Q1.Write in brief about OOPS Concept in java with Examples. (In your own words)

Object-Oriented Programming System (OOPs) is a programming concept that works on the principles of abstraction, encapsulation, inheritance, and polymorphism. It allows users to create objects they want and create methods to handle those objects. The basic concept of OOPs is to create objects, re-use them throughout the program, and manipulate these objects to get results.

Suppose, we have created a class called My book, we specify the class name followed by the object name, and we use the keyword new.

```
Public class Mybook {
int x=10;
Public static void main (String args []) {
Mybook Myobj= new Mybook ();
System.out.println(MyObj.x);
}
}
```

In the above example, a new object is created, and it returns the value of x which may be the number of books.

Mybook Myobj= new Mybook ();

This is the statement used for creating objects.

System.out.println(Myobj.x);

This statement is used to return the value of x of an object.

Q2.Write simple programs(wherever applicable) for every example given in Answer 2.

Multiple Choice Questions

Q1. Which of the following is used to make an Abstract class? A. Making at least one member function as pure virtual function B. Making at least one member function as virtual function C. Declaring as Abstract class using virtual keyword D. Declaring as Abstract class using static keyword

Ans A,B

An abstract class contains at least one pure virtual function. You declare a pure virtual function by using a pure specifier (= 0) in the declaration of a virtual member function in the class declaration.

Q2. Which of the following is true about interfaces in java.

1) An interface can contain the following type of members. ....public, static, final fields (i.e., constants) ....default and static methods with bodies

2) An instance of the interface can be created.

3) A class can implement multiple interfaces.

4) Many classes can implement the same interface.

A. 1, 3 and 4

B. 1, 2 and 4

C. 2, 3 and 4 D. 1,2,3 and 4

Ans-A

Explanation: The instance of an interface can't be created because it acts as an abstract class.

Q3. When does method overloading is determined?

A. At run time

B. At compile time

C. At coding time

D. At execution time

Ans-B

Overloading is determined at compile time. Hence, it is also known as compile time polymorphism.

Q4.What is the number of parameters that a default constructor requires? A. 0 B. 1 C. 2 D. 3

Ans-A

A default constructor is a constructor that either has no parameters, or if it has parameters, all the parameters have default values.

Q5.To access data members of a class, which of the following is used?

A. Dot Operator B. Arrow Operator C. A and B both as required D. Direct call

Ans-A

Dot operator is used to access the members with help of object of class.

Q6.Objects are the variables of the type ____?

A. String B. Boolean C. Class D. All data types can be included

Ans- C

Objects are the variables of the type Class.

Q7.A non-member function cannot access which data of the class?

A. Private data B. Public data C. Protected data D. All of the above

Ans-A

Generally, non-member functions cannot access the private members of a particular class.

Q8. Predict the output of following Java program

```
class Test {
 int i;
 }
class Main {
 public static void main(String args[]) {
 Test t = new Test(); System.out.println(t.i);
 }
 }
```

A. garbage value

B. 0

C. compiler error

D. runtime Error

Ans-C

Explanation:
t is just a reference, the object referred by t is not allocated any memory. Unlike C++, in Java all non-primitive objects must be explicitly allocated and these objects are allocated on heap. The following is corrected program.

Q9.Which of the following is/are true about packages in Java?

1) Every class is part of some package.

2) All classes in a file are part of the same package.

3) If no package is specified, the classes in the file go into a special unnamed package

4) If no package is specified, a new package is created with folder name of class and the class is put in this package.

A. Only 1, 2 and 3 B. Only 1, 2 and 4 C. Only 4 D. Only 1, 3 and 4

Ans-A

For Q10 to Q25 find output with explanation.

Q10.Predict the Output of following Java Program.

```
class Base {
 public void show() {
System.out.println("Base::show() called");
 } }
 class Derived extends Base { public void show() {
 System.out.println("Derived::show() called");
 } }
 public class Main {
 public static void main(String[] args) {
```

```
 Base b = new Derived();; b.show();

 } }
```

Ans- Derived::show() called

Q11. What is the output of the below Java program?

```
class Base { final public void show() {

 System.out.println("Base::show() called");

 } }

class Derived extends Base {

 public void show() {

 System.out.println("Derived::show() called");

 } }

class Main {

 public static void main(String[] args) {

 Base b = new Derived();; b.show();

 } }
```

Compiler Error

Q12.Find output of the program.

```
 class Base {

public static void show() {

 System.out.println("Base::show() called");

 }

 }
 class Derived extends Base {
 public static void show() {
System.out.println("Derived::show() called");
 }
 }
 class Main {
 public static void main(String[] args) {
 Base b = new Derived();
 b.show();
 }
 }
```

```
java -cp /tmp/AiGt7stCQD Main
Base::show() called
```

Q13.What is the output of the following program?

```
 class Derived {
 public void getDetails()
 { System.out.printf("Derived class ");
 } }
 public class Test extends Derived {
 public void getDetails()
 { System.out.printf("Test class ");
super.getDetails();
} public static void main(String[] args) {
```

```
Derived obj = new Test();
 obj.getDetails();
 } }
```

