class MarksOutOfBoundsException extends IndexOutOfBoundsException {}  
  
public class GradingProcess {  
  
    void verify(int marks) throws IndexOutOfBoundsException {  
        if (marks > 100) {  
            throw new MarksOutOfBoundsException();  
        }  
        if (marks > 50) {  
            System.out.print("Pass");  
        } else {  
            System.out.print("Fail");  
        }  
    }  
  
    public static void main(String[] args) {
```

```
int marks = Integer.parseInt(args[2]);
try {
    new GradingProcess().verify(marks);
} catch(Exception e) {
    System.out.print(e.getClass());
}
}
```

And the command line invocation:

Java grading process 89 50 104

What is the result?

- A. Pass
- B. Fail
- C. Class MarketOutOfBoundsException
- D. Class IndexOutOfBoundsException
- E. Class Exception

Answer: C

Explanation: The value 104 will cause a MarketOutOfBoundsException

Question No : 185

Given:

```

class X {
    static int i;
    int j;
    public static void main(String[] args) {
        X x1 = new X();
        X x2 = new X();
        x1.i = 3;
        x1.j = 4;
        x2.i = 5;
        x2.j = 6;
        System.out.println(
            x1.i + " " +
            x1.j + " " +
            x2.i + " " +
            x2.j);
    }
}

```

What is the result?

- A.** 3 4 5 6
- B.** 3 4 3 6
- C.** 5 4 5 6
- D.** 3 6 4 6

Answer: C

Question No : 186

Given:

```

public class Test2 {
    public static void doChange(int[] arr) {
        for(int pos = 0; pos < arr.length; pos++) {
            arr[pos] = arr[pos] + 1;
        }
    }
    public static void main(String[] args) {
        int[] arr = {10, 20, 30};
        doChange(arr);
        for(int x: arr) {
            System.out.print(x + ", ");
        }
        doChange(arr[0], arr[1], arr[2]);
        System.out.print(arr[0] + ", " + arr[1] + ", " + arr[2]);
    }
}

```

What is the result?

- A. 11, 21, 31, 11, 21, 31
- B. 11, 21, 31, 12, 22, 32
- C. 12, 22, 32, 12, 22, 32
- D. 10, 20, 30, 10, 20, 30

Answer: D

Question No : 187

What is the name of the Java concept that uses access modifiers to protect variables and hide them within a class?

- A. Encapsulation
- B. Inheritance
- C. Abstraction
- D. Instantiation
- E. Polymorphism

Answer: A

Question No : 188

```
int [] array = {1,2,3,4,5};
```

```
for (int i: array) {
```

```
if ( i < 2) {
```

```
keyword1 ;
```

```
}
```

```
System.out.println(i);
```

```
if ( i == 3) {
```

```
keyword2 ;
```

```
}
```

What should keyword1 and keyword2 be respectively, in order to produce output 2345?

- A. continue, break
- B. break, break
- C. break, continue
- D. continue, continue

Answer: D

Question No : 189

Given the code fragments:

Person.java:

```
public class Person {  
    String name;  
    int age;  
  
    public Person(String n, int a) {  
        name = n;  
        age = a;  
    }  
  
    public String getName() {  
        return name;  
    }  
  
    public int getAge() {  
        return age;  
    }  
}
```

Test.java:

```
public static void checkAge(List<Person> list, Predicate<Person> predicate) {  
    for (Person p : list) {  
        if (predicate.test(p)) {  
            System.out.println(p.name + " ");  
        }  
    }  
}  
  
public static void main(String[] args) {  
    List<Person> iList = Arrays.asList(new Person("Hank", 45),  
                                         new Person("Charlie", 40),  
                                         new Person("Smith", 38));  
    //line n1  
}
```

Which code fragment, when inserted at line n1, enables the code to print Hank?

- A. checkAge (iList, () -> p. get Age () > 40);
- B. checkAge(iList, Person p -> p.getAge() > 40);
- C. checkAge (iList, p -> p.getAge () > 40);
- D. checkAge(iList, (Person p) -> { p.getAge() > 40; });

Answer: C

Question No : 190

Given:

```
package p1;
public class Acc {
    int p;
    private int q;
    protected int r;
    public int s;
}
```

Test.java:

