

1) Write a Java program to display default value of all primitive data type of JAVA

Program:

```
class Default
{
    static int var1;
    static float var2;
    static long var3;
    static String var4;
    static boolean var5;
    static double var6;
    public static void main( String[ ] args)
    {
        System.out.println("var1="+var1);
        System.out.println("var2="+var2);
        System.out.println("var3="+var3);
        System.out.println("var4="+var4);
        System.out.println("var5="+var5);
        System.out.println("var6="+var6);
    }
}
```

2) Write a Java program to find the discriminant value D and find out the roots of the quadratic equation of the form $ax^2+bx+c=0$.

Problem:-

```
import java.util.Scanner;

class Quadratic
{
    public static void main( String[ ] args)
    {
        double a,b,c,r1,r2;

        Scanner sc=new Scanner(System.in);

        System.out.println("Enter the value of a");
        a=sc.nextInt();

        System.out.println("Enter the value of b");
        b=sc.nextInt();

        System.out.println("Enter the value of c");
```

```

c=sc.nextInt();
double d=b*b-4*a*c;
double e=2*a;
if(d>0)
{
    r1=(-b+Math.sqrt(d))/e;
    r2=(-b-Math.sqrt(d))/e;
    System.out.println("root1="+r1+" "+"root2="+r2);
    System.out.println("the root are real and distinct");
}
else if(d==0)
{
    r1=r2=-b/e;
    System.out.println("root1="+r1+" "+"root2="+r2);
    System.out.println("the root are equal");
}
else{
    r1=-b/e;
    r2=(Math.sqrt(-d))/e;
    System.out.println("root1="+ r1+ "+" + r2+ "i");
    System.out.println("root2="+ r1+ "-" + r2+ "i");
    System.out.println("The root are real and imaginary");
}
}
}
}

```

3. Five Bikers Compete in a race such that they drive at a constant speed which may or may not be the same as the other. To qualify the race, the speed of a racer must be more than the

average speed of all 5 racers. Take as input the speed of each racer and print back the speed of qualifying racers.

Program

```
import java.util.Scanner;

class Bike
{
    public static void main(String[] args)
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the speed of 5 biker racer");
        double r1=sc.nextInt();
        double r2=sc.nextInt();
        double r3=sc.nextInt();
        double r4=sc.nextInt();
        double r5=sc.nextInt();
        double avg=(r1+r2+r3+r4)/5;
        System.out.println("Average speed is" + avg);
        System.out.println("Qualify racer are:- ");
        if(r1>avg)
        {
            System.out.println("race1 speed is" + r1);
        }
        if(r2>avg)
        {
            System.out.println("race2 speed is" + r2);
        }
        if(r3>avg)
        {
```

```

        System.out.println("race3 speed is" + r3);
    }
    if(r4>avg)
    {
        System.out.println("race4 speed is" + r4);
    }
    if(r5>avg)
    {
        System.out.println("race5 speed is" + r5);
    }
}
}

```

3. Write a Java program to select all the prime numbers within the range of 1 to 100.

Program

```

class Prime
{
    public static void main(String[] args)
    {
        for(int i=1; i<=100; i++)
        {
            int cnt=0;
            for(int j=1; j<=i; j++)
            {
                if(i%j==0)
                {
                    cnt++;
                }
            }
        }
    }
}

```

```

        if(cnt==2)
        {
            System.out.print(i+ " ");
        }
    }
}
}

```

4) Write a Java program to Find the sum of all even terms in the Fibonacci sequence up to the given range N.

```

import java.util.Scanner;

class Fibonacci
{
    public static void main(String[] args)
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("enter a number");
        int n=sc.nextInt();
        int a=0, b=1,c,sum=0;
        c=a+b;
        while(c<n)
        {
            if(c%2==0)
                sum=sum+c;
            a=b;
            b=c;
            c=a+b;
        }
    }
}

```

```

        System.out.println(sum);
    }

}

```

5) Write a Java program to check whether a given number is Armstrong or not.