Ans –Compilation error

Q14. What is the output of the following program?

```
class Derived {
 public void getDetails(String temp) {
System.out.println("Derived class " + temp);
 } }
 public class Test extends Derived {
public int getDetails(String temp) {
System.out.println("Test class " + temp);
 return 0;
 }
 public static void main(String[] args) {
 Test obj = new Test();
 obj.getDetails("Name");
 } }
```

Compilation error

Explanation: Final and static methods cannot be overridden

Q15.What will be the output of the following Java program?

```
class test {
public static int y = 0;
 }
 class HasStatic {
```

```
 private static int x = 100;
 public static void main(String[] args) {
 HasStatic hs1 = new HasStatic();
 hs1.x++;
HasStatic hs2 = new HasStatic();
 hs2.x++;
 hs1 = new HasStatic();
 hs1.x++;
 HasStatic.x++;
System.out.println("Adding to 100, x = " + x);
 test t1 = new test();
t1.y++;
test t2 = new test();
 t2.y++; t1 = new test();
 t1.y++;
System.out.print("Adding to 0, ");
 System.out.println("y = " + t1.y + " " + t2.y + " " + test.y);
 }
 }
```

```
java -cp /tmp/9zEbpzyB1b HasStatic
Adding to 100, x = 104Adding to 0, y = 3 3 3
```

Q16.Predict the output class San { public void m1 (int i,float f) { System.out.println(" int float method"); } public void m1(float f,int i); { System.out.println("float int method"); } public static void main(String[]args) { San s=new San(); s.m1(20,20); } }

Ans_ compilation error

While resolving overloaded method, compiler automatically promotes if exact match is not found. But in this case, which one to promote is an ambiguity.

Q17.What is the output of the following program? public class Test { public static void main(String[] args) { int temp = null; Integer data = null; System.out.println(temp + " " + data); } }

Ans-Compilation error

Q18.Find output class Test { protected int x, y; } class Main { public static void main(String args[]) { Test t = new Test(); System.out.println(t.x + " " + t.y); } }

Ans- 00

Q19.Find output // filename: Test2.java class Test1 { Test1(int x) { System.out.println("Constructor called " + x); } } class Test2 { Test1 t1 = new Test1(10); Test2(int i) { t1 = new Test1(i); } public static void main(String[] args) { Test2 t2 = new Test2(5); } }

java -cp /tmp/XvQpy6molY Test2

Constructor called 10

Constructor called 5

Q20.What will be the output of the following Java program? class Main { public static void main(String[] args) { int []x[] = {{1,2}, {3,4,5}, {6,7,8,9}}; int [][]y = x; System.out.println(y[2][1]); } }

Ans-

java -cp /tmp/oUSyRly4Tu Main

7

Q21.What will be the output of the following Java program?

class A { int i; public void display() { System.out.println(i); } } class B extends A { int j; public void display() { System.out.println(j); } } class Dynamic_dispatch { public static void main(String args[]) { B obj2 = new B(); obj2.i = 1; obj2.j = 2; A r; r = obj2; r.display(); } }

java -cp /tmp/cNWEJNTMWd Dynamic_dispatch

2

Q22. What will be the output of the following Java code?

```
}

class A
{
int i;
void display()
{
System.out.println(i);
}
}
class B extends A
{
int j;
void display()
{
System.out.println(j);
}
}
class method_overriding
{
public static void main(String args[])
{
B obj = new B();
obj.i=1;
obj.j=2;
obj.display();
}
```

```
}
```

```
java -cp /tmp/V9wAOgZw9u method_overriding

2
```

Q23.What will be the output of the following Java code?

```
class A { public int i; protected int j; } class B extends A { int j; void display() { super.j = 3;
System.out.println(i + " " + j); } } class Output { public static void main(String args[]) { B
obj = new B(); obj.i=1; obj.j=2; obj.display(); } }
```

```
java -cp /tmp/DSLf2Y2Z2o Output

1 2
```

Q24.What will be the output of the following Java program?

```
class A

{

public int i;

public int j;

A()

{

i = 1;

j = 2;

}

}

class B extends A

{

int a;

B()

{
```

```
super();

}

}

class super_use

{

public static void main(String args[])

{

B obj = new B();

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

}

}
```

Ans -12

Q 25. Find the output of the following program

```
class Test { int a = 1; int b = 2; Test func(Test obj) { Test obj3 = new Test(); obj3 = obj; obj3.a = obj.a++ +
++obj.b; obj.b = obj.b; return obj3; } public static void main(String[] args) { Test obj1 = new Test(); Test
obj2 = obj1.func(obj1); System.out.println("obj1.a = " + obj1.a + " obj1.b = " + obj1.b);
System.out.println("obj2.a = " + obj2.a + " obj1.b = " + obj2.b); } }
```

```
java -cp /tmp/nSSV0OjXHz Test

obj1.a = 4 obj1.b = 3

obj2.a = 4 obj1.b = 3
```