Q) //Consider the following code and choose the correct option:

class X

{

int x;

X(int x)

{

x=2;

}

}

class Y extends X

{

Y(){}

void displayX()

{

System.out.print(x);

}

public static void main(String args[])

{

new Y().displayX();

}

}

a)Compiles and display 2

b) Compiles and runs without any output

c) Compiles and display 0

d) Compilation Error

ANS: d

Q) //Consider the following code and choose the correct option:

class Test

{

private void display()

{

System.out.println("Display()");

}

private static void show()

{

display();

System.out.println("show()");

}

public static void main(String arg[])

{

show();

}

}

a)Compiles and prints show()

2)Compiles and prints Display() show()

c)Compiles but throws runtime exception

d)Compilation error

ANS: d

Q) Consider the following code and choose the correct option:

class A

{

A()

{

System.out.print("From A");

}

}

class B extends A

{

B(int z)

{

z=2;

}

public static void main(String args[])

{

new B(3);

}

}

a)Compilation error

b)Comiples and prints From A

c)Compiles but throws runtime exception

d)Compiles and display 3

ANS::b

Q) class One

{

int var1;

One (int x)

{

var1 = x;

}

}

class Derived extends One

{

int var2;

void display()

{

System.out.println("var 1="+var1+"var2="+var2);

}

}

class Main

{

public static void main(String[] args)

{

Derived obj = new Derived();

obj.display();

}

}

consider the code above & select the proper output from the options.

a)0 , 0

b)compiles successfully but runtime error

c)compile error

d)none of these

ANS:: C

Q) package aj;

public class A {

protected int j;

}

package bj;

//import aj.A;

class B extends A

{

public static void main(String ar[])

{

System.out.print(new A().j=23);

}

}

a)code compiles fine and will display 23

b)code compiles but will not display output

c)j can not be initialized

d)compliation error

ANS:D

Q) class Order

{

Order()

{

System.out.println("Cat");

}

public static void main(String... Args)

{

Order obj = new Order();

System.out.println("Ant");

}

static

{

System.out.println("Dog");

}

{

System.out.println("Man");

}

}

**consider** the code above & select the proper output from the options.

a)compile error

b)Man Dog Cat Ant

c)Dog Man Cat Ant

d)Cat Ant Dog Man

ANS:C

Q) public class MyAr

{

public static void main(String argv[])

{

MyAr m = new MyAr();

m.amethod();

}

public void amethod()

{

final int i1;

System.out.println(i1);

}

}

What is the Output of the Program?

a)Unresolved compilation problem: The local variable i1 may not have been initialized

b)Compilation and output of null

c)None of the given optionsclass

ANS:A

Q)

class MyClass1

{

private int area(int side)

{

return(side \* side);

}

public static void main(String args[ ])

{

MyClass1 MC = new MyClass1( );

int area = MC.area(50);

System.out.println(area);

}

}

/\*What would be the output?

a)Compilation error

b)Runtime Exception

c)2500

d)50

ANS:2500

Q) class Sample

{

int a,b;

Sample()

{

a=1; b=2;

System.out.println(a+"\t"+b);

}

Sample(int x)

{

this(10,20);

a=b=x;

System.out.println(a+"\t"+b);

}

Sample(int a,int b)

{

this();

this.a=a;

this.b=b;

System.out.println(a+"\t"+b);

}

}

class This2

{

public static void main(String args[])

{

Sample s1=new Sample (100);

}

}

What is the Output of the Program?

a)100 100 1 2 10 20

b)1 2 100 100 10 20

c)10 20 1 2 100 100

d)1 2 10 20 100 100

ANS:D

Q) Consider the following code and choose the correct option:

public class MyClass

{

public static void main(String arguments[])

{

amethod(arguments);

}

public void amethod(String[] arguments)

{

System.out.println(arguments[0]);

System.out.println(arguments[1]);

}

}

/\*Command Line arguments -Hi, Hello

a)prints Hi Hello

b)Compiler Error

c)Runs but no output

d)Runtime Error

ANS:B

Q) Given:

public class Yikes

{

public static void go(Long n)

{

System.out.print("Long ");

}

public static void go(Short n)

{

System.out.print("Short ");

}

public static void go(int n)

{

System.out.print("int ");

}

public static void main(String [] args)

{

short y = 6;

long z = 7;

go(y);

go(z);

}

}

What is the result?

a)int Long

b)Short Long

c)Compilation fails.

d)An exception is thrown at runtime.

