Exp no : 1.a	**Sort in Ascending Order**
Date :	

Aim :

Algorithm:

Source Code :

```
import java.util.Scanner;
public class AscendingOrder
{
   public static void main(String[] args)
   {
      int n, temp;
      Scanner s = new Scanner(System.in);
      System.out.print("Enter no. of elements you want in array:");
      n = s.nextInt();
      int a[] = new int[n];
      System.out.println("Enter all the elements:");
      for (int i = 0; i < n; i++)
      {
         a[i] = s.nextInt();
      }
      for (int i = 0; i < n; i++)
      {
         for (int j = i + 1; j < n; j++)
         {
            if (a[i] > a[j])
            {
               temp = a[i];
               a[i] = a[j];
               a[j] = temp;
            }
         }
      }
      System.out.print("Ascending Order:");
      for (int i = 0; i < n - 1; i++)
      {
         System.out.print(a[i] + ",");
      }
      System.out.print(a[n - 1]);
   }
}
```

Output :

```
Enter no. of elements you want in array: 10
Enter all the elements:
2 5 3 4 8 6 0 1 7 9
Ascending Order:0,1,2,3,4,5,6,7,8,9
```

Result :

Exp no : 1.b	**Binary Search**
Date :	

Aim :

Algorithm:

Source Code :

```
import java.util.Scanner;

class BinarySearch
{
 public static void main(String ar[])
 {  int i,mid,first,last,x,n,flag=0;

   Scanner sc=new Scanner(System.in);
   System.out.println("Enter number of elements:");
   n=sc.nextInt();

   int a[]=new int[n];

   System.out.println("Enter elements of array:");
   for(i=0;i<n;++i)
    a[i]=sc.nextInt();

   System.out.println("Enter element to search:");
   x=sc.nextInt();

   first=0;
   last=n-1;

   while(first<=last)
   {
    mid=(first+last)/2;

    if(a[mid]>x)
     last=mid-1;
    else
     if(a[mid]<x)
      first=mid+1;
     else
     {
      flag=1;
      System.out.println("element found");
      break;
     }
   }
 if(flag==0)
  System.out.println("element not found");
 }
}
```

Output :

```
Enter number of elements:
5
Enter elements of array:
2 1 4 5 99
Enter element to search:
4
element found
```

Result :

Exp no : 1.c	**Finding Largest and Smallest Number in an Array**
Date :	

Aim :

Algorithm:

Source Code :

```
public class LargestSmallest
{
    public static void main(String[] args)
    {
        int a[] = new int[] { 23, 34, 13, 64, 72, 90, 10, 15, 9, 27 };

        int sum = 0;
        int min = a[0];
        int max = a[0];

        for (int i = 1; i < a.length; i++)
        {
            if (a[i] > max)
            {
                max = a[i];
            }
            if (a[i] < min)
            {
                min = a[i];
            }
            sum = sum + a[i];
        }
        System.out.println("The sum is : " + sum);
        System.out.println("Largest Number in a given array is : " + max);
        System.out.println("Smallest Number in a given array is : " + min);
    }

}
```

Output :

```
The sum is : 334
Largest Number in a given array is : 90
Smallest Number in a given array is : 9
```

Result :

Exp no : 1.d	**To print marks above 60**
Date :	

Aim :

Algorithm:

Source Code :

```
import java.util.Scanner;

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

        int marks[] = new int[6];
        int i;
        String name[] = new String[30];
        Scanner scanner = new Scanner(System.in);


for(i=0; i<6; i++) {
       System.out.print("Enter Name of Student and Marks of Subject"+(i+1)+":");
           name[i]  = scanner.next();
           marks[i] = scanner.nextInt();

        }
       for(i=0; i<6;i++) {
       if(marks[i]>=60)
        {
           System.out.println(name[i] + "    " + marks[i]);
        }
       }
    }
}
```

Output :

```
Enter Name of Student and Marks of Subject1:Harish 90
Enter Name of Student and Marks of Subject2:Natesh 87
Enter Name of Student and Marks of Subject3:laksh 20
Enter Name of Student and Marks of Subject4:Sarvesh 50
Enter Name of Student and Marks of Subject5:Deep 82
Enter Name of Student and Marks of Subject6:parth 1

Harish   90
Natesh   87
Deep   82
```

Result :

Exp no : 2.a	**Arithmetic Operations using Switch Case**
Date :	

Aim :

Algorithm:

Source Code :

```
import java.util.Scanner;
public class ArithmeticOperators
{
  public static void main(String args[])
  {
      Scanner s = new Scanner(System.in);
      while(true)
      {
         System.out.println("");
         System.out.println("Enter the two numbers to perform operations ");
         System.out.print("Enter the first number : ");
         int x = s.nextInt();
         System.out.print("Enter the second number : ");
         int y = s.nextInt();
         System.out.println("Choose the operation you want to perform ");
         System.out.println("Choose 1 for ADDITION");
         System.out.println("Choose 2 for SUBTRACTION");
         System.out.println("Choose 3 for MULTIPLICATION");
         System.out.println("Choose 4 for DIVISION");
         System.out.println("Choose 5 for MODULUS");
         System.out.println("Choose 6 for EXIT");
         int n = s.nextInt();
         switch(n)
         {
            case 1:
            int add;
            add = x + y;
            System.out.println("Result : "+add);
            break;

            case 2:
            int sub;
            sub = x - y;
            System.out.println("Result : "+sub);
            break;

            case 3:
            int mul;
            mul = x * y;
            System.out.println("Result : "+mul);
            break;
            case 4:
            float div;
            div = (float) x / y;
            System.out.print("Result : "+div);
            break;

            case 5:
            int mod;
            mod = x % y;
            System.out.println("Result : "+mod);
            break;

            case 6:
            System.exit(0);
          }
       }
    }
```

Output :

```
Enter the two numbers to perform operations
Enter the first number : 78
Enter the second number : 133
Choose the operation you want to perform
Choose 1 for ADDITION
Choose 2 for SUBTRACTION
Choose 3 for MULTIPLICATION
Choose 4 for DIVISION
Choose 5 for MODULUS
Choose 6 for EXIT
1
Result : 211
```

Result :

Exp no : 2.b	**Even or Odd using Switch Case**
Date :	

Aim :

Algorithm:

Source Code :

```
import java.util.*;

class EvenOddSwitch
{
	public static void main(String args[])
	{
		int n,i;
		Scanner s = new Scanner(System.in);
		n = s.nextInt();
		switch(n%2)
		{
			case 0 :
				System.out.println("This number is even");
				break;
			case 1 :
				System.out.println("This number is odd");
				break;
		}
	}
}
```

Output :

```
5
This number is odd
```

Result :

Exp no : 2.c	**Largest of three numbers**
Date :	

Aim :

Algorithm:

Source Code :

```
import java.util.Scanner;

class LargestOfThreeNumbers
{
  public static void main(String args[])
  {
    int x, y, z;
    System.out.println("Enter three integers");
    Scanner in = new Scanner(System.in);

    x = in.nextInt();
    y = in.nextInt();
    z = in.nextInt();

    if (x > y && x > z)
      System.out.println("First number is largest.");
    else if (y > x && y > z)
      System.out.println("Second number is largest.");
    else if (z > x && z > y)
      System.out.println("Third number is largest.");
    else
      System.out.println("The numbers are not distinct.");
  }
}
```

Output :

```
Enter three integers
45
0
100
Third number is largest.
```

Result :

Exp no : 2.d	**Leap Year**
Date :	

Aim :

Algorithm:

Source Code :

```
import java.util.Scanner;
public class LeapYear
{
   public static void main(String args[])
   {
      Scanner s = new Scanner(System.in);
      System.out.print("Enter any year:");
      int year = s.nextInt();
      boolean flag = false;
      if(year % 400 == 0)
      {
         flag = true;
      }
      else if (year % 100 == 0)
      {
         flag = false;
      }
      else if(year % 4 == 0)
      {
         flag = true;
      }
      else
      {
         flag = false;
      }
      if(flag)
      {
         System.out.println("Year "+year+" is a Leap Year");
      }
      else
      {
         System.out.println("Year "+year+" is not a Leap Year");
      }
   }
}
```

Output :

```
Enter any Year : 2012

Year is 2012 Leap Year
```

Result :

Exp no : 3.a	**Armstrong number**
Date :	

Aim :

Algorithm:

Source Code :

```
import java.util.Scanner;

public class Armstrong
{
   public static void main(String args[])
   {
      int n, nu, num=0, rem;
      Scanner scan = new Scanner(System.in);

      System.out.print("Enter any Positive Number : ");
      n = scan.nextInt();

      nu = n;

      while(nu != 0)
      {
         rem = nu%10;
         num = num + rem*rem*rem;
         nu = nu/10;
      }
      if(num == n)
      {
         System.out.print("Armstrong Number");
      }
      else
      {
         System.out.print("Not an Armstrong Number");
      }
   }
}
```

Output :

```
Enter any Positive Number : 153
Armstrong Number
```

Result :

Exp no : 3.b	**Fibonacci Series**
Date :	

Aim :

Algorithm:

Source Code :

```
import java.util.Scanner;

public class FibonacciSeries {

    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the value of n: ");
        int n = s.nextInt();
        fibonacci(n);
    }

    public static void fibonacci(int n) {
        if (n == 0) {
            System.out.println("0");
        } else if (n == 1) {
            System.out.println("0 1");
        } else {
            System.out.print("0 1 ");
            int a = 0;
            int b = 1;
            for (int i = 1; i < n; i++) {
                int nextNumber = a + b;
                System.out.print(nextNumber + " ");
                a = b;
                b = nextNumber;
            }
        }
    }
}
```

Output :

```
Enter the value of n: 5
0 1 1 2 3 5
```

Result :

Exp no : 3.c	**Multiplication Table**
Date :	

Aim :

Algorithm:

Source Code :

```
import java.util.Scanner;
public class MulTable {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      int no = 0;
      System.out.println("Enter the number : ");
      no = sc.nextInt();
      for (int i = 0; i < 10; i++) {
         System.out.println(no + " * " + (i + 1) + " = " + (no * (i + 1)));
      }
   }
}
```

Output :

```
Enter the number :
5
5 * 1 = 5
5 * 2 = 10
5 * 3 = 15
5 * 4 = 20
5 * 5 = 25
5 * 6 = 30
5 * 7 = 35
5 * 8 = 40
5 * 9 = 45
5 * 10 = 50
```

Result :

Exp no : 3.d	**Perfect Number**
Date :	

Aim :

Algorithm:

Source Code :

```
import java.util.Scanner;
public class Perfect
{
    public static void main(String[] args)
    {
        int n, sum = 0;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter any integer you want to check:");
        n = s.nextInt();
        for(int i = 1; i < n; i++)
        {
            if(n % i == 0)
            {
                sum = sum + i;
            }
        }
        if(sum == n)
        {
            System.out.println("Given number is Perfect");
        }
        else
        {
            System.out.println("Given number is not Perfect");
        }
    }
    int divisor(int x)
    {
       return x;
    }
}
```

Output :

```
Enter any integer you want to check:6
Given number is Perfect
```

Result :

Exp no : 3.e	Reverse of a Number
Date :	

Aim :

Algorithm:

Source Code :

```
import java.util.Scanner;

class ReverseNumber
{
  public static void main(String args[])
  {
    int n, reverse = 0;

    System.out.println("Enter an integer to reverse");
    Scanner in = new Scanner(System.in);
    n = in.nextInt();

    while(n != 0)
    {
       reverse = reverse * 10;
       reverse = reverse + n%10;
       n = n/10;
    }

    System.out.println("Reverse of the number is " + reverse);
  }
}
```

Output :

```
Enter an integer to reverse
12345
Reverse of the number is 54321
```

Result :

Exp no : 4.a	**Sub string of a String**
Date :	

Aim :

Algorithm:

Source Code :

```
import java.util.Scanner;

class SubstringsOfAString
{
  public static void main(String args[])
  {
    String string, sub;
    int i, n = 0, c, length;

    Scanner in = new Scanner(System.in);
    System.out.println("Enter a string to print it's all substrings");
    string  = in.nextLine();

    length = string.length();

    System.out.println("Substrings of \""+string+"\" are:");

    for (c = 0; c < length; c++)
    {
      for(i = 1; i <= length - c; i++)
      {
        sub = string.substring(c, c+i);
        System.out.println(sub);
      }
     n = n + i;
    }
    System.out.println("No of substrings present are: "+n);
  }
}
```

Output :

```
java
Substrings of "java" are:
j
ja
jav
java
a
av
ava
v
va
a
No of substrings present are: 14
```

Result :

