**Question:01 What is the exact output of this code?**

class A {

}

public class B {

void m1() {

System.out.println ("This is method of Class B");

}

}

public class C {

public static void main (String [] args) {

B objB = new B ();

System.out.print("This is Class C");

objB.m1();

}

}

**option:**

A. This is method of Class B

B. This is Class C.

C. This is Class C, This is method of Class B.

D. Compilation Error.

**Answer:** option D-Compilation Error.

**Explanation:** we cannot declare more than one public class in the single java program.

In this program, we have two public classes B and C that's why it shows compilation error.

**Question:02 What is the output of this code?**

Note: Save this code as GlobalClass.java, Compile it and execute it.

class A {

public static void main (String [] args) {

System.out.print("This is Class A");

}

}

class B {

public static void main (String [] args) {

System.out.print("This is Class B");

}

}

class C {

public static void main (String [] args) {

System.out.print("This is Class C");

}

}

class D {

}

**options:**

A. In a Class, Cannot be define more than one Main method.

B. Code successfully compile and Execute.

C. NoClassDefFoundError.

D. None of the above

**Answer:** options C-NoClassDefFoundError.

**Explanation:** Its compile successfully but at the time of execution it can't able to

load main class GlobalClass because we didn't declare any class as GlobalClass in this program.

**Question:03 What is the output of this code?**

public class DemoTestArrays {

public static void main (String [] args) {

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

int arrTwo [] = { 0, 0, 0, 0, 0 };

for (int i = 0; i < arrOne.length; i++) {

arrTwo[i] = arrOne[arrOne.length - i - 1];

}

System.out.println(Arrays.toString(arrTwo));

}

}

**option:**

A. [0, 0, 0, 0, 0].

B. [5, 4, 3, 2, 1].

C. [1, 2, 3, 4, 5].

D. Runtime Error.

**Answer:** option D-Runtime Error.

**Explanation:** The Arrays.toString() does not works because we didn't use import java.util.Arrays package.

**Question:04 What is the output of this code?**

public class DemoTestClass {

public static void main (String [] args) {

String [] elements = { "AAA", "BBB", "CCC" };

String first = (elements.length > 0) ? elements[0] : null;

System.out.println(first);

}

}

**Option:**

A. BBB.

B. CCC.

C. AAA.

D. Runtime Error.

**Answer:** option C-AAA.

**Question:05 Is there a destructor for Java?**

**option:**

A. No, Because Java is a garbage collected language, you cannot predict when (or even if)

an object will be destroyed.

B. Yes, Java is quite mature as a language and memory leak can be fixed.

C. Java objects are heap allocated and garbage collected, that's why destructor used in

java.

D. None of the above

**Answer:** option A-No, Because Java is a garbage collected language, you cannot predict when (or even if)

an object will be destroyed.

**Explanation:** Java contains garbage collector that works the same as destructor.

**Question:06 Read carefully below code and identify the correct answer?**

public class ClassMain {

public static void main (String [] args) {

String main = "main is incorrect defined";

System.out.println(main);

}

}

**option:**

A. Yes, it compiles and execute because, the character sequence "main" is an identifier.

B. No, because main is a keyword/reserve word in java.

C. It does not compile.

D. In Java, Main keyword is not used twice.

**Answer:** option A-Yes, it compiles and execute because, the character sequence "main" is an identifier.

**Explanation:** main is not a reserved identifier and it is allowed to name variables as main.

**Question:07 Read the given below code and identify correct Output?**

class MyProgram {

int count = 0;

public static void main (String [] args) {

System.out.println(count);

}

}

**option:**

A. null.

B. 0.

C. Error.

D. None of the above.

**Answer:** option C-Error.

**Explanation:** count is nonstatic variables because it is declared before the static main method hence it is not referenced by static main method.

**Question:08 How many Objects created in the below code?**

class X {

X () {

System.out.println(this.hashCode());

}

}

class Y extends X {

Y () {

System.out.println(this.hashCode());

}

}

public class TestClass {

public static void main (String [] args) {

Y y = new Y ();

System.out.println(y.hashCode());

}

}

**option:**

A. 3.

B. 2.

C. 1.

D. None of the above

**Answer:** option C-1.

**Explanation:** In main method,we created only one object for class Y.

**Question :09 What is the correct output of the given code?**

public class Test {

public static double calculation (double a, double b) {

if (a == b) {

return 0;

} else {

return 2 / (a - b);

}

}

public static void main (String [] args) {

double d1 = Double.MIN\_VALUE;

double d2 = 2.0 \* Double.MIN\_VALUE;

System.out.println("Result: " + calculation (d1, d2));

}

}

**option:**

A. 0.0

B. 0

C. Error

D. -Infinity

**Answer:** option:D.-Infinity

**Explanation:** In d1 variable its store Double.MIN\_VALUE(2^-1074).In d2 it multiplies

with 2 and then passess to calculation in that it checks a==b it is not equal,then its goes

to else part and returns 2 / (a - b) hence the answer is -infinity.