ANS:A

Q) abstract class MineBase

{

abstract void amethod();

static int i;

}

public class Mine extends MineBase

{

public static void main(String argv[])

{

int[] ar=new int[5];

for(i=0;i < ar.length;i++)

System.out.println(ar[i]);

}

}

a)A Sequence of 5 zero's will be printed like 0 0 0 0 0

b)A Sequence of 5 one's will be printed like 1 1 1 1 1

c)IndexOutOfBoundes Error

d)Compilation Error occurs and to avoid them we need to declare Mine class as abstract

ANS:D

Q) What will be the result when you attempt to compile this program?

public class Rand

{

public static void main(String argv[])

{

int iRand;

iRand = Math.random();

System.out.println(iRand);

}

}

/\*

a)Compile time error referring to a cast problem

b)A random number between 1 and 10

c)A random number between 0 and 1

d)A compile time error as random being an undefined method

ANS:A

Q) Which of the following declarations are correct? (Choose TWO)

public class Declaration {

boolean b = TRUE; //line 1

byte b = 256; //line 2

String s = "null"; ////line 3

int i = new Integer("56"); ////line 4

}

a)line 1

b)line 2

c)line 3

d)line 4

ANS: C & D

Q) class A, B and C are in multilevel inheritance hierarchy repectively .

In the main method of some other class if class C object is created, in what sequence the three constructors execute?

a)Constructor of A executes first, followed by the constructor of B and C

b)Constructor of C executes first followed by the constructor of A and B

c)Constructor of C executes first followed by the constructor of Band A

d)Constructor of A executes first followed by the constructor of C and B

ANS:A

Q) What will be the result when you try to compile and run the following code?

private class Base

{

Base()

{

int i = 100;

System.out.println(i);

}

}

public class Pri extends Base

{

static int i = 200;

public static void main(String argv[])

{

Pri p = new Pri();

System.out.println(i);

}

}

/\*

a)200

b)100 followed by 200

c)Compile time error

d)100

ANS:C

Q) Suppose class B is sub class of class A:

A) If class A doesn't have any constructor, then class B also must not have any constructor

B) If class A has parameterized constructor, then class B can have default as well as parameterized constructor

C) If class A has parameterized constructor then call to class A constructor should be made explicitly by constructor of class B

🡪

1. Only B and C is TRUE
2. Only A is TRUE
3. All are FALSE
4. Only A and C is TRUE

ANS:C

Q) What will be printed out if you attempt to compile and run the following code ?

public class AA

{

public static void main(String[] args)

{

int i = 9;

switch (i)

{

default:

System.out.println("default");

case 0:

System.out.println("zero");

break;

case 1:

System.out.println("one");

case 2:

System.out.println("two");

}

}

}

a)default zero

b)one two

c)default zero

c)Compilation Error

d)default

ANS:C

Q)Consider the following code and choose the correct option:

package aj;

private class S

{

int roll;

S(){roll=1;

}

}

package aj;

class T

{

public static void main(String ar[])

{

System.out.print(new S().roll);

}

}

/\*

a)Compiles but no output

b)Compiles and diplay 0

c)Compilation error

d)Compiles and display 1

ANS: C

Q) public class Q

{

public static void main(String argv[])

{

int anar[] = new int[] { 1, 2, 3 };

System.out.println(anar[1]);

}

}

a)Compiler Error: anar is referenced before it is initialized

b)2

c)1

d)Compiler Error: size of array must be defined

ANS:B

Q) Which statements, when inserted at (1), will not result in compile-time errors?

public class ThisUsage

{

int planets;

static int suns;

public void gaze()

{

int i;

// (1) INSERT STATEMENT HERE

}

}

a)i = this.planets;

b)i = this.suns;

c)this = new ThisUsage();

d)this.i = 4;

ANS: A Or B

Q) Given the following code what will be output?

public class Pass

{

static int j=20;

public static void main(String argv[])

{

int i=10;

Pass p = new Pass();

p.amethod(i);

System.out.println(i);

System.out.println(j);

}

public void amethod(int x)

{

x=x\*2;

j=j\*2;

}

}

/\*

a)Error: amethod parameter does not match variable

b)10 and 40

c)10, and 20

d)20 and 40

ANS:B

Q) class Order

{

Order()

{

System.out.println("Cat");

}

public static void main(String... Args)