Program

```

import java.util.Scanner;
class Armstrong
{
    public static void main(String[] args)
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("enter a number");
        int n=sc.nextInt();
        int c=0;
        int a=n;
        int b=n;
        int sum=0;
        while(n!=0)
        {
            int r=n%10;
            c++;
            n=n/10;
        }
        while(b!=0)
        {
            int rem=b%10;
            sum=sum+(int)Math.pow(rem,c);
            b=b/10;
        }
        if(sum==a)
        {
            System.out.println("Armstrong");
        }
        else{
            System.out.println("Not Armstrong");
        }
    }
}

```

6) Write a Java program to implement binary search.

Program

```
import java.util.Scanner;

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

        System.out.println("Enter a number");

        int n=sc.nextInt();

        int a[]=new int[n];

        System.out.println("Enter array elements");

        for(int i=0; i<a.length; i++)
        {
            a[i]=sc.nextInt();
        }

        System.out.println("Enter a elements to be searched");

        int ser=sc.nextInt();

        int l=0, h=n-1;

        while(l<=h)
        {
            int mid=(l+h)/2;

            if(a[mid]==ser)
            {
                System.out.println("Element is found at index" +mid);

                break;
            }

            else if(a[mid]<ser)
            {
                l=mid+1;
            }
        }
    }
}
```

```

        else{
            h=mid-1;
        }
    }
    if(l>h)
    {
        System.out.println("sorry!! Element is not found");
    }
}
}

```

USING PACKAGE for implementing binary search

```

import java.util.Scanner;
import java.util.Arrays;
public class Main
{
    public static void main(String[] args)
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter a number");
        int n=sc.nextInt();
        int a[]=new int[n];
        System.out.println("Enter array elements");
        for(int i=0; i<a.length; i++)
        {
            a[i]=sc.nextInt();
        }
        System.out.println("Enter a element to be searched");
    }
}

```



```

        int ser=sc.nextInt();

        System.out.println(ser +" is found at index="+ Arrays.binarySearch(a,ser));
    }
}

```

USING METHOD(FUNCTION) for implementing binary search

```

import java.util.Scanner;

public class Main
{
    public static void binarysearch(int a[], int n, int ser)
    {
        int l=0, h=n-1;
        while(l<=h)
        {
            int mid=(l+h)/2;
            if(a[mid]==ser)
            {
                System.out.println("Element is found at index " +mid);
                break;
            }
            else if(a[mid]<ser)
            {
                l=mid+1;
            }
            else{
                h=mid-1;
            }
        }
    }
}

```

```

        if(l>h)
        {
            System.out.println("sorry!! Element is not found");
        }
    }

    public static void main(String[] args)
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter a number");
        int n=sc.nextInt();
        int a[]=new int[n];
        System.out.println("Enter array elements");
        for(int i=0; i<a.length; i++)
        {
            a[i]=sc.nextInt();
        }
        System.out.println("Enter a elements to be searched");
        int ser=sc.nextInt();
        binarysearch(a,n,ser);
    }
}

```

7) **Write a Java program to sort elements in a given list of elements using bubble sort.**

Program

```

import java.util.Scanner;

public class Bubble
{
    public static void main(String[] args)
    {

```

```

        Scanner sc=new Scanner(System.in);

        System.out.println("Enter a number");

        int n=sc.nextInt();

        int a[]=new int[n];

        System.out.println("Enter array elements");

        for(int i=0; i<a.length; i++)

        {

            a[i]=sc.nextInt();

        }

        for(int i=0; i<n; i++)

        {

            for(int j=0; j<n-i-1; j++)

            {

                if(a[j]>a[j+1])

                {

                    int temp=a[j];

                    a[j]=a[j+1];

                    a[j+1]=temp;

                }

            }

        }

        for(int i=0; i<n; i++)

        {

            System.out.print(a[i]+ " ");

        }

    }

}

