Name: Vaibhav Chopade J-30

Section A

1) ⇒ b) for(int i=7; i<=77; i+=7);

2) ⇒ ~~b) for(int i=99; i>=0; i/9)~~

3) ⇒

```
class Test
{
    public static void main(String[] args)
    {
        for (int i = 0; true ; i++)
        {
            System.out.println("Hello");
            break;
        }
    }
}
```

⇒ output is = Hello.. ✓ (2)

3) ⇒ error ✓ (2)

4) ⇒ output is = 20. ✓ (2)

5) ⇒ output = false. ✓

Section B

1) what is class and object in java ?

ans

class :- The class is blueprint of object.
- we can create as many class object.
- In the class there are fun^n, variable functions, variable, members - fun^n, method are included.

Object : object is real world entity that have behaviour, name or state.

ex

```
public class Main {
    public void input() {
        System.out.println("Enter name");
        String name = new Scanner(System.in).
                                .nextLine();
    }
    void display()
    {
        System.out.println(name);
    }
}
public static void main(String[] args)
{
    Main ob = new Main();
    ob.input();
    ob.display();
}
}
```

(10)

Syntax of class

class class_Name {
}

Syntax of object.

className objectOfClass = new className();

→ Class has file
→ In the above code.
public class Main ()
↳ this is our class name.

Main ob = new Main ();
↳ here we can create an object of class.

Q 2) What is Constructor?

→ So, A Constructor is a special mechanism feature of object oriented programming language

Def" = Constructor is as same name same name of class.
- at the time of object creation it automatically invoked.
- we can use constructor for object creation.
- it does not have any return type.

There 3 types of Constructor.
① Default Constructor.
which has without parameter.

② Parameterise Constructor.
constructors having parameters.

③ Copy constructor.
→ it copies one constructor to another

→ but there's no copy constructor in Java, we can just

Eg

class student {
private int id;
private string name;
private int rollno;

student ()
{
System.out.println ("Default Constructor");
}

student (int id, string name)
{
System.out.println ("parameterise constructor");
this.id = id;
this.name = name;
System.out.println (id + " " + name);
}

Name:
Class:

```
public static void main (String [] args)
    {
        Student s = new Student ();
        s.studen ();
        s.student (101, "Vaibha");
    }
}
```

3) Make user defined exception checked exception for negative number.

```
class NegativeNumException extends Exception {
    int num;
    { String s;
      super (s);
    }
    public void isValid (int num) throws NegativeNum
                                         Exception
    { if (num > 0) {
        try {
        throw new = NegativeNumException ("Valid
        }
        catch (NegativeNumExcep e)
        {
            e.printStackTrace ();
        }
}
Public class TestException {
    Public Static void main (String [] args) throws
                                    NegativeNumExcep
    {
        int num;
        isValid (5);
        isValid (-6);
        System.out.println (isValid);
}
```

4) Explain One D Array and Two D array with example.

→ Array is The homogeneous collection of similar data types is called array.

One D Array

→ It also called as single dimensional array.
→ array having one [] called One D array.

Syntax

```
int []a = {1, 2, 3, 4};
```

```
public class Array {
    public Static void main (String []args)
    {
        int arr [] = {1, 2, 3, 4};
        for (int i=0; i<4; i++)
        {
            System.out.println (a[i]);
        }
    }
```

Two D Array

→ It called as multidimenssional array
→ In which array having two [][]

Syntax

```
int arr[][] = {{1, 2},
               {2, 3},
               {3, 4}
              };
```

Ex

```
Public class Array {
    public static void main (String [] args)
    {
    int [][] a = { {1, 2, 3},
                   {2, 4, 5},
                   {5, 6, 6} };

    for (int i=0; i<=a.length ; i++)
      {
        for ( int j=0; j<= a[i].length; i++)
        {
          sout System.out.print (a[i][j]);
        }
    }
```

Q) 6) what is Inheritance? why multiple inheritance is not supported in java?

⇒ - one class Inherit or aquires the property and method of another class called as Inheritance.
- one of the most important feature of object-oriented programming.

There are 5 TYPES of Inheritance.

① Single level :- In that Inheritance one base & one child class.