{

System.out.println("Ant");

}

static

{

System.out.println("Dog");

}

{

System.out.println("Man");

}

}

/\*consider the code above & select the proper output from the options.

a)Dog Ant

b)Dog Man Cat Ant

c)Man Dog Ant

d)Dog Man Ant

ANS:A

Q) public class C123 {

private C123()

{

System.out.println("Hellow");

}

public static void main(String args[])

{

C123 o1 = new C123();

C213 o2 = new C213();

}

}

class C213

{

private C213()

{

System.out.println("Hello123");

}

}

What is the output?

a)Hellow

b)It is not possible to declare a constructor as private

c)Compilation Error

d)Runs without any output

ANS:C

Q) class A

{

int i, j;

A(int a, int b)

{

i = a;

j = b;

}

void show()

{

System.out.println("i and j: " + i + " " + j);

}

}

class BB extends A

{

int k;

BB(int a, int b, int c)

{

super(a, b);

k = c;

}

void show(String msg)

{

System.out.println(msg + k);

}

}

public class Override

{

public static void main(String args[])

{

BB subOb = new BB(3, 5, 7);

subOb.show("This is k: "); // this calls show() in B

subOb.show();// this calls show() in A

}

}

/\*What would be the ouput?

a)This is j: 5 i and k: 3 7

b)This is i: 3 j and k: 5 7

c)This is i: 7 j and k: 3 5

d)This is k: 7 i and j: 3 7

ANS:D

Q) public class MyAr

{

static int i1;

public static void main(String argv[])

{

MyAr m = new MyAr();

m.amethod();

}

public void amethod()

{

System.out.println(i1);

}

}

What is the output of the program?

a)Compilation Error

b)Garbage Value

c)It is not possible to access a static variable in side of non static method

d)0

ANS:D

Q) Given:

package bj;

class Meal

{

Meal()

{

System.out.println("Meal()");

}

}

class Cheese

{

Cheese()

{

System.out.println("Cheese()");

}

}

class Lunch extends Meal

{

Lunch()

{

System.out.println("Lunch()");

}

}

class PortableLunch extends Lunch

{

PortableLunch()

{

System.out.println("PortableLunch()");

}

}

class Sandwich extends PortableLunch

{

private Cheese c = new Cheese();

public Sandwich()

{

System.out.println("Sandwich()");

}

}

public class MyClass7

{

public static void main(String[] args)

{

new Sandwich();

}

}

/\*What would be the output?

a)Meal() Lunch() PortableLunch() Cheese() Sandwich()

b)Meal() Cheese() Lunch() PortableLunch() Sandwich()

c)Meal() Lunch() PortableLunch() Sandwich() Cheese()

d)Cheese() Sandwich() Meal() Lunch() PortableLunch()

ANS:A

Q) Consider the following code and choose the correct option:

class AA

{

int a;

AA(int a)

{

a=4;

}

}

class BBB extends AA

{

BBB()

{

super(3);

}

void displayA()

{

System.out.print(a);

}

public static void main(String args[])

{

new BBB().displayA();

}

}

a)compiles and display 0

b)compilation error

c)Compiles and display 4

d)Compiles and display 3

ANS:A\

Q) class Order1

{

Order1()

{

System.out.println("Cat");

}

public static void main(String... Args)

{

Order1 obj = new Order1();

System.out.println("Ant");

}

static

{

System.out.println("Dog");

}

}

consider the code above & select the proper output from the options.

a)Cat Ant Dog

b)Dog Cat Ant

c)Ant Cat Dog

d)none

ANS:B

Q) What will happen if a main() method of a "testing" class tries to access a private instance variable of an object using dot notation?

a)The compiler will automatically change the private variable to a public variable

b)The compiler will find the error and will not make a .class file

c)The program will compile and run successfully

d)The program will compile successfully, but the .class file will not run correctly

ANS:C

Q) What will be the result of compiling the following program?

**public** **class** MyClass

{

**long** var;

**public** **void** MyClass(**long** param)

{

var = param;

}

// (Line no 1)

**public** **static** **void** main(String[] args)

{

MyClass a, b;

a = **new** MyClass(); // (Line no 2)

}

}

/\*

a)The program will compile without errors.

b)A compilation error will occur at (Line no 2), since the class does not have a constructor that takes one argument of type int.

c)A compilation error will occur at (Line no 1), since constructors cannot specify a return value

d)A compilation error will occur at (2), since the class does not have a default constructor

ANS:A

Q) What would be the output?

