

## Practical 1

Aim: Java Basics

A) Write a Java program that takes a number as input and prints its multiplication table upto 10.

Source code:

```
import java.util.*;
class program
{
    public static void main(String[] args)
    {
        Scanner s=new Scanner(System.in);
        System.out.print("Enter a no.");
        int a=s.nextInt();
        for(int i=1;i<=10;i++)
        {
            System.out.println(a*i);
        }
    }
}
```

B) Write a Java program to display the following pattern.

```
*
* *
* * *
* * * *
* * * * *
```

Source code:

```
class pattern
{
    public static void main(String[] args)
    {
        for(int i=0;i<=3;i++)
        {
            for(int j=0;j<=i;j++)
            {
                System.out.print("* ");
            }
            System.out.println();
        }
    }
}
```

C) Write a Java program to print the area and perimeter of a circle.

Source code:

```
import java.util.*;
class circle
{
    public static void main(String[] args)
    {
        Scanner s=new Scanner(System.in);
        System.out.print("Enter the radius: ");
        int r=s.nextInt();
        double area=3.14*r*r;
        double cir=2*3.14*r;
```

```

System.out.println("Area of circle: "+area);
System.out.println("Circumference of circle: "+cir);
}
}

```

## Practical 2

Aim: Use of Operators

A) Write a Java program to add two binary numbers.

Source code:

```

import java.util.*;
class addition
{
    public static void main(String[] args)
    {
        String a,b;
        Scanner s=new Scanner(System.in);
        System.out.print("Enter first binary no.");
        a=s.nextLine();
        System.out.print("Enter second binary no.");
        b=s.nextLine();
        int num1=Integer.parseInt(a,2);
        int num2=Integer.parseInt(b,2);
        int sum=num1+num2;
        System.out.println(Integer.toBinaryString(sum));
    }
}

```

B) Write a Java program to convert a decimal number to binary number and vice versa.

Source code:

```

import java.util.*;
import java.lang.Math;
class bd
{
    public static int BtoD(int n)
    {
        int rem,ans=0;
        int i=0;
        while(n!=0)
        {
            rem=n%10;
            n=n/10;
            ans+=rem*Math.pow(2,i);
            ++i;
        }
        return ans;
    }

    public static long DtoB(int n)
    {
        long bn=0;
        int i=1,rem;
        while(n!=0)
        {
            rem=n%2;
            n=n/2;
            bn+=rem*i;
        }
    }
}

```

```

i*=10;
}
return bn;
}
public static void main(String[] args)
{
System.out.println(BtoD(1101));
System.out.println(DtoB(141));
}
}

```

C) Write a Java program to reverse a string.

Source code:

```

import java.util.*;
class reverse
{
public static void main(String[] args)
{
Scanner s=new Scanner(System.in);
System.out.print("Enter a string: ");
String a=s.nextLine();
char[] c=a.toCharArray();
int l=c.length;
String rev="";
for(int i=l-1;i>=0;i--)
{
rev=rev+c[i];
}
System.out.print("Reverse of string "+a+" is "+rev);
}
}

```

### Practical 3

Aim: Java Data Types

A) Write a Java program to count the letters, spaces, numbers and other characters of an input string.

Source code:

```

import java.util.*;
class count
{
public static void main(String[] args)
{
Scanner s=new Scanner(System.in);
System.out.print("Enter a string: ");
String a=s.nextLine();
char[] c=a.toCharArray();
int l=c.length;
int digit=0,alpha=0,space=0,other=0;
for(int i=0;i<l;i++)
{
if(Character.isLetter(c[i]))
{
alpha++;
}
else if(Character.isDigit(c[i]))
{

```

```

digit++;
}
else if(Character.isSpaceChar(c[i]))
{
space++;
}
else
{
other++;
}
}
System.out.println("No. of characters: "+alpha);
System.out.println("No. of digits: "+digit);
System.out.println("No. of spaces: "+space);
System.out.println("No. of other characters: "+other);
}
}

```

B) Implement a Java function that calculates the sum of digits for a given char array consisting of the digits '0' to '9'. The function should return the digit sum as a long value.

Source code:

```

import java.util.*;
class sum
{
public static void main(String[] args)
{
Scanner s=new Scanner(System.in);
System.out.print("Enter a no.: ");
String a=s.nextLine();
char[] c=a.toCharArray();
int l=c.length;
total(c);
}
public static void total(char[] c)
{
int ans=0,l=c.length;
for(int i=0;i<l;i++)
{
ans=ans+Character.getNumericValue(c[i]);
}
System.out.println("Sum of digits in array: "+ans);
}
}

```

C) Find the smallest and largest element from the array.

Source code:

```

import java.util.*;
class array
{
public static void main(String[] args)
{
Scanner s=new Scanner(System.in);
System.out.print("Enter size of array: ");
int size=s.nextInt();
int[] a=new int[size];
System.out.print("Enter "+size+" elements: ");
for(int i=0;i<size;i++)

```

```

{
a[i]=s.nextInt();}
int max=a[0];
int min=a[0];
for(int i=1;i<size;i++)
{
if(a[i]>max)
{
max=a[i];
}
if(a[i]<min)
{
min=a[i];
}
}
System.out.println("Maximum value is: "+max);
System.out.println("Minimum value is: "+min);
}
}

for(int j=i+1;j<a.length;j++)
{
if(a[i]<a[j])
{
int t;
t=a[i];
a[i]=a[j];
a[j]=t;
}
}
System.out.println(a[i]+" ");
}
}
}

```

B) Design a class that demonstrates the use of constructor and destructor.