Ex

② multilevel Inheritance :- In this Inheritance one base is inherited by more than one Derived class.
⇒ The Inheritance having one base class and more than on subclass.

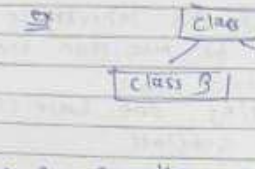
Ex ⇒

③ Hierachical Inheritance :-
⇒ In this inheritance one base class and ~~more than~~ multiple Derived class are present

Ex.

⇒ why java does not support multiple Inheritance?
⇒ due to multiple inheritance compiler get Confuse which object ~~class~~ it will be compile. and this situation is called "ambiguity"
⇒ ambiguity ⇒ Compiler get Confused at the time of compilation.

~~In multiple Inheritance two class having same object of different~~



⇒ So compiler get confused because of which object is which class.
⇒ multiple classes having same object.
⇒ These is all about Inheritance.

Q5) Encapsulation ⇒ The process in which wrapping up data and member fun^n in a single entity is called ~~inter~~ Encapsulation.

```
Public class Encapsulation{
    Private int roll;
    Privat String name;
        public void setroll(){
            this.roll = roll;
        }
        public int getroll(){
            return roll;
        }
        public void setname(){
            this.name = name;
        }
        public String getname(String name)
            return name;
    }
public static void main(String[] args)
    {
        Encapsulation e = new Encapsulation();
        e.setroll(); e.setname();
System.out.print(e.getroll(101));
System.out.println(e.getname("vaibhav"));
}
```

Section C

1) Java Program to Find Amstrong number.

```
import java.util.Scanner;
public class Amstrong{
    Public static void main(String[] args)
    {
        System.out.println("enter Number");
        int n = new Scanner(System.in).nextInt();
        int temp;
        temp = n;
        int p = 0;
    while(n > 0)
        {
            int rem = n % 10;
            p = (p) + rem * rem * rem;
            n = n / 10;
        }
        if( temp == p)
        {
            System.out.println(n + " is Amstrong");
        }
        else
        {
            System.out.println(n + " is Not Amstrong");
        }
    }
}
```

⇒ ~~output = 153 is Amstrong.~~
Input = 123
Output = 123 is not Amstrong Number.

Input = 153
Output = 153 is Amstrong Number.

2) Java program to find factorial of a number.

```
import java.util.Scanner;
public class Factorial {
    public static void main(String [] args)
    { System.out.println("Enter number");
      int f = 0;
      int n = new Scanner(System.in).nextInt();
      int f=1;
      for(i=1; i<n; i++)
      {
          f = f*i;
      } System.out.println(i);
      System.out.println("factorial of number"+n
                         +" is" + f );
    }
}
```

(8)

3) Prime number in Java.

```
import java.util.Scanner;
public class Prime {
  public static void main (String [] args)
  { Scanner sc = new Scanner (System.in);
    System.out.println("enter number");
    int n= sc.nextInt();
    for(int i=1; i<=n; i++)
    {
    int temp=;
    for(int i=2; i<=n-1; i++)
    {
      if(n % i==0)
      {
        temp = 0
      }
    }
    if(temp ==0)
    {
      System.out.println(n);
    }
}
```

(3)

Double

3) Prime in range.

```
import java.util.Scanner;
public class Prime {
    public static void main (String [] args){
      int n= new Scanner(System.in).nextInt();
    for(int i=1; i<=n; i++)
    {
      int temp = 1
      for(int j=2; j<=n-1; j++)
      {
        if(n%j==0)
        {
          temp=0
          temp = temp+1;
        }
      }
      if(temp==0)
      {
        System.out.println(n);
      }
    }
}
```

4) Java program calculate Compound Interest.

```
public class CompoundInterest {
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);

        int CI;
        int p = ;
        System.out.println("Enter principal amount");
    double p = sc.nextDouble();
        System.out.println("Enter time ");
        float time = sc.nextFloat();
    System.out.println("Enter Rate on Amount");
        float rate = sc.nextFloat();

    CI = p *(Math.pow((1+rate/100), time));

    System.out.println("Compound Interest"
                        + CI);

    }
}
```