```

8) Write a Java program to sort an element in a given list of elements using merge

```
import java.util.Scanner;

public class Main
{
    public static void merge(int a[], int lb, int mid, int ub)
    {
        int p=ub-lb+1;
        int b[] = new int[p];
        int i=lb;
        int j=mid+1;
        int k=0;
        while(i<=mid && j<=ub)
        {
            if(a[i]<a[j])
            {
                b[k]=a[i];
                i++;
                k++;
            }
            else
            {
                b[k]=a[j];
                j++;
                k++;
            }
        }
        while(i<=mid)
        {
            b[k]=a[i];
            i++;
            k++;
        }
    }
}
```

```

        k++;
    }
    while(j<=ub)
    {
        b[k]=a[j];

        j++;
        k++;
    }
    int s=0;
    for( k=lb; k<=ub; k++)
    {
        a[k]=b[s];

        s++;
    }
}

public static void mergesort(int a[], int lb,int ub)
{
    if(lb>=ub)
        return;

    int mid=(ub+lb)/2;
    mergesort(a,lb,mid);
    mergesort(a,mid+1, ub);
    merge(a,lb,mid,ub);

}

public static void printarrays(int a[], int n)
{
    for( int i=0; i<n; i++)

```

```

        {
            System.out.print(a[i]+" ");
        }

    }

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

        System.out.println("Enter the size of array");

        int n=sc.nextInt();

        System.out.println("Enter the array elements");

        int a[]=new int [n];

        for(int i=0; i<n; i++)
        {
            a[i]=sc.nextInt();
        }

        int lb=0;

        int ub=n-1;

        mergesort(a,lb,ub);

        printarrays(a,n);

    }
}

```

9) Write a Java program to implement constructor and constructor overloading

Write a java program of default constructor

//default constructor

class Rectangular

{

int l,b;

Rectangular()

```

{
    l=20;
    b=10;
}
void show()
{
    System.out.print("Area of rectangle is:"+l*b));
}
}
class Demo
{
    public static void main(String[] args)
    {

        Rectangular r=new Rectangular();
        r.show();
    }
}

```

Write a java program for parameterized constructor

//parameterized constructor

```

class Rectangular
{
    int length,breadth;
    Rectangular(int l, int b)
    {
        length=l;
        breadth=b;
    }
}

```

```

void show()
{
    System.out.print("Area of rectangle is:"+(length*breadth));
}
}
class Demo
{
    public static void main(String[] args)
    {

        Rectangular r=new Rectangular(10,20);
        r.show();
    }
}

```

Constructor overloading method

//constructor overloading method

```

import java.util.*;
class Box
{
    int lenght, breadth, height;
    Box()
    {
        lenght=20;
        breadth=10;
        height=10;
    }
    Box(int l, int b, int h)

```



```

{
    lenght=l;
    breadth=b;
    height=h;
}
void show()
{
    System.out.println("Volume of box is:" + (lenght*breadth*height));
}
}
class Demo
{
    public static void main(String[] args)
    {
        Box b=new Box();
        b.show();
        Box b1=new Box(10,10,5);
        b1.show();
    }
}

```

10) Write a Java program to implement method overloading

//overloading method

```
import java.util.*;
```

```
class Shape
```

```

{
    void area(int l, int b )
    {

```

```

        System.out.println("Area of rectangle is: "+(l*b));
    }
    void area(float r)
    {
        System.out.println("Area of circle is: "+ (Math.PI*r*r));
    }
    void area(int side)
    {
        System.out.println("Area of circle is: "+ (side*side));
    }
}
class Demo
{
    public static void main(String[] args)
    {

        Shape r=new Shape();
        r.area(20,10);
        r.area(7.0f);
        r.area(10);

    }
}

```

11) Write a Java program to sort given set of strings.

```

import java.util.Scanner;
public class Sort
{