```
package p2;
import p1.Acc;
public class Test extends Acc {
    public static void main(String[] args) {
        Acc obj = new Test();
    }
}
```

Which statement is true?

- A. Both p and s are accessible by obj.
- B. Only s is accessible by obj.
- C. Both r and s are accessible by obj.
- D. p, r, and s are accessible by obj.

Answer: B

Question No : 191

Given:

```
public class TestLoop {  
    public static void main(String[] args) {  
        int array[] = {0, 1, 2, 3, 4};  
        int key = 3;  
        for (int pos = 0; pos < array.length; ++pos) {  
            if (array[pos] == key) {  
                break;  
            }  
        }  
        System.out.print("Found " + key + " at " + pos);  
    }  
}
```

What is the result?

- A. Found 3 at 2
- B. Found 3 at 3
- C. Compilation fails
- D. An exception is thrown at runtime

Answer: C

Explanation: The following line does not compile:

```
System.out.print("Found " + key + " at " + pos);
```

The variable pos is undefined at this line, as its scope is only valid in the for loop.
Any variables created inside of a loop are LOCAL TO THE LOOP.

| |
|--------------------------|
| Question No : 192 |
|--------------------------|

Given:

```
public class TestTry {
    public static void main(String[] args) {
        StringBuilder message = new StringBuilder("hello java!");
        int pos =0;
        try {
            for ( pos = 0; pos < 12; pos++) {
                switch (message.charAt(pos)) {
                    case 'a':
                    case 'e':
                    case 'o':
                        String uc=Character.toString(message.charAt(pos)).toUpperCase();
                        message.replace(pos, pos+1, uc);
                }
            }
        } catch (Exception e) {
            System.out.println("Out of limits");
        }
        System.out.println(message);
    }
}
```

What is the result?

- A. hEllOjAvA!
- B. Hello java!
- C. Out of limits
- hEllOjAvA!
- D. Out of limits

Answer: C

Question No : 193

Which two statements are true for a two-dimensional array of primitive data type?

- A. It cannot contain elements of different types.
- B. The length of each dimension must be the same.
- C. At the declaration time, the number of elements of the array in each dimension must be specified.
- D. All methods of the class object may be invoked on the two-dimensional array.

Answer: C,D

Explanation: <http://stackoverflow.com/questions/12806739/is-an-array-a-primitive-type-or-an-object-or-something-else-entirely>

Question No : 194

Given:

```
public class Natural {  
    private int i;  
  
    void disp() {  
        while (i <= 5) {  
            for (int i=1; i <=5;) {  
                System.out.print(i + " ");  
                i++;  
            }  
            i++;  
        }  
    }  
  
    public static void main(String[] args) {  
        new Natural().disp();  
    }  
}
```

What is the result?

- A. Prints 1 2 3 4 5 once
- B. Prints 1 3 5 once
- C. Prints 1 2 3 4 5 five times
- D. Prints 1 2 3 4 5 six times
- E. Compilation fails

Answer: D

Explanation: 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5

Question No : 195

Given the code fragment

```
int var1 = -5;

int var2 = var1--;
int var3 = 0;
if (var2 < 0) {
    var3 = var2++;
} else {
    var3 = --var2;
}
System.out.println(var3);
```

What is the result?

- A. - 6
- B. - 4
- C. - 5
- D. 5
- E. 4
- F. Compilation fails

Answer: C

Question No : 196

Given:

```
abstract class X {
    public abstract void methodX();
}
interface Y{
    public void methodY();
}
```

Which two code fragments are valid?

```
 A) class Z extends X implements Y{
    public void methodZ(){}
}

 B) abstract class Z extends X implements Y{
    public void methodZ(){}
}

 C) class Z extends X implements Y{
    public void methodX(){}
}

 D) abstract class Z extends X implements Y{
}

 E) class Z extends X implements Y{
    public void methodY(){}
}
```

- A. Option A
- B. Option B
- C. Option C
- D. Option D
- E. Option E

Answer: B,C

Explanation: When an abstract class is subclassed, the subclass usually provides implementations for all of the abstract methods in its parent class (C). However, if it does not, then the subclass must also be declared abstract (B).

Note: An abstract class is a class that is declared abstract—it may or may not include abstract methods. Abstract classes cannot be instantiated, but they can be subclassed.

Question No : 197

Given the following code for the classes MyException and Test:

```
public class MyException extends RuntimeException {}

public class Test {
    public static void main(String[] args) {
        try {
            method1();
        }
        catch (MyException ne) {
            System.out.print("A");
        }
    }
    public static void method1() { // line n1
        try {
            throw Math.random() > 0.5 ?new MyException() :new RuntimeException();
        }
        catch (RuntimeException re) {
            System.out.print("B");
        }
    }
}
```

What is the result?

- A. A
- B. B
- C. Either A or B
- D. A B
- E. A compile time error occurs at line n1

Answer: B

Question No : 198

Given the following code:

```
public static void main(String[] args){
    String[] planets = {"Mercury", "Venus", "Earth", "Mars"};
    System.out.println(planets.length);
    System.out.println(planets[1].length());
}
```

What is the output?

- A. 4
- 4
- B. 3
- 5
- C. 4
- 7

D. 5

4

E. 4

5

F. 4

21

Answer: E

Question No : 199

Which of the following data types will allow the following code snippet to compile?

```
float i = 4;  
float j = 2;  
_____ z = i + j;
```

- A. long
- B. double
- C. int
- D. float
- E. byte

Answer: B,D

Explanation:

Option B and D are the correct answer.

Since the variables I and j are floats, resultant will be float type too. So we have to use float or primitive type which can hold float, such a primitive type is double, it has wider range and also can hold floating point numbers, hence we can use double or float for the blank.

As explained above options B and D are correct.

long and int can't be used with floating point numbers so option A is incorrect.

Option E is incorrect as it have smaller range and also can't be used with floating point numbers.

[hnpsy/docs.oracle.com/javase/tutorial/java/javaOO/variables.html](http://docs.oracle.com/javase/tutorial/java/javaOO/variables.html)

Question No : 200

Given:

Class A { }

Class B { }

Interface X { }

Interface Y { }

Which two definitions of class C are valid?

- A. Class C extends A implements X { }
- B. Class C implements Y extends B { }
- C. Class C extends A, B { }
- D. Class C implements X, Y extends B { }
- E. Class C extends B implements X, Y { }

Answer: A,E

Explanation: extends is for extending a class.

implements is for implementing an interface.

Java allows for a class to implement many interfaces.

Question No : 201

Which two items can legally be contained within a java class declaration?

- A. An import statement
- B. A field declaration
- C. A package declaration
- D. A method declaration

Answer: B,D

Reference:

Question No : 202

Given the code fragment:

System.out.println(2 + 4 * 9 - 3); //Line 21

System.out.println((2 + 4) * 9 - 3); // Line 22

System.out.println(2 + (4 * 9) - 3); // Line 23

System.out.println(2 + 4 * (9 - 3)); // Line 24

System.out.println((2 + 4 * 9) - 3); // Line 25

Which line of codes prints the highest number?

- A. Line 21
- B. Line 22
- C. Line 23
- D. Line 24
- E. Line 25

Answer: B

Explanation: The following is printed:

35

51

35

26

35

Question No : 203

Given:

```
public class MainMethod {  
    void main() {  
        System.out.println("one");  
    }  
    static void main(String args) {  
        System.out.println("two");  
    }  
    public static void main(String[] args) {  
        System.out.println("three");  
    }  
    void mina(Object[] args) {  
        System.out.println("four");  
    }  
}
```

What is printed out when the program is executed?

- A. one
- B. two
- C. three
- D. four

Answer: C

| |
|--------------------------|
| Question No : 204 |
|--------------------------|

Given the code fragment:

```
public static void main(String[] args) {
    String[][] arr = {{"A", "B", "C"}, {"D", "E"}};
    for (int i = 0; i < arr.length; i++) {
        for (int j = 0; j < arr[i].length; j++) {
            System.out.print(arr[i][j] + " ");
            if (arr[i][j].equals("B")) {
                break;
            }
        }
        continue;
    }
}
```

What is the result?

- A. A B C
- B. A B C D E
- C. A B D E
- D. Compilation fails.

Answer: C

Question No : 205

Given the code fragment:

```
List colors = new ArrayList();
colors.add("green");
colors.add("red");
colors.add("blue");
colors.add("yellow");
colors.remove(2);
colors.add(3,"cyan");
System.out.print(colors);
```

What is the result?

-
- A. [green, red, yellow, cyan]
 - B. [green, blue, yellow, cyan]
 - C. [green, red, cyan, yellow]
 - D. An IndexOutOfBoundsException is thrown at runtime

Answer: A

Explanation: First the list [green, red, blue, yellow] is build.

The blue element is removed:

[green, red, yellow]

Finally the element cyan is added at then end of the list (index 3).

[green, red, yellow, cyan]

Question No : 206

Given the code fragment:

```
3. public static void main(String[] args) {  
4.     int x = 5;  
5.     while (isAvailable(x)) {  
6.         System.out.print(x);  
7.     }  
8. }  
10.  
11. public static boolean isAvailable(int x) {  
12.     return x-- > 0 ? true : false;  
13. }
```

Which modification enables the code to print 54321?

- A. Replace line 6 with System.out.print(--x);
- B. At line 7, insert x--;
- C. Replace line 6 with --x; and, at line 7, insert system.out.print(x);
- D. Replace line 12 With return (x > 0) ? false: true;

Answer: B

Question No : 207

Given:

```
public class SumTest {  
  
    public static void doSum(Integer x, Integer y) {  
        System.out.println("Integer sum is " + (x + y));  
    }  
  
    public static void doSum(double x, double y) {  
        System.out.println("double sum is " + (x + y));  
    }  
  
    public static void doSum(float x, float y) {  
        System.out.println("float sum is " + (x + y));  
    }  
  
    public static void doSum(int x, int y) {  
        System.out.println("int sum is " + (x + y));  
    }  
  
    public static void main(String[] args) {  
        doSum(10, 20);  
        doSum(10.0, 20.0);  
    }  
}
```

What is the result?

- A) int sum is 30
float sum is 30.0
- B) int sum is 30
double sum is 30
- C) Integer sum is 30
double sum is 30.0
- D) Integer sum is 30
float sum is 30.0

- A. Option A
B. Option B

-
- C. Option C
 - D. Option D

Answer: B

Question No : 208

Given the code fragment:

```
String[] strs = new String[2];
int idx = 0;
for (String s : strs) {
    strs[idx].concat(" element " + idx);
    idx++;
}
for (idx = 0; idx < strs.length; idx++) {
    System.out.println(strs[idx]);
}
```

What is the result?

- A. Element 0
Element 1
- B. Null element 0
Null element 1
- C. Null
Null
- D. A NullPointerException is thrown at runtime.

Answer: D

Question No : 209

```
public class StringReplace {
public static void main(String[] args) {
String message = "Hi everyone!";
```

```
System.out.println("message = " + message.replace("e", "X")); }  
}
```

What is the result?

- A. message = Hi everyone!
- B. message = Hi XvXryonX!
- C. A compile time error is produced.
- D. A runtime error is produced.
- E. message =
- F. message = Hi Xeveryone!

Answer: B

Question No : 210

Given:

```
public class App {  
    // Insert code here  
    System.out.print("Welcome to the world of Java");  
}  
}
```

Which two code fragments, when inserted independently at line // Insert code here, enable the program to execute and print the welcome message on the screen?

- A. static public void main (String [] args) {
- B. static void main (String [] args) {
- C. public static void Main (String [] args) {
- D. public static void main (String [] args) {
- E. public void main (String [] args) {

Answer: A,D

Explanation:

Incorrect:

Not B: No main class found.

Not C: Main method not found

not E: Main method is not static.

Question No : 211

Given:

```
interface Readable {  
    public void readBook();  
    public void setBookMark();  
}  
  
abstract class Book implements Readable { // line n1  
    public void readBook() { }  
    // line n2  
}  
  
class EBook extends Book { // line n3  
    public void readBook() { }  
    // line n4  
}
```

Which option enables the code to compile?

- A) Replace the code fragment at line n1 with:

```
class Book implements Readable {
```
- B) At line n2 insert:

```
public abstract void setBookMark();
```
- C) Replace the code fragment at line n3 with:

```
abstract class EBook extends Book {
```
- D) At line n4 insert:

```
public void setBookMark() { }
```

A. Option A

-
- B.** Option B
 - C.** Option C
 - D.** Option D

Answer: C,D

Question No : 212

Given the code fragment:

```
7.  StringBuilder sb1 = new StringBuilder("Duke");
8.  String str1 = sb1.toString();
9.  // insert code here
10. System.out.print(str1 == str2);
```

Which code fragment, when inserted at line 9, enables the code to print true?

- A.** String str2 = str1;
- B.** String str2 = new String (str1);
- C.** String str2 = sb1. toString ();
- D.** String str2 = "Duke";

Answer: A

Question No : 213

Given the code fragment:

```
String[] cartoons = {"tom", "jerry", "micky", "tom"};
int counter =0;

if ("tom".equals(cartoons[0])) {
    counter++;
} else if ("tom".equals(cartoons[1])) {
```

```
counter++;

} else if ("tom".equals(cartoons[2])) {

counter++;

} else if ("tom".equals(cartoons[3])) {

counter++;

}

System.out.print(counter);
```

What is the result?

- A. 1
- B. 2
- C. 4
- D. 0

Answer: A

Explanation: Counter++ will be executed only once because of the else if constructs.

Question No : 214

Given:

```
1. public class Whizlabs{
2.     private String name;
3.     private boolean pass;
4.
5.     public static void main(String[] args) {
6.         Whizlabs wb = new Whizlabs();
7.         System.out.print("name = " + wb.name);
8.         System.out.print(", pass = " + wb.pass);
9.     }
10. }
```

What would be the output, if it is executed as a program?

- A. name =, pass =
- B. name = null, pass = null
- C. name = null, pass = false
- D. name = null pass = true
- E. Compile error.

Answer: C

Explanation:

Both name and pass variables are instance variables, and we haven't given them any values, so they take their default values. For Boolean default value is false and for string which is not a primitive type default is null So at line 7, null will printed as the value of the variable name, and at line 8 false will be printed. Hence Option C is correct.

As explained above options A, B and D are incorrect.

Code compiles fine so option E is incorrect.

Reference:

<https://docs.oracle.com/javaseTutorial/java/javaOOVariables.html>

Question No : 215

Given:

```
public class X implements Z {
    public String toString() {
        return "X ";
    }
    public static void main(String[] args) {
        Y myY = new Y();
        X myX = myY;
        Z myZ = myX;
        System.out.print(myX);
        System.out.print((Y)myX);
        System.out.print(myZ);
    }
}

class Y extends X {
    public String toString() {
        return "Y ";
    }
}
```

-
- A. X XX
 - B. X Y X
 - C. Y Y X
 - D. Y YY

Answer: D

Question No : 216

Which of the following can fill in the blank in this code to make it compile?

```
public class Exam {  
    void method() {}  
}
```

```
public class OCAJP extends Exam{  
    ___ void method() {}  
}
```

- A. abstract
- B. final
- C. private
- D. default
- E. int

Answer: C

Explanation:

From Java SE 8, we can use static and/or default methods in interfaces, but they should be non abstract methods. SO in this case using default in blank is completely legal. Hence option C is correct.

Option A is incorrect as given method is not abstract, so can't use abstract there.

Options B and E are incorrect as we can't have non abstract method interface if they are not default or static.

<https://docs.oracle.com/javase/tutorial/java/lang1/defaultmethods.html>

Question No : 217

Given the code fragment:

```
int x = 100;
int a = x++;
int b = ++x;
int c = x++;
int d = (a < b) ? (a < c) ? a: (b < c )? b: c;
System.out.println(d);
```

What is the result?

- A. 100
- B. 101
- C. 102
- D. 103
- E. Compilation fails

Answer: E

Question No : 218

Given:

```
package p1;

public interface Dolinterface {

void method1(int n1); // line n1

}

package p3;

import p1.Dolinterface;
```

```
public class DoClass implements DoInterface {  
    public DoClass(int p1) {}  
    public void method1(int p1) {} // line n2  
    private void method2(int p1) {} // line n3  
}  
  
public class Test {  
    public static void main(String[] args) {  
        DoInterface doi= new DoClass(100); // line n4  
        doi.method1(100);  
        doi.method2(100);  
    }  
}
```

Which change will enable the code to compile?

- A. Adding the public modifier to the declaration of method1 at line n1
- B. Removing the public modifier from the definition of method1 at line n2
- C. Changing the private modifier on the declaration of method 2 public at line n3
- D. Changing the line n4 DoClass doi = new DoClass ();

Answer: C

Explanation: Private members (both fields and methods) are only accessible inside the class they are declared or inside inner classes. private keyword is one of four access modifier provided by Java and its a most restrictive among all four e.g. public, default(package), protected and private.

Read more: <http://javarevisited.blogspot.com/2012/03/private-in-java-why-should-you-always.html#ixzz3Sh3mOc4D>

Question No : 219

Given the code fragment:

```
class Student {  
    String name;  
    int age;  
}
```

And,

```
1. public class Test {  
2.     public static void main(String[] args) {  
3.         Student s1 = new Student();  
4.         Student s2 = new Student();  
5.         Student s3 = new Student();  
6.         s1 = s3;  
7.         s3 = s2;  
8.         s2 = null;  
9.     }  
10. }
```

Which statement is true?

- A. After line 8, three objects are eligible for garbage collection
- B. After line 8, two objects are eligible for garbage collection
- C. After line 8, one object is eligible for garbage collection
- D. After line 8, none of the objects are eligible for garbage collection

Answer: C

Question No : 220

Given:

```
class Cake {  
  
    int model;  
  
    String flavor;  
  
    Cake() {  
  
        model = 0;  
  
        flavor = "Unknown";  
  
    }  
  
}
```

```
public class Test {  
    public static void main(String[] args) {  
        Cake c = new Cake();  
        bake1(c);  
        System.out.println(c.model + " " + c.flavor);  
        bake2(c);  
        System.out.println(c.model + " " + c.flavor);  
    }  
    public static Cake bake1(Cake c) {  
        c.flavor = "Strawberry";  
        c.model = 1200;  
        return c;  
    }  
    public static void bake2(Cake c) {  
        c.flavor = "Chocolate";  
        c.model = 1230;  
        return;  
    }  
}
```

What is the result?

- A. 0 unknown
0 unknown
- B. 1200 Strawberry
1200 Strawberry
- C. 1200 Strawberry
1230 Chocolate
- D. Compilation fails

Answer: C

Explanation: 1200 Strawberry

1230 Chocolate

Question No : 221

Which of the following exception will be thrown due to the statement given here?

int array[] = new int[-2];

- A. NullPointerException
- B. NegativeArraySizeException
- C. ArrayIndexOutOfBoundsException
- D. IndexOutOfBoundsException
- E. This statement does not cause any exception.

Answer: B

Explanation:

In given statement we can see that, we have passed negative value for creating int array, which results a NegativeArraySize Exception. Hence option B is correct.

Option A is incorrect as it is thrown when an application attempts to use null in a case where an object is required.

Option D is incorrect as IndexOutOfBoundsException thrown to indicate that an index of some sort (such as to an array, to a string, or to a vector) is out of range.

REFERENCE

<http://docs.oracle.com/javase/8/docs/api/java/lang/NegativeArraySizeException.html>

Question No : 222

Given:

```
class Vehicle {
    String type = "4W";
    int maxSpeed = 100;

    Vehicle(String type, int maxSpeed) {
        this.type = type;
        this.maxSpeed = maxSpeed;
    }
}

class Car extends Vehicle {
    String trans;

    Car(String trans) {           //line n1
        this.trans = trans;
    }

    Car(String type, int maxSpeed, String trans) {
        super(type, maxSpeed);
        this(trans);           //line n2
    }
}
```

And given the code fragment:

```
7. Car c1 = new Car("Auto");
8. Car c2 = new Car("4W", 150, "Manual");
9. System.out.println(c1.type + " " + c1.maxSpeed + " " + c1.trans);
10. System.out.println(c2.type + " " + c2.maxSpeed + " " + c2.trans);
```

What is the result?

- A.** 4W 100 Auto
4W 150 Manual
- B.** Null 0 Auto
4W 150 Manual
- C.** Compilation fails only at line n1
- D.** Compilation fails only at line n2
- E.** Compilation fails at both line n1 and line n2

Answer: E

Explanation:

On line n1 implicit call to parameterized constructor is missing and n2 this() must be the first line.

Question No : 223

Given the code fragment:

```
String[] colors = {"red", "blue", "green", "yellow", "maroon", "cyan"};
```

Which code fragment prints blue, cyan, ?