Source code:

```

import java.lang.*;
class area
{
public area(int r)
{
System.out.println("Area of a circle: "+(3.14*r*r));
}
public area(int l,int b)
{
System.out.println("Area of a rectangle: "+(l*b));
}
}
class program
{
public static void main(String[] args)
{
area s=new area(16);

area s1=new area(15,15);
}
}

```

```
}
```

C) Write a java program to demonstrate the implementation of abstract class.

Source code:

```
abstract class p4c
{
public p4c()
{
System.out.println("Bike");
}
public void gear()
{
System.out.println("Gear");
}
abstract void speed();
}
class vehicle extends p4c
{
void speed()
{
System.out.println("Speed");
}
}
class program
{
public static void main(String[] args)
{
vehicle v=new vehicle();
v.gear();
v.speed();
}
}
```

## Practical 5

Aim: Inheritance

A) Write a java program to implement single level inheritance.

Source code:

```
import java.util.*;
class display
{
public void disp()
{
System.out.print("Area of circle: ");
}
}
class area extends display
{
public void area1(int r)
{
System.out.println(3.14*r*r);
}
}
class program
{
public static void main(String[] args)
{
```

```

Scanner s=new Scanner(System.in);
System.out.print("Enter radius: ");
int r=s.nextInt();
area a=new area();
a.disp();
a.area1(r);
}
}

```

B) Write a java program to implement method overriding.

Source code:

```

class bank
{
int rateOfInterest()
{
return 0;
}
}
class axis extends bank
{
int rateOfInterest()
{
return 7;
}
}
class icici extends bank
{
int rateOfInterest()
{
return 9;
}
}
class method
{
public static void main(String[] args)
{
bank b=new bank();
System.out.println(b.rateOfInterest());
axis a=new axis();
System.out.println(a.rateOfInterest());
icici i=new icici();
System.out.println(i.rateOfInterest());
}
}

```

C) Write a java program to implement multiple inheritance.

Source code:

```

interface vone
{
int speed=100;
public void speed();
}
interface vtwo
{
int distance=500;
public void distance();
}

```

```

class vehicle implements vone,vtwo
{
public void speed()
{
System.out.println("Speed: "+speed);
}
public void distance()
{
System.out.println("Distance: "+distance);
}
}
class multiple
{
public static void main(String[] args)
{
vehicle v=new vehicle();
v.speed();
v.distance();
}
}

```

## Practical 6

Aim: Packages and Arrays

A) Create a package, Add the necessary classes and import the package in java class.

Source code:

mathmethods.java-  
package Mathematics;

```

public class mathmethods
{
public static int sqr(int r)
{
return r*r;
}
public static int cube(int a)
{
return a*a*a;
}
}

```

packagedemo.java-  
import java.util.\*;

```

import Mathematics.*;
class packagedemo
{
public static void main(String[] args)
{
Scanner s=new Scanner(System.in);
System.out.println("ENTER NO FOR SQUARE");
int n=s.nextInt();
System.out.println("ENTER NO FOR CUBE");
int a=s.nextInt();
System.out.println("Square = "+Mathematics.mathmethods.sqr(n));
}
}

```



```

System.out.println("Cube = "+Mathematics.mathmethods.cube(a));
}
}

```

B) Write a java program to add two matrices and print the resultant matrix.

Source code:

```

import java.util.*;
class addition
{
    public static void main(String[] args)
    {
        int a[][]=new int [3][3];
        int b[][]=new int [3][3];
        int c[][]=new int [3][3];
        Scanner s=new Scanner(System.in);
        System.out.println("Enter the elements for Matrix 1");
        for(int i=0;i<3;i++)
        {
            for(int j=0;j<3;j++)
            {
                a[i][j]=s.nextInt();
            }
        }
        System.out.println("Enter the elements for Matrix 2");
        for(int i=0;i<3;i++)
        {
            for(int j=0;j<3;j++)
            {
                b[i][j]=s.nextInt();
            }
        }
        System.out.println("Addition of Matrix 1s");
        for(int i=0;i<3;i++)
        {
            for(int j=0;j<3;j++)
            {
                c[i][j]=a[i][j]+b[i][j];
                System.out.print(c[i][j]+"\\t");
            }
            System.out.println();
        }
    }
}

```

Vectors and MultithreadingC) Write a java program for multiplying two matrices and print the product for the same.

Source code:

```

import java.util.*;
class mult
{
    public static void main(String[] args)
    {
        int a[][]=new int [3][3];
        int b[][]=new int [3][3];
        int c[][]=new int [3][3];
        int sum=0;
        Scanner s=new Scanner(System.in);

```



```
v.remove(2);  
System.out.println("Is march present: "+v.contains("march"));  
v.setSize(50);  
System.out.println("Size: "+v.size());  
}  
}
```