```

```

public static void main(String args[])
{
    Scanner sc=new Scanner(System.in);
    int n=sc.nextInt();
    String a[]=new String[n];
    for(int i=0; i<n; i++)
    {
        a[i]=sc.next();
    }
    for(int i=0; i<n; i++)
    {
        for(int j=0; j<n-1-i; j++)
        {
            if(a[j].compareTo(a[j+1])>0)
            {
                String temp=a[j];
                a[j]=a[j+1];
                a[j+1]=temp;
            }
        }
    }
    for(int i=0; i<n; i++)
    {
        System.out.print(a[i]+" ");
    }
}

```

12) Write a Java program for using String Buffer to remove or delete a character

```

import java.util.*;

```

```

class Buffer
{
    public static void main(String args[])
    {
        StringBuffer s1=new StringBuffer("hello World");
        System.out.println(s1.delete(0,4));
        StringBuffer s2=new StringBuffer("hello World");
        System.out.println(s2.deleteCharAt(0));
    }
}

```

Inheritance

13) Write a Java program to implement Single Inheritance.

```

class Person
{
    String name;
    int age;
    void getp()
    {
        name="sanjit";
        age=21;
    }
}

class Student extends Person
{
    String pinno, clz;
    void gets()
    {
        pinno="5k1" ;
        clz="AEC";
    }
}

```

```

    }
    void show()
    {
        System.out.println("Name "+name);
        System.out.println("Age: "+age);
        System.out.println("pinno "+pinno);
        System.out.println("clz: "+clz);
    }
}
class Demo
{
    public static void main(String[] args)
    {
        Student s=new Student();
        s.gets();
        s.getp();
        s.show();
    }
}

```

14) Write a Java program to implement multi-level Inheritance

```

import java.util.*;

class Person
{
    String name;
    int age;
    void getp()
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter name and age");
    }
}

```

```
        name=sc.next();
        age=sc.nextInt();
    }
}
class Student extends Person
{
    String clz;
    String pinno;
    void gets()
    {
        getp();
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter college name and pinno");
        clz=sc.next();
        pinno=sc.next();
    }
}
```

```
class Marks extends Student
{
    int m1,m2,m3;
    void getm()
    {
        gets();
        m1=90;
        m2=89;
        m3=87;
    }
    void show()
```

```

{
    System.out.println("Name: "+ name);
    System.out.println("Age: "+ age);
    System.out.println("Pin_no: "+ pinno);
    System.out.println("College Name: "+ clz);
    System.out.println("Java: "+ m1);
    System.out.println("DBMS: "+ m2);
    System.out.println("DSA: "+ m3);

}
}
class Demo
{
    public static void main(String args[])
    {
        Marks m=new Marks();
        m.getm();
        m.show();
    }
}

```

15) Write a Java program to find the areas of different shapes using abstract classes.

Abstract class using method

```

abstract class Shapes
{
    abstract void area();
}
class Circle extends Shapes
{
    int r;
    void area()
    {
        r=6;
        System.out.println("Area of Circle is : "+(Math.PI*r*r));
    }
}

```

```

    }
}
class Rectangle extends Shapes
{
    int l,b;
    void area()
    {
        l=10;
        b=20;
        System.out.println("Area of Rectangle is : "+(l*b));
    }
}
class Square extends Shapes
{
    int s;
    void area()
    {
        s=5;
        System.out.println("Area of square is : "+(s*s));
    }
}
class Main
{
    public static void main(String args[])
    {
        Circle c=new Circle();
        c.area();
        Rectangle r= new Rectangle();
        r.area();
        Square s=new Square();
        s.area();
    }
}

```

Abstract class using constructor

```

abstract class Shapes
{
    abstract void area();
}
class Rectangel extends Shapes
{
    int length,breadth;
    Rectangel()

```



```
{
    length=10;
    breadth=5;
}
void area()
{
    System.out.println("Area of rectangle: "+length*breadth);
}
}
class Circle extends Shapes
{
    double r;
    Circle()
    {
        r=9.9;
    }
    void area()
    {
        System.out.println("Area of circle: "+ Math.PI*r*r);
    }
}
class Square extends Shapes
{
    int side;
    Square()
    {
        side=10;
    }
    void area()
```

```
{  
    System.out.println("Area of square: "+ side*side);  
}  
}
```

class Abstract

```
{  
    public static void main(String args[])  
    {  
        Rectangel r=new Rectangel();  
        r.area();  
        Circle c=new Circle();  
        c.area();  
        Square s=new Square();  
        s.area();  
    }  
}
```

