1. What would be the output for the below code

public class ArgsDemo {

String company = "Hexaware";

public static void main(String[] args) {

System.out.println(args[0] + " " +args[2]);

}

}

Java ArgsDemo asil prajwal sashwat

**Asil Sashwat**

1. What is output

public class P1 {

public static void main(String[] args) {

int i=10;

// int j=i++;

int j = ++i;

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

}

}

**11 11**

1. What is the output for

class Demo {

static {

System.out.println("Demo Static...");

}

}

class C1 {

static {

System.out.println("C1 con");

}

public C1() {

System.out.println("Constructor C1");

}

}

class C2 extends C1 {

static {

System.out.println("C2 con");

}

public C2() {

System.out.println("Constructor C2");

}

}

public class P6 {

public static void main(String[] args) {

C2 obj = new C2();

}

}

1. What is the output for

Int[] a = new int[]{12,4,6,33,23};

for(int i : a) {

System.out.println(a);

}

Address will be printed 5 times

1. What is the return type for constructor (No Return Type)
2. Which statements are true
3. NullPointerException thrown when an object used without allocating memory
4. You can create custom exception class by extending the Exception class
5. Division by zero generates ArrayIndexOutOfBoundsException
6. An object that has not been allocated memory will holds a null value
7. **A,b,d**
8. Final class
9. Can be inherited
10. **Cannot be inherited**
11. Cannot create an object
12. Cannot have constructor

8)

interfaceIOne {

void f1();

public void f2();

protected void f3();

private void f4();

}

which lines will give compilation error

**protected and private, because interface methods are public and abstract by default.**

9) What is the output

int[] a=new int[6];

a[10]=66;

ArrayIndexOutOfBoundsException

10) public class Ex2 {

int a=18;

public static void show() {

System.out.println("Age " +a);

}

public static void main(String[] args) {

new Ex2().show();

}

}

**What is output Compile time error static methods will not access not static variables**

12) Identify which statements are true or false

Overloaded methods should differ in arg list (True)

Overloaded methods may differ in return type (False)

Overloaded methods may differ in access specifier (False)

13)

public class Ex3 {

public static void main(String[] args) {

String month;

month = "Feb";

switch(month) {

case "Feb" : System.out.println("Hi");

break;

case "Mar" : System.out.println("Hello");

break;

case "Apr" : System.out.println("Bye");

break;

case "May" : System.out.println("Test");

break;

default :

System.out.println("Default");

break;

}

}

}

**Hi**

14) What would be output

interface Mobile {

int memory=10;

public void calling();

}

interface Accessories extends Mobile {

public abstract void headPhone();

}

public class Ex3 implements Accessories {

public static void main(String[] args) { // line 4

public void headPhone() {

System.out.println("Hi");

}

}

}

Compile Time Error as interface methods are abstract by default, don’t define any abstract methods.

15) What would be output for

class Test {

void tests() {

System.out.println("Hi");

}

}

class Fest extends Test {

static void tests() {

System.out.println("Bye");

}

}

public class Ex3 {

public static void main(String[] args) { // line 4

Test obj = new Test();

obj.tests();

}

}

**Compile Time Error : as method in base class cannot be static method in derived class…**

16) What would be output for the below code

public class Ex3 {

static Ex3() { System.out.println("Hello");}

public static void main(String[] args) {

System.out.println("Hi");

}

}

**Compile Time Error as static Ex3() -> No static constructor in java, as static { } acts as static constructor**

17) What would be output

public class Ex3 {

public static void main(String[] args) {

String s1="abc";

String s2="abc";

System.out.println(s1==s2);

System.out.println(s1.equals(s2));

}

}

**True true**

18) To access a method in same package and other package derived classes, which modifier we want

**Public , protected**

19) public class Ex3 {

public Ex3() {}

public static void main(String[] args) {

}

}

What is public Ex3() here

**constructor**

A static method can be called as class method (T/F) **YES**

20)

public class Ex3 {

public Ex3() {}

public static void main(String[] args) {

String res = "i" + "like" + "java";

System.out.println(res);

}

}

**ilikejava**

22) What would be output

public class Ex3 {

static {

System.out.println("Hi");

}

static {

System.out.println("Bye");

}

}

**Program will compile, but will not execute**

**If you provide main(), then it prints both hi bye**

23)

interfaceIOne {

int a=10;

void draw();

}

public class Ex3 implements IOne {

public void draw() {}

public static void main(String[] args) {

System.out.println(a++);

}

}

**Compile time error as interface variables are final by default, so a++ will throw compile time error**

24) What

public class P9 {

static int count=0;

public void increment()

{

count++;

}

public static void main(String args[])

{

P9 obj1=new P9();

P9 obj2=new P9();

obj1.increment();

obj2.increment();

System.out.println("Obj1: count is="+obj1.count);

System.out.println("Obj2: count is="+obj2.count);

}

}

**2**

**2**

25) What would be output for

public class Hello extends String {

public static void main(String[] args) {

System.out.println("Hello World...");

}

}

**Compile time error : as String is final class cannot be inherited.**