Exp no : 4.b	**Replacing of Substring**
Date :	

Aim :

Algorithm:

Source Code :

```
import java.util.*;
public class Replace
{
      public static void main(String args[])
      {
            String a, e;
            Scanner sc = new Scanner(System.in);
            String s1=sc.nextLine();
            System.out.println("Enter the variable to be replace and variable that is
          changed");
            a = sc.next();
            e = sc.next();
            String replaceString=s1.replace(a,e);
            System.out.println(replaceString);
      }
}
```

Output :

```
jtvt
Enter the variable to be replace and variable that is changed
t a
java
```

Result :

Exp no : 5.a	**Arithmetic Operations using Packages**
Date :	

Aim :

Algorithm:

Source Code :

```
package  add;

public class Add
{
	int res;
	public void addop(int a, int b)
	{

		res = a + b;
		System.out.println("Add :"+res);
	}
}

package sub;

public class Sub
{
	int res;
	public void subop(int a,int b)
	{
		res = a - b;
		System.out.println("Sub:"+	res);
	}
}

package mul;

public class Mul
{
	int res;
	public void mulop(int a,int b)
	{

		res = a * b;
		System.out.println("Mul :"+res);
	}
}
```

```
package div;

public class Div
{
    int res;
    public void divop(int a,int b)
    {

        res = a / b;
        System.out.println("Div :"+res);
    }
}

import java.util.*;
import add.*;
import sub.*;
import mul.*;
import div.*;


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

        Add ad = new Add();
        Sub su = new Sub();
        Mul mu = new Mul();
        Div di = new Div();

        ad.addop(20,10);
        su.subop(20,10);
        mu.mulop(20,10);
        di.divop(20,10);
    }
}
```

Output :

```
Add :30
Sub:10
Mul :200
Div :2
```

Result :

Exp no : 5.b	**Find perimeter and area of different shapes using Packages**
Date :	

Aim :

Algorithm:

Source Code :

```
package Shape;
public class Circle
{
private int radius;
public Circle(int r)
{
radius=r;
}
public double perimeter()
{
return (2*3.14*radius);
}
public double area()
{
return (3.14*radius*radius);
}
}

package Shape;
public class Square
{
private int side;
public Square(int s)
{
side=s;
}
public int perimeter()
{
return (4*side);
}
public int area()
{
return (side*side);
}
}

package Shape;
public class Triangle
{
private int side1,side2,side3;
public Triangle(int s1,int s2,int s3)
```

```
{
side1=s1;
side2=s2;
side3=s3;
}
public int perimeter()
{
return side1+side2+side3;
}
public double area()
{
double s=(side1+side2+side3)/2;
double a=Math.sqrt((s-side1)+(s-side2)+(s-side3));
return a;
}
}

import Shape.*;
import java.util.*;

class Calculate
{
public static void main(String []args)
{
Scanner sc=new Scanner(System.in);
System.out.println("Enter The side of the Square : ");
int s=sc.nextInt();
Square sq=new Square(s);
System.out.println("Perimeter of Square is " + sq.perimeter());
System.out.println("Area of Square is " + sq.area());

System.out.println("Enter The radius of the Circle : ");
int r=sc.nextInt();
Circle ci=new Circle(s);
System.out.println("Perimeter of Circle is " + ci.perimeter());
System.out.println("Area of Circle is " + ci.area());

System.out.println("Enter The Side1 of the Triangle : ");
int s1=sc.nextInt();
System.out.println("Enter The Side2 of the Triangle : ");
int s2=sc.nextInt();
System.out.println("Enter The Side3 of the Triangle : ");
int s3=sc.nextInt();
Triangle t=new Triangle(s1,s2,s3);
System.out.println("Perimeter of Triangle is " + t.perimeter());
System.out.println("Area of Triangle is " + t.area());
}
}
```

Output :

```
Enter The side of the Square :
4
Perimeter of Square is 16
Area of Square is 16
Enter The radius of the Circle :
5
Perimeter of Circle is 25.12
Area of Circle is 50.24
Enter The Side1 of the Triangle :
3
Enter The Side2 of the Triangle :
4
Enter The Side3 of the Triangle :
5
Perimeter of Triangle is 12
Area of Triangle is 2.449489742783178
```

Result :

Exp no : 6.a	Students detail using interface
Date :	

Aim :

Algorithm:

Source Code :

```
import java.io.*;

interface Exam {

 void percent_cal();
}

class Student {
 String name;
 int roll_no,mark1,mark2;

 Student(String n, int r, int m1, int m2) {

  name=n;
  roll_no=r;
  mark1=m1;
  mark2=m2;
 }

 void display() {

  System.out.println ("Name of Student: "+name);
  System.out.println ("Roll No. of Student: "+roll_no);
  System.out.println ("Marks of Subject 1: "+mark1);
  System.out.println ("Marks of Subject 2: "+mark2);
 }
}

class Result extends Student implements Exam {

 Result(String n, int r, int m1, int m2) {

  super(n,r,m1,m2);
 }

 public void percent_cal() {

  int total=(mark1+mark2);
  float percent=total*100/200;
  System.out.println ("Percentage: "+percent+"%");
 }
 void display() {

  super.display();
 }
}

public class StuDetail {

 public static void main(String[] args) {



  Result R = new Result("Ragini",12,93,84);
  R.display();
  R.percent_cal();
 }
```

Output :

```
Name of Student: Ragini
Roll No. of Student: 12
Marks of Subject 1: 93
Marks of Subject 2: 84
Percentage: 88.0%
```

Result :

Exp no : 6.b	**Volume of Cylinder and Sphere using interface**
Date :	

Aim :

Algorithm:

Source Code :

```
import java.io.*;
import java.util.*;
interface Shape
{
 final float pi=3.14f;
 float area(int r);
}
interface CylinderShape
{
 final float pie=3.14f;
 float volume(int r , int h);
}
class Cylinder implements CylinderShape
{
 public float volume(int r, int h)
 {
  return(pie*r*r*h);
 }
}
class Sphere implements Shape
{
 public float area(int r)
 {
  return((4/3)*pi*r*r);
 }
}
class ShapeDemo
{
 public static void main(String arg[])
 {
  int r, h;
  Scanner sc = new Scanner(System.in);
   System.out.print("Enter radius & height-");
   r=sc.nextInt();
   h=sc.nextInt();
   Cylinder c1=new Cylinder();
   Sphere s1=new Sphere();
   System.out.println("Volume of Cylinder:"+c1.volume(r, h));
   System.out.println("Area of Sphere:"+s1.area(r));
 }
}
```