16) Write a Java program for “super” keyword

class A

```
{  
    void show()  
    {  
        System.out.println("Class A method is invoked");  
    }  
}
```

class B extends A

```
{  
    void show()  
    {  
        super.show();  
    }  
}
```

```

        System.out.println("Class B method is invoked");
    }
}
class Main
{
    public static void main(String args[])
    {
        B b=new B();
        b.show();
    }
}

```

17) Take the details of internal exam marks in one Interface. Take the details of external exam marks in another interface. Write a Java program to find the total marks obtained in each subject by a student. (Note: Make use of Multiple Inheritance using interfaces.)

```

interface Internal
{
    public void intmarks();
}
interface External
{
    public void extmarks();
}
class Total implements Internal,External
{
    int i1,i2,i3,i4,i5,e1,e2,e3,e4,e5;
    public void intmarks()
    {
        i1=28;
        i2=30;
    }
}

```

```

        i3=27;

        i4=29;

        i5=25;
    }

    public void extmarks()
    {
        e1=65;

        e2=62;

        e3=59;

        e4=63;

        e5=69;
    }

    public void total()
    {
        System.out.println("Marks in java: "+ (i1+e1));
        System.out.println("Marks in DSA: "+ (i2+e2));
        System.out.println("Marks in CPP: "+ (i3+e3));
        System.out.println("Marks in Python: "+ (i4+e4));
        System.out.println("Marks in c: "+ (i5+e5));
    }
}

class Main
{
    public static void main(String args[])
    {
        Total t=new Total();

        t.intmarks();

        t.extmarks();

        t.total();
    }
}

```

```
}  
}
```

18) Write a JAVA program that implements Runtime polymorphism.(Dynamic Method Dispatching)

```
class A  
{  
    void show()  
    {  
        System.out.println("Class A methods are invoked");  
    }  
}  
class B extends A  
{  
    void show ()  
    {  
        System.out.println("Class B methods are invoked");  
    }  
}  
class C extends B  
{  
    void show ()  
    {  
        System.out.println("Class C methods are invoked");  
    }  
}  
class Main  
{  
    public static void main(String args[])
```

```
{  
    A a=new A();  
    B b=new B();  
    C c=new C();  
    A r;  
    r=a;  
    a.show();  
    r=b;  
    b.show();  
    r=c;  
    c.show();  
  
}  
}
```

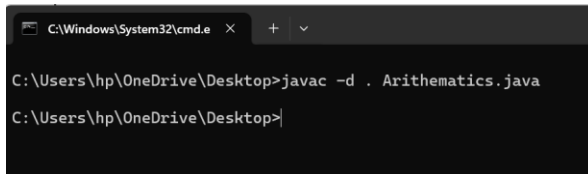
Packages

19) Write a Java program that import and use user defined package.

Step-1

```
package sanjit;  
  
public class Arithmetics  
{  
    public void show(int a, int b)  
    {  
        System.out.println("Addition of two number is: "+(a+b));  
        System.out.println("Subtraction of two number is: "+(a-b));  
        System.out.println("Multiplication of two number is: "+(a*b));  
        System.out.println("Division of two number is: "+(a/b));  
    }  
}
```

Step-2



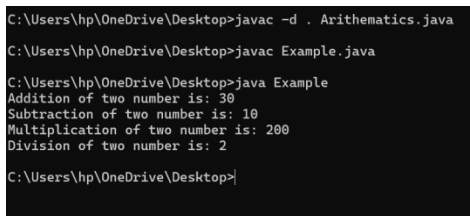
```
C:\Windows\System32\cmd.e  X  +  v
C:\Users\hp\OneDrive\Desktop>javac -d . Arithmetics.java
C:\Users\hp\OneDrive\Desktop>
```