```
C A) for (String c:colors){  
    if (c.length() != 4) {  
        continue;  
    }  
    System.out.print(c+", ");  
}  
  
C B) for (String c:colors[]) {  
    if (c.length() <= 4) {  
        continue;  
    }  
    System.out.print(c+", ");  
}  
  
C C) for (String c:String[] colors) {  
    if (c.length() >= 3) {  
        continue;  
    }  
    System.out.print(c+", ");  
}  
  
C D) for (String c:colors){  
    if (c.length() != 4) {  
        System.out.print(c+", ");  
        continue;  
    }  
}
```

- A. Option A
 - B. Option B
 - C. Option C
-

D. Option D

Answer: A

Question No : 224

Given the code fragment:

```
int num[][] = new int[1][3];
for (int i = 0; i < num.length; i++) {
    for (int j = 0; j < num[i].length; j++) {
        num[i][j] = 10;
    }
}
```

Which option represents the state of the num array after successful completion of the outer loop?

- A) num[0][0]=10
 num[0][1]=10
 num[0][2]=10
 - B) num[0][0]=10
 num[1][0]=10
 num[2][0]=10
 - C) num[0][0]=10
 num[0][1]=0
 num[0][2]=0
 - D) num[0][0]=10
 num[0][1]=10
 num[0][2]=10
 num[0][3]=10
 num[1][0]=0
 num[1][1]=0
 num[1][2]=0
 num[1][3]=0
-

-
- A.** Option A
 - B.** Option B
 - C.** Option C
 - D.** Option D

Answer: A

Question No : 225

Given:

```
class Alpha {  
    int ns;  
    static int s;  
    Alpha(int ns) {  
        if (s < ns) {  
            s = ns;  
            this.ns = ns;  
        }  
    }  
    void doPrint() {  
        System.out.println("ns = " + ns + " s = " + s);  
    }  
}
```

And,

```
public class TestA {  
    public static void main(String[] args) {  
        Alpha ref1 = new Alpha(50);  
        Alpha ref2 = new Alpha(125);  
        Alpha ref3 = new Alpha(100);  
        ref1.doPrint();  
        ref2.doPrint();  
        ref3.doPrint();  
    }  
}
```

- A.** ns = 50 S = 125
ns = 125 S = 125
ns = 100 S = 125
 - B.** ns = 50 S = 125
ns = 125 S = 125
ns = 0 S = 125
 - C.** ns = 50 S = 50
ns = 125 S = 125
ns = 100 S = 100
 - D.** ns = 50 S = 50
ns = 125 S = 125
ns = 0 S = 125
-

Answer: B

Question No : 226

Given:

```
public class Palindrome {  
    public static int main(String[] args) {  
        System.out.print(args[1]);  
        return 0;  
    }  
}
```

And the commands:

```
javac Palindrome.java  
java Palindrome Wow Mom
```

What is the result?

- A. Compilation fails
- B. The code compiles, but does not execute.
- C. Paildrome
- D. Wow
- E. Mom

Answer: B

Question No : 227

Given:

```
public class Test2 {  
    public static void main(String[] args) {  
        int ar1[] = {2, 4, 6, 8};  
        int ar2[] = {1, 3, 5, 7, 9};  
        ar2 = ar1;  
        for (int e2 : ar2) {  
            System.out.print(" " + e2);  
        }  
    }  
}
```

What is the result?

- A. 2 4 6 8
- B. 2 4 6 8 9
- C. 1 3 5 7
- D. 1 3 5 7 9

Answer: D

Question No : 228

Given:

```
class Sports {  
    int num_players;  
    String name, ground_condition;  
    Sports(int np, String sname, String sground){  
        num_players = np;  
        name = sname;  
        ground_condition = sground;  
    }  
}  
  
class Cricket extends Sports {  
    int num_umpires;  
    int num_substitutes;
```

Which code fragment can be inserted at line //insert code here to enable the code to compile?

- A. Cricket() {
super(11, "Cricket", "Condidtion OK");
num_umpires =3;

```
num_substitutes=2;
}
B. Cricket() {
super.ground_condition = "Condition OK";
super.name="Cricket";
super.num_players = 11;
num_umpires =3;
num_substitutes=2;
}
C. Cricket() {
this(3,2);
super(11, "Cricket", "Condidtion OK");
}
Cricket(int nu, ns) {
this.num_umpires =nu;
this.num_substitutes=ns;
}
D. Cricket() {
this.num_umpires =3;
this.num_substitutes=2;
super(11, "Cricket", "Condidtion OK");
}
```

Answer: A

Explanation:

Incorrect:

not C, not D: call to super must be the first statement in constructor.

Question No : 229

Which usage represents a valid way of compiling java source file with the name "Main"?

- A.** javac Main.java
- B.** java Main.class
- C.** java Main.java
- D.** javac Main
- E.** java Main

Answer: A

Explanation: The compiler is invoked by the javac command. When compiling a Java class, you must include the file name, which houses the main classes including the Java

extension. So to run Main.java file we have to use command in option A.
TO execute Java program we can use Java command but can't use it for compiling.
<https://docs.oracle.com/javase/tutorial/getStarted/application/index.html>

Question No : 230

Given the code fragment:

```
public static void main(String[] args) {  
    String[] arr = {"A", "B", "C", "D"};  
    for (int i = 0; i < arr.length; i++) {  
        System.out.print(arr[i] + " ");  
        if (arr[i].equals("C")) {  
            continue;  
        }  
        System.out.println("Work done");  
        break;  
    }  
}
```

What is the result?

- A. A B C Work done
- B. A B C D Work done
- C. A Work done
- D. Compilation fails

Answer: C

Question No : 231

Which two are valid array declaration?

- A. Object array[];
- B. Boolean array[3];

-
- C. int[] array;
 - D. Float[2] array;

Answer: A,C

Question No : 232

Given the code fragment:

```
public class Test {  
    public static void main(String[] args) {  
        //line n1  
        switch (x) {  
            case 1:  
                System.out.println("One");  
                break;  
            case 2:  
                System.out.println("Two");  
                break;  
        }  
    }  
}
```

Which three code fragments can be independently inserted at line n1 to enable the code to print one?

- A. Byte x = 1;
- B. short x = 1;
- C. String x = "1";
- D. Long x = 1;
- E. Double x = 1;
- F. Integer x = new Integer ("1");

Answer: A,B,F

Question No : 233

Given:

```
1. public class Whizlabs{  
2.     public static void main(String[] args){  
3.         StringBuilder sb = new StringBuilder("1Z0");  
4.         sb.concat("-808");  
5.         System.out.println(sb);  
6.     }  
7. }
```

What is the output?

- A. 1Z0
- B. 1Z0-808
- C. An exception will be thrown.
- D. Compilation fails due to error at line 3.
- E. Compilation fails due to error at line 4.

Answer: E

Explanation:

Option E is the correct answer.

Code fails to compile because there is no method called concert in StringBuilder class. The concert method is in String class. Hence option E is correct. Here we should have used append method of StringBuilder class, in that case option B would be correct.

<https://docs.oracle.com/javase/tutorial/java/data/buffers.html>

Question No : 234

Given the code fragment:

```
public static void main(String[] args) {  
    StringBuilder sb = new StringBuilder(5);  
    String s = "";  
  
    if (sb.equals(s)) {  
        System.out.println("Match 1");  
    } else if (sb.toString().equals(s.toString())) {  
        System.out.println("Match 2");  
    } else {  
        System.out.println("No Match");  
    }  
}
```

What is the result?

- A. Match 1
- B. Match 2
- C. No Match
- D. A NullPointerException is thrown at runtime.

Answer: B

Explanation:

it will compare the string contents of the StringBuilder with string object.

Question No : 235

Given the following class:

```
public class CheckingAccount {  
    public int amount;  
    public CheckingAccount(int amount) {  
        this.amount = amount;  
    }  
    public int getAmount () {  
        return amount;  
    }  
    public void changeAmount(int x) {  
        amount += x;  
    }  
}
```

And given the following main method, located in another class:

```
public static void main(String[] args) {  
    CheckingAccount acct = new CheckingAccount((int)(Math.random()*1000));  
    //line n1  
    System.out.println(acct.getAmount());  
}
```

Which three lines, when inserted independently at line n1, cause the program to print a 0 balance?

- A. this.amount = 0;
- B. amount = 0;
- C. acct (0) ;
- D. acct.amount = 0;
- E. acct. getAmount () = 0;
- F. acct.changeAmount(0);
- G. acct.changeAmount(-acct.amount);
- H. acct.changeAmount(-acct.getAmount());

Answer: D,G,H

Question No : 236

Given:

```
class Overloading {  
    int x(double d) {  
        System.out.println("one");  
        return 0;  
    }  
    String x(double d) {  
        System.out.println("two");  
        return null;  
    }  
    double x(double d) {  
        System.out.println("three");  
        return 0.0;  
    }  
    public static void main(String[] args) {  
        new Overloading().x(4.0);  
    }  
}
```

What is the result?

- A. One
- B. Two
- C. Three
- D. Compilation fails.

Answer: D