Output :

```
Enter radius & height-4 5
Volume of Cylinder:251.20001
Area of Sphere:50.24
```

Result :

Exp no : 7.a	**Account detail using inheritance**
Date :	

Aim :

Algorithm:

Source Code :

```
import java.io.*;

class Account
{
    int accountNumber, balance;
    Account(int accountNumber, int balance)
    {
        this.accountNumber = accountNumber;
        this.balance = balance;
    }
    void credit(int amount)
    {
        balance = balance + amount;
    }
    void debit(int amount)
    {
        balance = balance - amount;
    }
    void displayBalance()
    {
        System.out.println("Balance = "+balance);
    }
}

class SBAccount extends Account
{
    double rate;
    SBAccount(int accountNumber, int balance, double rate)
    {
        super(accountNumber, balance);
        this.rate = rate;
    }
    void displayInterestRate()
    {
        System.out.println("Interest Rate = " +rate);
    }
}

class AccountDemo
{
    public static void main(String args[])
    {
        SBAccount obj1;
        obj1 = new SBAccount(1001, 5000, 6.5);
        obj1.credit(2000);
        obj1.debit(1000);
        obj1.displayBalance();
        obj1.displayInterestRate();
    }
}
```

Output :

```
Balance = 6000
Interest Rate = 6
```

Result :

Exp no : 7.b	**College Details using inheritance**
Date :	

Aim :

Algorithm:

Source Code :

```
import java.io.*;
class College
{
String collegeName,principalName;
College(String collegeName,String principalName)
{
this.collegeName = collegeName;
this.principalName = principalName;
}
void  displayCollegeDetails()
{
System.out.println("CollegeName:"+collegeName);
System.out.println("PrincipalName:"+principalName);
}
}
class Department extends College
{
String departmentName,HODName;
Department(String collegeName,String principalName,String departmentName,String HODName)
{
super(collegeName,principalName);
this.departmentName=departmentName;
this.HODName=HODName;
}
void displayDepartmentDetails()
{
System.out.println("DepartmentName:"+departmentName);
System.out.println("HODName:"+HODName);
}
}
class SingleInherit
{
public static void main(String[]args)
{
Department obj;
obj=new Department("NIT","B.K.Sinha","Electronics","B.C.Rai");
obj.displayCollegeDetails();
obj.displayDepartmentDetails();
}
}
```

Output :

```
CollegeName:NIT
PrincipalName:B.K.Sinha
DepartmentName:Electronics
HODName:B.C.Rai
```

Result :

Exp no : 8.a	**Creating simple Thread**
Date :	

Aim :

Algorithm:

Source Code :

```
import java.io.*;
class A extends Thread
{
public void run()
{
for(int i=1;i<=5;i++)
{
if(i==1)
yield();
System.out.println("from thread A i="+i);
}
System.out.println("exit from A");
}
}
class B extends Thread
{
public void run()
{
for(int j=1;j<=5;j++)
{
System.out.println("from thread B j="+j);
if(j==3)
System.out.println("exit from 5");
stop();
}
}
}
class C extends Thread
{
public void run()
{
for(int k=1;k<=5;k++)
{
System.out.println("thread c ="+k);
if(k==1)
try
{
sleep(1500);
}
catch(Exception c)
{
```

```
System.out.println("exit from c");
}
}
}
}
class Threadtest
{
public static void main(String[]args)
{
A a = new A();
B b = new B();
C c = new C();
System.out.println("Start thread A");
a.start();
b.start();
c.start();
System.out.println("exit from main thread");
}
}
```

Output :

```
Start thread A
exit from main thread
Thread A started
Thread B started
from thread A i=1
thread C started
from thread A i=2
from thread B j=1
from thread A i=3
thread c =1
from thread A i=4
from thread B j=2
exit from A
thread c =2
from thread B j=3
thread c =3
thread c =4
from thread B j=4
exit from c
exit from B
```

Result :

Exp no : 8.b	**Thread Priority**
Date :	

Aim :

Algorithm:

Source Code :

```
import java.io.*;
class A extends Thread
{
public void run()
{
System.out.println("Thread A started");
for(int i=1;i<=4;i++)
{
System.out.println("from thread A i="+i);
}
System.out.println("exit from A");
}
}
class B extends Thread
{
public void run()
{
System.out.println("Thread B started");
for(int j=1;j<=4;j++)
{
System.out.println("from thread B j="+j);
}
System.out.println("exit from B");
}
}
class C extends Thread
{
public void run()
{
System.out.println("thread C started");
for(int k=1;k<=4;k++)
{
System.out.println("thread c ="+k);
}
System.out.println("exit from c");
}
}
class ThreadPriority
{
public static void main(String[]args)
{
A threadA = new A();
B threadB = new B();
C threadC = new C();
threadC.setPriority(Thread.MAX_PRIORITY);
threadB.setPriority(threadA.getPriority()+1);
threadA.setPriority(Thread.MIN_PRIORITY);
System.out.println("start thread A");
threadA.start();
System.out.println("start thread B");
threadB.start();
System.out.println("start thread C");
threadC.start();
System.out.println("end of main thread");
}
}
```

Output :

```
start thread A
start thread B
Thread A started
start thread C
from thread A i=1
Thread B started
thread C started
from thread A i=2
end of main thread
from thread A i=3
thread c =1
from thread B j=1
thread c =2
thread c =3
thread c =4
from thread A i=4
exit from c
from thread B j=2
exit from A
from thread B j=3
from thread B j=4
exit from B
```

Result :

Exp no : 9.a	Pre-defined Exception Handling
Date :	

Aim :

Algorithm:

Source Code :

```
import java.io.*;
import java.util.*;
class Predefined
{
public static void main(String args[])
{
int num1,num2;
try
{
num1=0;
num2=62/num1;
System.out.println("try block msg");
}catch(ArithmeticException e)
{
System.out.println("error divided by zero");
}
System.out.println("out of catch block");
}
}
```

Output :

```
error divided by zero
out of catch block
```

Result :

Exp no : 9.b	**User-defined Exception Handling**
Date :	

Aim :

Algorithm:

Source Code :

```
import java.io.*;
import java.util.*;
class MyException extends Exception
{
String str1;
MyException(String str2)
{
str1=str2;
}
public String toString()
{
return("output String="+str1);
}
}
class Userdefined
{
public static void main(String[]args)
{
try
{
throw new MyException("custom");
}
catch(MyException exp)
{
System.out.println("this is catch block");
System.out.println("exp");
}
}
}
```

Output :

```
this is catch block
exp
```

Result :

Exp no : 10.a	**File Reader**
Date :	

Aim :

Algorithm:

Source Code :

```
import java.io.*;
class Filereader
{
public static void main(String[]args)
{
try
{
FileReader fr=new FileReader("sample2.txt");
int i;
while((i=fr.read())!=-1)
{
System.out.println((char)i);
}
fr.close();
}
catch(Exception e)
{
System.out.println("Exception:"+e);
}
}
}
```

Output :

```
A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
```

S
T
U
V
W
X
Y
Z

Result :

Exp no : 10.b	File Writer
Date :	

Aim :

Algorithm:

Source Code :

```
import java.io.*;
class Filewriter
{
public static void main(String[]args)
{
try
{
FileWriter fw= new FileWriter("sample2.txt");
for(char i=65;i<91;i++)
{
fw.write(i);
}
fw.close();
}
catch(Exception e)
{
System.out.println("Exception :"+e);
}
}
}
```

Output :

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
```

Result :

Exp no : 11.a	**Awaking a Human Face using Applets**
Date :	

Aim :

Algorithm:

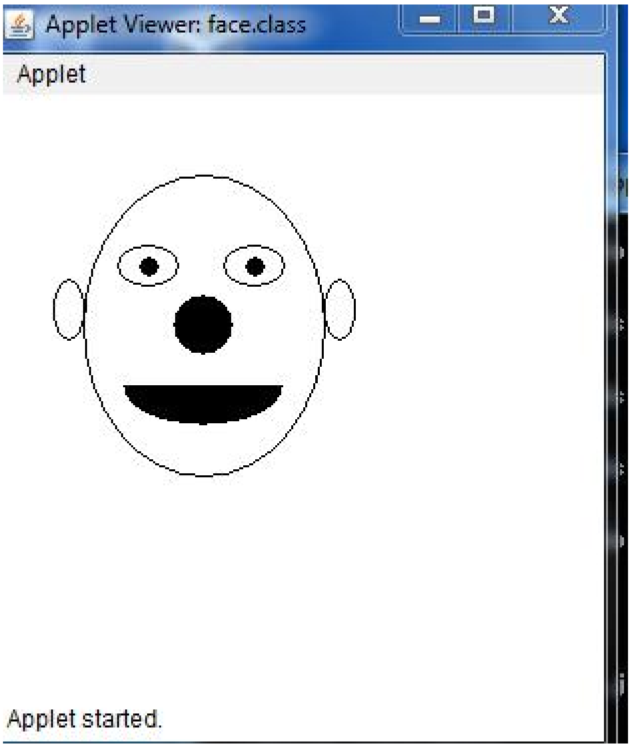
Source Code :

```
import java.awt.*;
import java.io.*;
import java.applet.*;

/*<applet code="face.class" height=300 width=300>
</applet>*/


public class face extends Applet
{
public void paint(Graphics g)
{
g.drawOval(40,40,120,150);
g.drawOval(57,75,30,20);
g.drawOval(110,75,30,20);
g.fillOval(68,81,10,10);
g.fillOval(121,81,10,10);
g.fillOval(85,100,30,30);
g.fillArc(60,125,80,40,180,180);
g.drawOval(25,92,15,30);
g.drawOval(160,92,15,30);
}
}
```

Output :



Applet started.

Result :

Exp no : 11.b	Drawing Line and Rectangle using Applets
Date :	

Aim :

Algorithm:

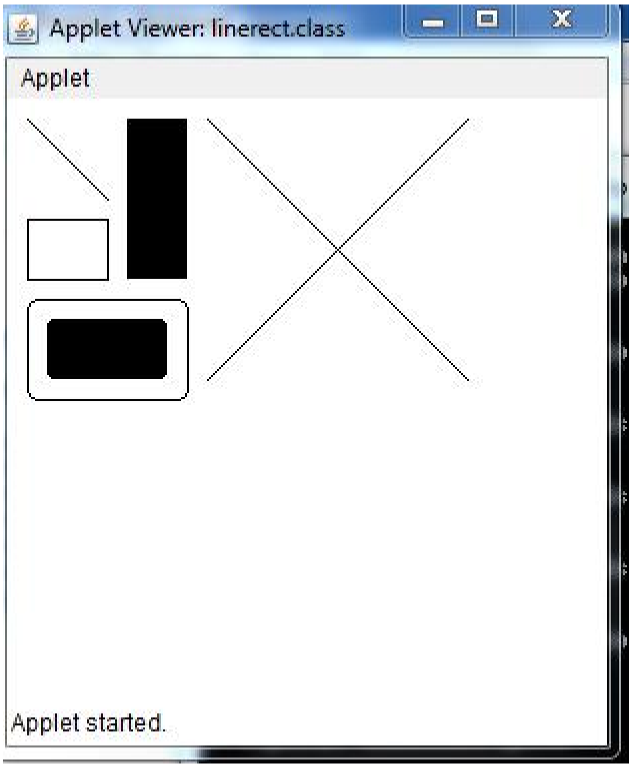
Source Code :

```
    import java.awt.*;
    import java.io.*;
    import java.applet.*;

    /*<applet code="linerect.class" height=300 width=300>
    </applet>*/
    public class linerect extends Applet
    {
    public void paint(Graphics g)
    {
    g.drawLine(10,10,50,50);
    g.drawRect(10,60,40,30);
    g.fillRect(60,10,30,80);
    g.drawRoundRect(10,100,80,50,10,10);
    g.fillRoundRect(20,110,60,30,5,5);
    g.drawLine(100,10,230,140);
    g.drawLine(100,140,230,10);