Step-3

```
import sanjit.Arithmetics;

public class Example
{
    public static void main(String args[])
    {
        Arithmetics a=new Arithmetics();
        a.show(20,10);
    }
}
```

Step-4



```
C:\Users\hp\OneDrive\Desktop>javac -d . Arithmetics.java
C:\Users\hp\OneDrive\Desktop>javac Example.java
C:\Users\hp\OneDrive\Desktop>java Example
Addition of two number is: 30
Subtraction of two number is: 10
Multiplication of two number is: 200
Division of two number is: 2
C:\Users\hp\OneDrive\Desktop>
```

20) Write a Java program to illustrate the use of protected members in a package.

Case1:-

Step-1

```
package pack1;

public class A
{
    protected void show()
```

```
{
    System.out.println("procted member is invoked");
}
}
```

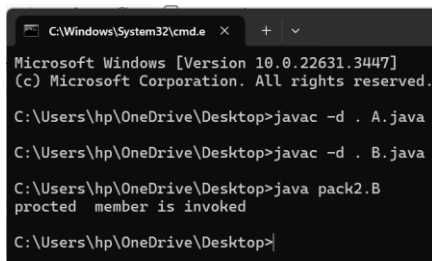
Step-2

```
package pack2;

import pack1.A;

public class B extends A
{
    public static void main(String args[])
    {
        B b= new B();
        b.show();
    }
}
```

Step-3



```
C:\Windows\System32\cmd.e
Microsoft Windows [Version 10.0.22631.3447]
(c) Microsoft Corporation. All rights reserved.

C:\Users\hp\OneDrive\Desktop>javac -d . A.java
C:\Users\hp\OneDrive\Desktop>javac -d . B.java
C:\Users\hp\OneDrive\Desktop>java pack2.B
procted member is invoked
C:\Users\hp\OneDrive\Desktop>
```

Case2

Step-1

```
package pack1;

public class A
{
    protected void show()
```



```

{
    System.out.println("protected member is invoked");
}
}

```

Step-2

```

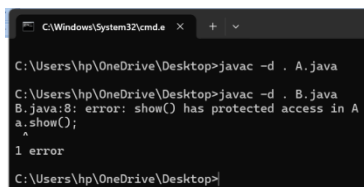
package pack2;

import pack1.A;

public class B extends A
{
    public static void main(String args[])
    {
        A a= new A();
        a.show();
    }
}

```

Step-3



```

C:\Windows\System32\cmd.exe
C:\Users\hp\OneDrive\Desktop>javac -d . A.java
C:\Users\hp\OneDrive\Desktop>javac -d . B.java
B.java:8: error: show() has protected access in A
    a.show();
      ^
1 error
C:\Users\hp\OneDrive\Desktop>

```

21) Write a Java program to illustrate exception handling mechanism using multiple catch clauses.

```

class MultipleCatch
{
    public static void main(String args[])
    {
        try
        {
            int a=Integer.parseInt(args[0]);

```

```

        int b=Integer.parseInt(args[1]);
        System.out.println(a/b);
    }
    catch(ArithmeticException e)
    {
        System.out.println(e);
    }
    catch(NumberFormatException e)
    {
        System.out.println(e);
    }
}

```

22) Write a Java program to make use of Built-in and user-defined Exceptions in handling a run time exception

```

import java.util.Scanner;

class InvalidAgeException extends Exception
{
    InvalidAgeException(String msg)
    {
        super(msg);
    }
}

class User
{
    public static void main(String args[])
    {
        try
        {

```

```

Scanner sc=new Scanner(System.in);
System.out.println("Enter your age");
int age=sc.nextInt();
if(age<18)
{
    throw new InvalidAgeException("age should be greater than 18");
}
else
{
    System.out.println("YOu are eligible to vote");
}
}
catch(InvalidAgeException e)
{
    System.out.println(e);
}
}
}