**Question :10 What is the correct answer of the below code?**

public class Test {

public static void main (String [] args) {

int j = 0;

if ((8 > 4) | (j++ == 7))

System.out.println("j = " + j);

}

}

option:

A. 0

B: 1

C. 2

D. ArithmeticException (Divided by zero)

**Answer:** option B: 1

**Explanation:** If condition checks bitwise OR operator((8>4)true and (1==7)false) which is true|false =true then returns j value as 1.

**Question :11 What is the output of below code?**

public class Test {

public static void main(String[] args) {

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

int sum = 0;

for (int i : array)

sum += ++i;

System.out.println(--sum);

}

}

**option:**

A. 15

B. 16

C. 20

D. 19

**Answer:** option D: 19

**Explanation:** In For each loop,i access all the values of array elements and perform sum

atlast it prints --sum.

**Question : 12 Find Out the correct output of the given code?**

public class MathTest {

public void main(String[] args) {

int x = 10 \* 10 - 10;

System.out.println(++x);

}

}

**option:**

A. 0

B. 90

C. 91

D. Runtime Error

**Answer:** option D: Runtime Error

**Explanation:** main method is not static in class mathtest.

**Question : 13 Can we create a user defined immutable class, pick the correct option?**

**option:**

A.Make the class as final and

B.Make the data members as private and final.

C.Both A and B are Correct

D.None of the above

**Answer:** option C:Both A and B are Correct

**Question : 14 How to define Vector class?**

**option:**

A. Synchronized and Non-serialized

B. Non-Synchronized and Serialized.

C.Both A and B are Correct

D.None of the above

**Answer:** option D:None of the above

**Explanation:** Vector is synchronized, which means only one thread at a time can access the code and vector is serialized

**Question : 15 What is the output of the below code?**

public class TestString1 {

public static void main(String[] args) {

String str = "420";

str += 42;

System.out.print(str);

}

}

**option:**

A. 420

B. 42042.

C. Compilation fails

D. An exception is thrown at runtime

**Answer:** option B: 42042.

**Explanation:** 420+42=42042

420 is string value so it accepts integer value as string and perform concatenation.

**Question : 16 What is the output of the below code?**

class Test {

public static void main(String[] args) {

int x = 0;

int y = 10;

do {

y--;

++x;

} while (x < 5);

System.out.print(x + "," + y);

}

}

**option:**

A. 5, 6

B. 5, 5.

C. 6, 5

D. Error

**Answer:** option B: 5, 5.

**Explanation:** At first iteration,y--=9,++x=1 its continued until (x<5),atlast

y--=5,++x=5.

**Question : 17 What definition exactly match for abstract class? ?**

**option:**

A. public abstract class A {

public Bark speak();

}

B. public abstract class A {

public Bark speak() {

}

}

C. public class A {

public abstract Bark speak();

}

D. public class A abstractt{

public abstract Bark speak();

}

**Answer:** None of the above

**Explanation:** syntax:public abstract class A{}

public abstract void Bark speak();

**Question : 18 Read the below code and pick correct option?**

class LoopTestDemo {

public static void main(String[] args) {

int x = 12;

while (x < 10) {

x--;

}

System.out.print(x);

}

}

**option:**

A. 11

B. 10

C. 12

D. 9

**Answer:** option c:12.

**Explanation:** (x<10)false because x is greater than 10.so condition is false

and print x value as 12.

**Question : 19 Read the below code and pick correct option?**

class BitwiseTestDemo {

public static void main(String[] args) {

int x = 5;

int y = 7;

System.out.print(((y \* 2) % x));

System.out.print(" " + (y % x));

}

}

**option:**

A. 6, 8

B. 7, 9

C. 4, 6

D. 4, 2

**Answer:** option D: 4, 2

**Explanation:** (7\*2)/5 = 4 and 7%5 = 2.

**Question : 20 Read the below code and pick correct option?**

class TestFormatSpecifier {

static final long num = 343L;

static long testMethod(long num) {

System.out.print(++num + " ");

return ++num;

}

public static void main(String[] args) {

System.out.print(num + " ");

final long num = 340L;

new TestString1().testMethod(num);

System.out.println(num);

}

}

**option:**

A.343 340 342

B.343 341 342

C.343 341 340

D.An exception is thrown at runtime

**Answer:** option D:An exception is thrown at runtime

**Explanation:** TestString1() is not declared.

**Question : 21 Read the below code and pick correct option?**

public class TestBooleanDemo {

public static void main(String[] args) {

int x = 5;

boolean b1 = true;

boolean b2 = false;

if ((x == 4) && !b2)