}
    }
```

Output :



Result :

Exp no : 12.a	**Grid Layout**
Date :	

Aim :

Algorithm:

Source Code :

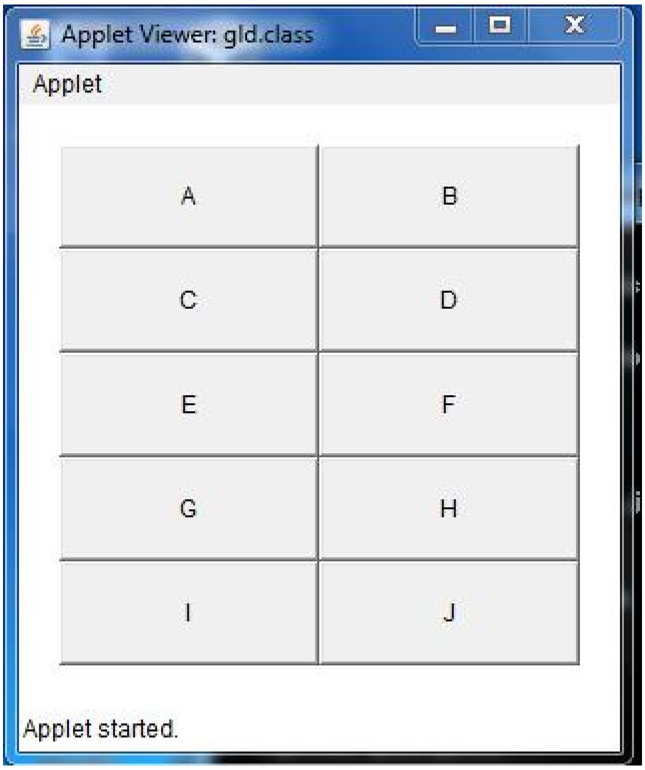
```
import java.awt.*;
import java.io.*;
import java.applet.Applet.*;
import java.applet.*;

/*<applet code="gld.class" height=300 width=300>
</applet>*/



public class gld extends Applet
{
public void init()
{
setLayout(new GridLayout(5,2));
Button b1=new Button("A");
Button b2=new Button("B");
Button b3=new Button("C");
Button b4=new Button("D");
Button b5=new Button("E");
Button b6=new Button("F");
Button b7=new Button("G");
Button b8=new Button("H");
Button b9=new Button("I");
Button b10=new Button("J");
add(b1);
add(b2);
add(b3);
add(b4);
add(b5);
add(b6);
add(b7);
add(b8);
add(b9);
add(b10);
}
public Insets getInsets()
{
return new Insets(20,20,20,20);
}}
```

Output :



Result :

Exp no : 12.b	**Card Layout**
Date :	

Aim :

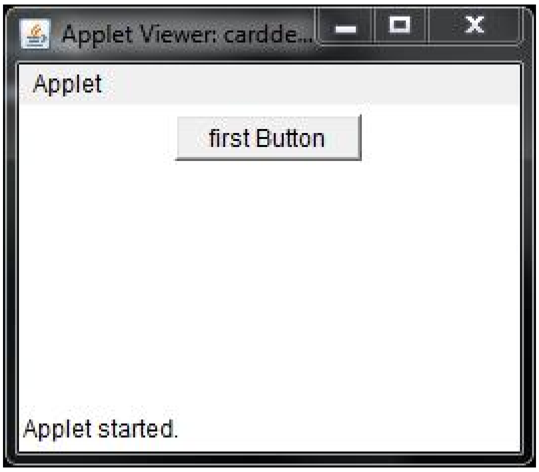
Algorithm:

Source Code :

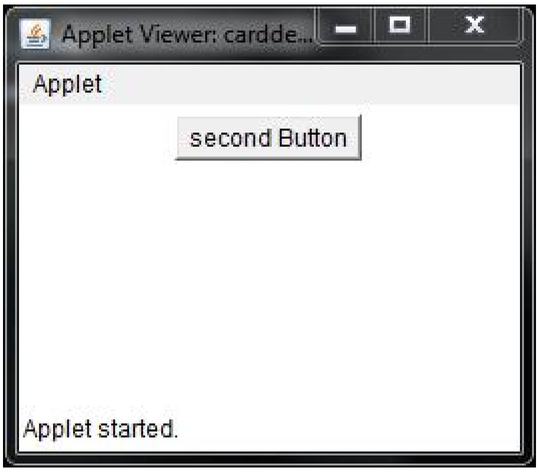
```
import java.io.*;
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
/*
<applet code="carddemo" width=250 height=150>
</applet>
*/

public class carddemo extends Applet implements ActionListener
{
Button b1,b2,b3,b4;
Panel buttonPanel;
CardLayout buttonCardLayout;
public void init()
{
 buttonPanel=new Panel();
add(buttonPanel);
buttonCardLayout=new CardLayout();
buttonPanel.setLayout(buttonCardLayout);
b1=new Button("first Button");
b1.addActionListener(this);
buttonPanel.add(b1,"first Button");
b2=new Button("second Button");
b2.addActionListener(this);
buttonPanel.add(b2,"second Button");
b3=new Button("third button");
b3.addActionListener(this);
buttonPanel.add(b3,"third Button");
}
public void actionPerformed(ActionEvent e)
{
buttonCardLayout.next(buttonPanel);
}
}
```

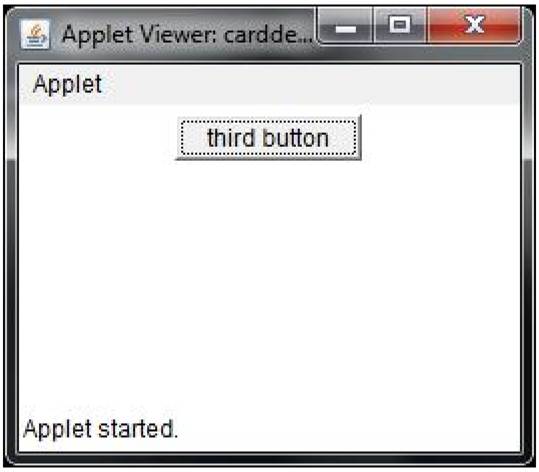
Output :



Applet started.



Applet started.



Applet started.

Result :

Exp no : 12.c	**Border Layout**
Date :	

Aim :

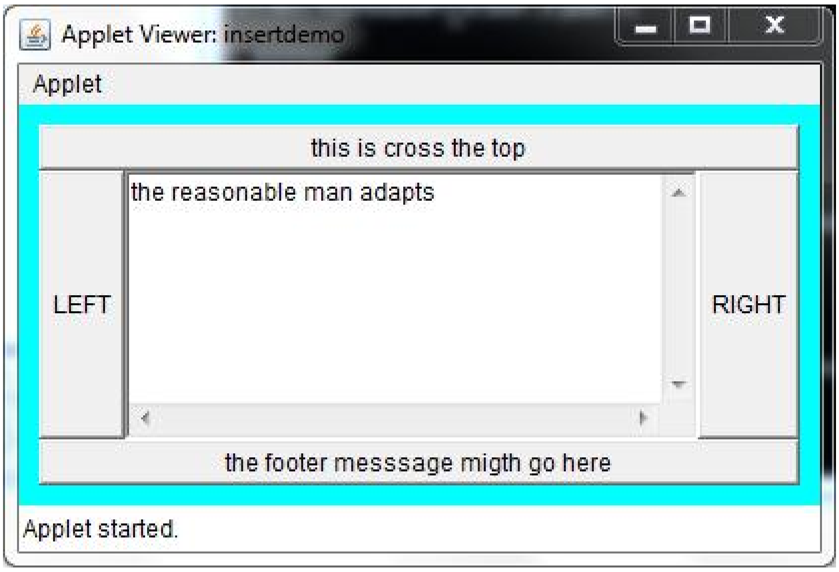
Algorithm:

Source Code :

