

FULL STACK DEVELOPMENT – WORKSHEET 4

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

OOPs stands for Object-oriented programming. OOPs in Java organizes a program around the various objects and well-defined interfaces. The OOPs Concepts in Java are abstraction, encapsulation, inheritance, and polymorphism. These concepts aim to implement real-world entities in programs.

The four basics of OOP are abstraction, encapsulation, inheritance, and polymorphism. These are the main ideas behind Java's Object-Oriented Programming.

OOPs, concepts in Java is known as object-oriented programming System. Here's the OOPs concepts in Java with examples:

1. Class
2. Object
3. Inheritance
4. Polymorphism
5. Abstraction
6. Encapsulation
7. association
8. Aggression
9. Composition

mainly used in different object-oriented programming languages such as Java, C#, C++, Python, Perl, Ruby, etc.

Object Example:

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

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

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. Making at least one member function as pure virtual function.

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

Ans. 1, 3 and 4

Q3. When does method overloading is determined?

Ans. At compile time

Q4.What is the number of parameters that a default constructor requires?

Ans. Zero(0)

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

Ans. A and B both as required

Q6.Objects are the variables of the type ____?

Ans. Class

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

Ans Private data

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);  
}  
}
```

Ans. Compiler error.

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.

Ans. Only 1, 2 and 3

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. Compiler error

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();  
    }  
}
```

```
}
```

Ans compiler error

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. Test class Derived class

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");
}
}

```

Ans. Compiler error

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);
}
}

```

```
}  
}
```

Ans. Adding to 100, x = 104

Adding 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. Compile time error

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 due to temp

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. 0 0

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);  
}  
}
```

Ans. 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.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();
}
}
```

Ans.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();
}
}
```

Ans.12

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);
}
}

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