System.out.print("1 ");

else

System.out.print("2 ");

if ((b2 = true) && b1)

System.out.print("3 ");

}

}

**option:**

A. 2, 3

B. 1, 2

C. 3, 2

D. An exception is thrown at runtime

**Answer:** option A: 2, 3

**Explanation:** First if condition is false so it execute the else part and print 2

then second if condition is true and print 3

**Question : 22 Read the below code and pick correct option?**

public class Test {

public void main(String[] args) {

int x = 6;

Test test = new Test();

test.doSomething(x);

System.out.print(" main x = " + x);

}

void doSomething(int x) {

System.out.print(" method x = " + x++);

}

}

**option:**

A. An exception is thrown at runtime

B. method x = 6, main x = 6

C. method x = 6 main x = 7

D. method x = 7 main x = 6

**Answer:** option A: An exception is thrown at runtime

**Explanation:** main method is not static

**Question : 23 Read the below code and pick correct option?**

class TernanryTestDemo {

public static void main(String[] args) {

int i = 42;

String str = (i < 40) ? "Computer" : (i > 50) ? "Java" : "Everything";

System.out.println(str);

}

}

**option:**

A. An exception is thrown at runtime

B. Computer

C. Java

D. Everything

**Answer:** option D: Everything

**Explanation:** Its check Condition (42<40) is false. Then switch to the next condition (42>50), which is also false. Therefore it prints the else condition as Everything.

.

**Question : 24 Read the below code and pick correct option?**

class ExceptionTestDemo {

public static void main(String[] args) {

Float valuePie = new Float(3.14f);

try {

if (valuePie > 3)

System.out.print("Pie value is greater than 3"+", ");

else

System.out.print("Pie value is not greater than 3"+", ");

}

catch (Exception e) {

e.printStackTrace();

} finally {

System.out.println ("Have a nice day.");

}

}

}

**option:**

A.Pie value is not greater than 3, Have a nice day.

B.Pie value is greater than 3, Have a nice day.

C.Pie value is not greater than 3.

D.An exception is thrown at runtime.

**Answer:** option B:Pie value is greater than 3, Have a nice day.

**Explanation:** it checks condition(3.14>3)is true then its prints Pie value is greater than 3.

And catch is not executed because if an exception occurs in try block catch is executed at that time.Final condition is always executed. Therefore it prints Have a nice day.

**Question : 25 Read the below code and pick correct option?**

class TernaryDemo {

public static void main(String[] args) {

int a = 8;

System.out.println ("" + (int) ((a < 8) ? 9.9 : 9));

}

}

**option:**

A. 9.9

B. 0.

C. 9.

D. Error.

**Answer:** option C: 9.

**Explanation:** it checks the condition ,it is false so else part is executed and 9 is printed.

**Question : 26 Read the below code and pick correct option?**

class TestDoubleDemo {

public static long round(double a) {

if (a != 0x1.fffffffffffffp-2) {

return (long)Math.floor(a + 0.5d);

} else {

return 0;

}

}

public static void main(String[] args) {

TestDoubleDemo t = new TestDoubleDemo();

t.round(2.5);

}

}

**option:**

A. 3

B. 0.

C. -1.

D. None of the above.

**Answer:** option D: None of the above.

**Explanation:** nothing is displayed, we didn't include print statements.

**Question : 27 Create a parent class as below**

class A {

private int a = 0;

}

Which one is tightly encapsulated in the below options

**option:**

A. class B extends A {

int a = 0;

}

B. class C extends A {

private int a = 0;

}

C. class B extends A {

static int a = 0;

}

D. class C extends A {

final int a = 0;

}

**Answer:** option B: class C extends A {

private int a = 0;

}

**Explanation:** If all the data members in class are declared as private and if it’s inherited by another class

which too has all private data members. Then it is called tightly encapsulated.

**Question : 28 Cyclic inheritance allowed in Java or Not??**

class A extends B {

// some methods

}

class B extends A {

// some methods

}

**option:**

A. No, Not Allowed.

B. Yes, Definitely Allowed.

C. With Some condition, Allowed

D. None of the Above

**Answer:** option A:No, Not Allowed.

**Explanation:** A is parent class B is child class.parent class only allowed to extend its class

**Question : 29 Read the below code and find correct output?**

public class Main {

public static void main(String[] args) {

Integer x = 400, y = 400;

if (x == y)

System.out.println("Number is Same");

else

System.out.println("Number is Not Same");

}

}

**option:**

A. Number is Same

B. Number is Not Same

C. Runtime Exception

D. None of the Above.

**Answer:** option B: Number is Not Same

**Explanation:** x==y, it compares the address of the variables that's why its not same.