```
import java.awt.*;
import java.applet.*;
import java.io.*;
import java.util.*;
/*
<applet code="insertdemo" width=400 height=200>
</applet>
*/

public class insertdemo extends Applet
{
public void init()
{
setBackground(Color.cyan);
setLayout(new BorderLayout());
add(new Button("this is cross the top"),BorderLayout.NORTH);
add(new Button("the footer messsage migth go here"),BorderLayout.SOUTH);
add(new Button("RIGHT"),BorderLayout.EAST);
add(new Button("LEFT"),BorderLayout.WEST);
String msg="the reasonable man adapts\n\n";
add(new TextArea(msg),BorderLayout.CENTER);
}
public Insets getInsets()
{
return new Insets(10,10,10,10);
}
}
```

Output :



the reasonable man adapts

Applet started.

Result :

Exp no : 13	STUDENT DATABASE IMPLEMENTATION
Date :	

Aim :

Algorithm:

Source Code :

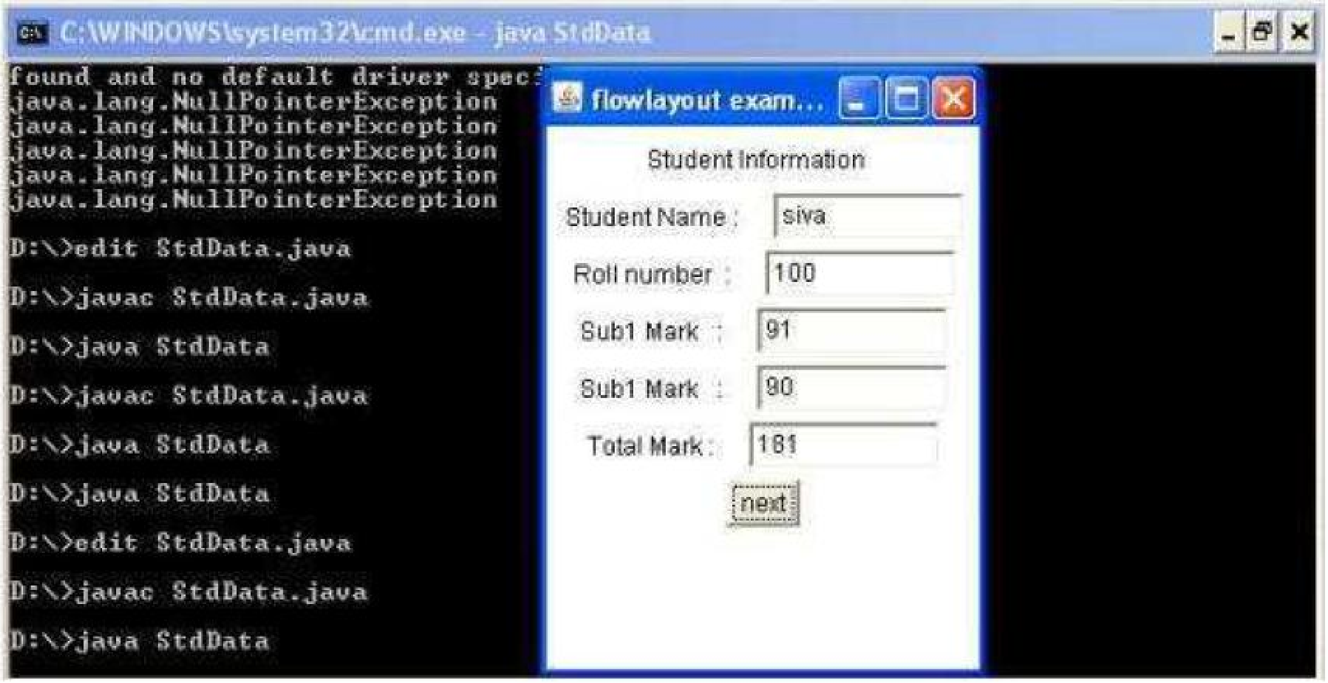
```
import java.awt.*;
 import java.util.*;
 import java.lang.*;
import java.io.*;
import java.sql.*;
 import java.lang.*;
import java.awt.event.*;
 public class StdData extends Frame implements WindowListener,ActionListener {
String tname,tresult,troll=null;
 int ttot=0,tm1,tm2;
 ResultSet rs;
 Connection con=null;
Statement stmt=null;
 Label lblhead=new Label(" Student Information");
Label lbname=new Label("Student Name :");
 Label lbroll=new Label("Roll number :");
Label lbm1=new Label("Sub1 Mark :");
Label lbm2=new Label("Sub1 Mark :");
 Label lbtot=new Label("Total Mark :");
TextField txtfldname = new TextField(10);
TextField txtfldroll = new TextField(10);
TextField txtfldm1 = new TextField(10);
TextField txtfldm2 = new TextField(10);
 TextField txtfldtot = new TextField(10);
 Button btnclick;
public static void main(String[] args) {
StdData myWindow = new StdData("flowlayout example");
 myWindow.setSize(220,300);
myWindow.setVisible(true);
myWindow.databaseInit();
}
public StdData(String title)
 {
super(title);
setLayout(new FlowLayout()); addWindowListener(this);
btnclick= new Button("next");
add(lblhead);
 add(lbname);
 add(txtfldname);
add(lbroll);
```