```

Multithreading

23) Write a Java program that creates threads by extending Thread class .First thread display “Good Morning “every 1 sec, the second thread displays “Hello “every 2 seconds and the third display “Welcome” every 3 seconds, (Repeat the same by implementing Runnable).

```

import java.util.*;
class A extends Thread
{
    public void run()
    {
        try
        {
            for(int i=1; i<=5; i++)

```

```

    {
        Thread.sleep(1000);
        System.out.println("Good Morning");
    }
}
catch(Exception e)
{
    System.out.println(e);
}
}
}
class B extends Thread
{
    public void run()
    {
        try
        {
            for(int i=1; i<=5; i++)
            {
                Thread.sleep(2000);
                System.out.println("Hello ");
            }
        }
        catch(Exception e)
        {
            System.out.println(e);
        }
    }
}
class C extends Thread

```

```
{
    public void run()
    {
        try{
            for(int i=1; i<=5; i++)
            {
                Thread.sleep(3000);
                System.out.println("Welcome");
            }
        }
        catch(Exception e)
        {
            System.out.println(e);
        }
    }
}

class Multithread
{
    public static void main(String args[])
    {
        A a=new A();
        B b=new B();
        C c =new C();
        a.start();
        b.start();
        c.start();
    }
}
```

24) Write a JDBC program to perform the following operations by connecting to MYSQL database.

- I) Inserting Data into Table**
- II) Updating Data in the Table.**
- III) Deleting Data From the Table based on a column value.**

1: Program

```
import java.sql.*;
import java.util.Scanner;

import javax.swing.JOptionPane;
class Insertions {
    public static void main(String[] args) {
        try{
            String url = "jdbc:mysql://localhost:3306/newdatabase";
            String user = "root";
            String pass = "Sanjit@123";
            Connection con = DriverManager.getConnection(url,user, pass);
            if(con != null) System.out.println("Connection Successful");
            Statement st = con.createStatement();
            Scanner sc = new Scanner(System.in);
            System.out.println("How many insertions : ");
            int n = sc.nextInt();
            for (int i = 1; i <= n; i++ )
            {
                String roll = JOptionPane.showInputDialog("Enter rollno of Student" + i + " : ");
                String name = JOptionPane.showInputDialog("Enter name of student" + i + " : ");
                String age = JOptionPane.showInputDialog("Enter age of student" + i + " : ");
                String query = "insert into Studetails values (" + roll + ", '" + name + "'," + age +
                "));";
                st.executeUpdate(query);
            }

        }catch (SQLException e){
            System.out.println(e);
        }
    }
}
```

25) JDBC program to perform Updating Data into Table.

Program:

```
import java.sql.*;
import java.util.Scanner;

import javax.swing.JOptionPane;
class Updations {
    public static void main(String[] args) {
        try{
            String url = "jdbc:mysql://localhost:3306/newdatabase";
            String user = "root";
            String pass = "sanjit@123";
            Connection con = DriverManager.getConnection(url,user, pass);
            if(con != null) System.out.println("Connection Successful");
            Statement st = con.createStatement();
            Scanner sc = new Scanner(System.in);
            String query = "Update Studetails SET name ='Sanjit' where rollno = 123 ";
            st.executeUpdate(query);
        }
        catch (SQLException e){
            System.out.println(e);
        }
    }
}
```

26) JDBC program to perform Deleting Data from A Table

Program:

```
import java.sql.*;
import java.util.Scanner;

import javax.swing.JOptionPane;
class Deletions {
    public static void main(String[] args) {
        try{
            String url = "jdbc:mysql://localhost:3306/newdatabase";
            String user = "root";
            String pass = "Sanjit@123";
            Connection con = DriverManager.getConnection(url,user, pass);
            if(con != null) System.out.println("Connection Successful");
            Statement st = con.createStatement();
            Scanner sc = new Scanner(System.in);
            String query = " Delete from Studetails where age = 18 ";
            st.executeUpdate(query);
        }
    }
}
```

```
    }  
    catch (SQLException e){  
        System.out.println(e);  
    }  
}  
}
```