**public** **class** MyClass1

{

**static** **void** print(String s, **int** i)

{

System.***out***.println("String: " + s + ", int: " + i);

}

**static** **void** print(**int** i, String s)

{

System.***out***.println("int: " + i + ", String: " + s);

}

**public** **static** **void** main(String[] args)

{

*print*("String first", 11);

*print*(99, "Int first");

}

}

a)String: String first, int: 11 int: 99, String: Int first

b)int: 27, String: Int first String: String first, int: 27

c)Compilation Error

d)Runtime Exception

ANS:A

Q)Here is the general syntax for method definition:

accessModifier returnType methodName( parameterList )

{

Java statements

return returnValue;

}

What is true for the returnType and the returnValue?

a)The returnValue can be any type, but will be automatically converted to returnType when the method returns to the caller

b)If the returnType is void then the returnValue can be any type

c)The returnValue must be the same type as the returnType, or be of a type that can be converted to returnType without loss of information

d)The returnValue must be exactly the same type as the returnType.

ANS:C

Q) **public** **class** Mud

{

// insert code here line 12

System.***out***.println("hi");

}

}

/\*

And the following five fragments:

public static void main(String...a){

public static void main(String.\* a){

public static void main(String... a){

public static void main(String[]... a){

public static void main(String...[] a){

//How many of the code fragments, inserted independently at line 12, compile?

a)3

b)1

c)2

d)0

e)4

f)5

ANS:A

Explanation:

public static void main(String...a){

public static void main(String... a){

public static void main(String[]... a){ are allowed

Q) **public** **class** MyClass2

{

**public** **static** **void** main(String[] args)

{

**double** x = 0.89;

Sphere sp = **new** Sphere();

// Some code missing

}

}

to get the radius value what is the code of line to be added ?

a)methodRadius(x);

b)sp.methodRadius(x);

c)Nothing to add

d)Sphere.methodRadius();

ANS:C

Q) A) A call to instance method can not be made from static context.

B) A call to static method can be made from non static context.

a)Only B is TRUE

b)Only A is TRUE

c)Both are TRUE

d)Both are FALSE

ANS:A

Q) Consider the following code and choose the correct option:

**public** **class** A

{

**private** **void** display()

{

System.***out***.print("Hi");

}

**public** **static** **void** main(String ar[])

{

display();

}

}

/\*

a)Compiles and displays Hi

b)Compiles and throws run time exception

c)Compilation fails

d)Compiles but doesn't display anything

ANS:C

Q)

**class** One

{

**int** var1;

One (**int** x)

{

var1 = x;

}

}

**class** Derived **extends** One

{

**int** var2;

Derived()

{

**super**(10);

var2=10;

}

**void** display()

{

System.***out***.println("var1="+var1+" , var2="+var2);

}

}

**class** Main

{

**public** **static** **void** main(String[] args)

{

Derived obj = **new** Derived();

obj.display();

}

}

/\*consider the code above & select the proper output from the options.

a)var1=10 , var2=10

b)0,0

c)compile error

d)runtime error

ANS:A

Q)

**public** **class** MyAr

{

**public** **static** **void** main(String argv[])

{

MyAr m = **new** MyAr();

m.amethod();

}

**public** **void** amethod()

{

**static** **int** i1 = 0;

System.***out***.println(i1);

}

}

/\*What is the Output of the Program?

a)Compile time error because i has not been initialized

b)Compilation and output of null

c)It is not possible to declare a static variable in side of non static method

or instance method. Because Static variables are class level dependencies.

ANS:C

Q)from 1 to 12

Q)from 75

Q) Which of the following statement gives the use of CLASSPATH?

a)Holds the location of Core Java Class Library (Bootstrap classes)

b)Holds the location of Java Extension Library

c) Holds the location of User Defined classes, packages and JARs

d) Holds the location of Java Software

ANS:B

Q) Which of the following options give the valid argument types for main() method? (Choose 2)

a)String[] args

b)String args[]

c)String [][]args

d)String args

ANS:A & B

Q) Which of the following are true about packages? (Choose 2)

a)Packages can contain only Java Source files

b)Packages can contain both Classes and Interfaces (Compiled Classes)

c)Packages can contain non-java elements such as images, xml files etc.

d)Sub packages should be declared as private in order to deny importing them

ANS:

Q)