```
     add(txtfldroll);
      add(lbm1);
     add(txtfldm1);
      add(lbm2);
     add(txtfldm2);
     add(lbtot);
     add(txtfldtot);
     add(btnclick);
     btnclick.addActionListener(this);
      }

     public void databaseInit()
      {
      try { }
     Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
con=DriverManager.getConnection("Jdbc:Odbc:dsn1");
     stmt=con.createStatement();
      rs=stmt.executeQuery("Select * from std");
      catch(SQLException slqe)
      {
     System.out.println("sql"+slqe);
      }
     catch(ClassNotFoundException clnf)
      {
     System.out.println("sql"+clnf);
      }
     }
     public void actionPerformed(ActionEvent e)
     {
     String btnstr=e.getActionCommand();
      if(btnstr.equals("next"))
      {
     Try
      {
      if(rs.next())
      {
     tname=rs.getString("name");
     txtfldname.setText(tname);
      troll=rs.getString("roll");
      txtfldroll.setText(troll);
      tm1=rs.getInt("m1");
      tm2=rs.getInt("m2");
      ttot=tm1+tm2;
      txtfldm2.setText(Integer.toString(tm1));
      txtfldm1.setText(Integer.toString(tm2));
txtfldtot.setText(Integer.toString(ttot));
 }
 }
 catch(Exception e1)
 {
 System.out.println(e1);
 }
}
 }
 public void windowClosing(WindowEvent e)
 {
 dispose();
System.exit(0);
 }
 public void windowOpened(WindowEvent e) {}
public void windowActivated(WindowEvent e) {}
 public void windowIconified(WindowEvent e) {}
 public void windowDeiconified(WindowEvent e) {}
 public void windowDeactivated(WindowEvent e) {}
public void windowClosed(WindowEvent e) {}

 }
```

Output :



```
C:\WINDOWS\system32\cmd.exe - java StdData
found and no default driver spec
java.lang.NullPointerException
java.lang.NullPointerException
java.lang.NullPointerException
java.lang.NullPointerException
java.lang.NullPointerException

D:\>edit StdData.java

D:\>javac StdData.java

D:\>java StdData

D:\>javac StdData.java

D:\>java StdData

D:\>java StdData

D:\>edit StdData.java

D:\>javac StdData.java

D:\>java StdData
```

flowlayout exam...

Student Information

Student Name : siva

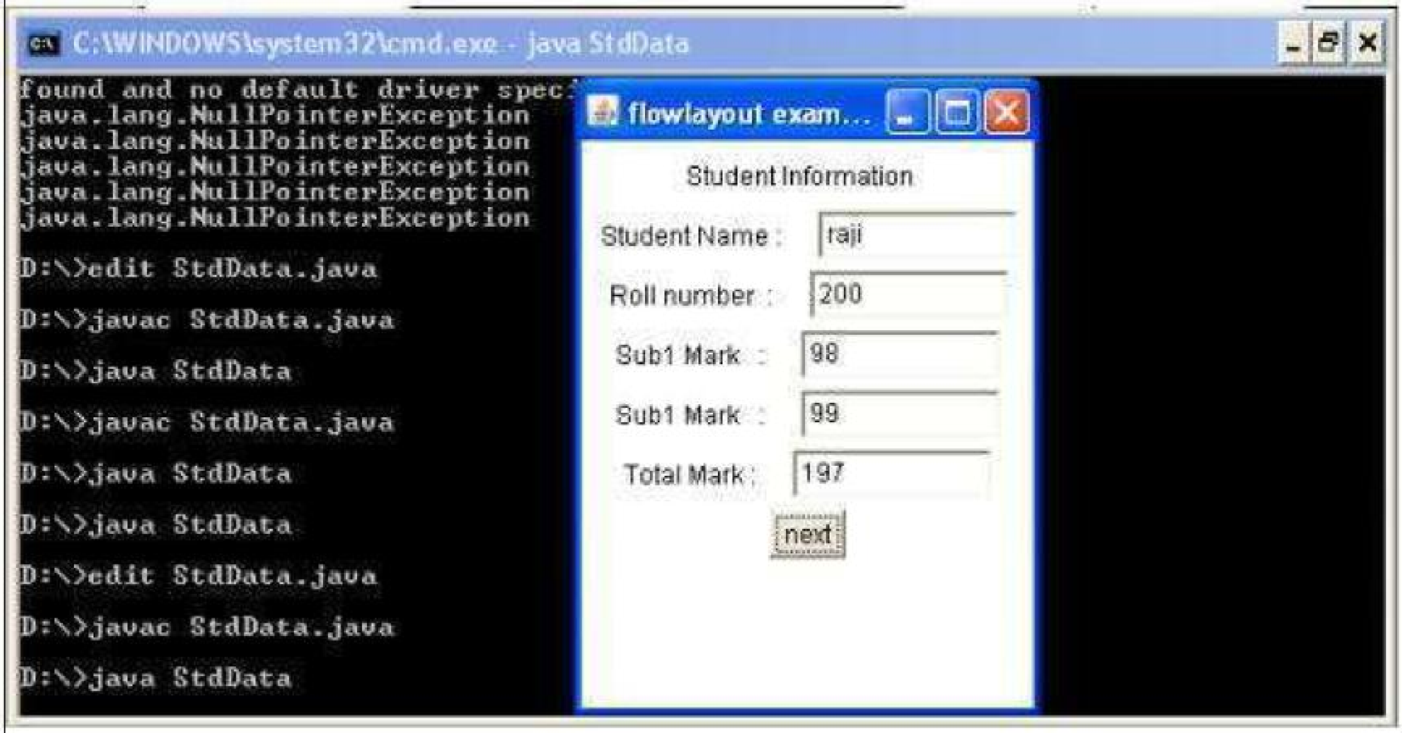
Roll number : 100

Sub1 Mark : 91

Sub1 Mark : 90

Total Mark : 181

Click next button



```
C:\WINDOWS\system32\cmd.exe - java StdData
found and no default driver spec
java.lang.NullPointerException
java.lang.NullPointerException
java.lang.NullPointerException
java.lang.NullPointerException
java.lang.NullPointerException

D:\>edit StdData.java

D:\>javac StdData.java

D:\>java StdData

D:\>javac StdData.java

D:\>java StdData

D:\>java StdData

D:\>edit StdData.java

D:\>javac StdData.java

D:\>java StdData
```

flowlayout exam...

Student Information

Student Name : raji

Roll number : 200

Sub1 Mark : 98

Sub1 Mark : 99

Total Mark : 197

Result :