**Program :**

import java.util.Scanner ;

import java.io.\* ;

public class QuadraticEquationSolving

{

public static void main(String [] array)

{

Scanner s = new Scanner(System.in) ;

double a, b, c, r1, r2 ;

System.out.println("\nRoots of quadratic equation\nThe format is\nax2+bx+c") ;

System.out.print("Enter a value :\t") ;

a = s.nextInt() ;

System.out.print("\nEnter b value :\t") ;

b = s.nextInt() ;

System.out.print("\nEnter c value :\t") ;

c = s.nextInt() ;

double det = b\*b - 4\*a\*c ;

if(det > 0)

{

r1 = (-b + Math.sqrt(det)) / (2\*a) ;

r2 = (-b - Math.sqrt(det)) / (2\*a) ;

System.out.format("\nr1 = %.2f\nr2 = %.2f\nRoots are Real and Distinct.", r1, r2) ;

}

else if(det == 0)

{

r1 = r2 = -b / (2\*a) ;

System.out.format("\nr1 = %.2f\nr2 = %.2f\nRoots are Real and Equal.", r1, r2) ;

}

else

{

double real = -b / (2\*a) ;

double img = Math.sqrt(-det) / (2\*a) ;

System.out.format("\nr1 = %.2f + %.2f i", real, img) ;

System.out.format("\nr2 = %.2f + %.2f i", real, img) ;

System.out.print("\nRoots are imaginary.") ;

}

}

}

**Output :**

Roots of quadratic equation

The format is

ax2+bx+c

Enter a value : 2

Enter b value : 2

Enter c value : 2

r1 = -0.50 + 0.87 i

r2 = -0.50 + 0.87 i

Roots are imaginary.

**Program :**

import java.util.Scanner ;

public class FibonacciSeries

{

public static void main(String [] array)

{

Scanner s = new Scanner(System.in) ;

System.out.print("Fibonacci Series\nEnter no.of items to be printed :\t") ;

int n = s.nextInt() ;

int a = 0, b = 1, c = 0 ;

System.out.print("\n0\t1\t") ;

for(int i = 0 ; i <= n-2 ; i++)

{

c = a + b ;

a = b ;

b= c ;

System.out.print(c+"\t") ;

}

}

}

**Output :**

Fibonacci Series

Enter no.of items to be printed : 7

0 1 1 2 3 5 8 13

**Program :**

import java.util.Scanner ;

import java.io.\* ;

public class StringSorting

{

public static void main(String [] array)

{

Scanner s = new Scanner(System.in) ;

String temp ;

System.out.print("String Sorting\nEnter no.of Strings :\t") ;

int n = s.nextInt() ;

String str[] = new String[n] ;

System.out.println("\nEnter Strings :") ;

int i, j ;

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

{

str[i] = s.next() ;

}

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

{

for(j = 0 ; j < n ; j++)

{

if((str[i].compareTo(str[j])) < 0)

{

temp = str[i] ;

str[i] = str[j] ;

str[j] = temp ;

}

}

}

System.out.println("\nSorted Strings :") ;

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

{

System.out.println(str[i]) ;

}

}

}

**Output :**

Enter no.of Strings : 5

Enter Strings :

SteveRogers

TonyStark

ChrisHemsworth

MarkRuffalo

TomHolland

Sorted Strings :

ChrisHemsworth

MarkRuffalo

SteveRogers

TomHolland

TonyStark

**Program :**

import java.util.Scanner ;

public class PrimeNumber

{

public static void main(String [] array)

{

Scanner s = new Scanner(System.in) ;

System.out.print("Enter a number to check if it is truly prime number or not :\t") ;

int n= s.nextInt() ;

if(isPrime(n))

{

System.out.print("\n" + n + " is prime number.") ;

}

else

{

System.out.print("\n" + n + " is a non-prime number.") ;

}

}

static boolean isPrime(int num)

{

if(num <= 1)

{

return false ;

}

for(int i = 2 ; i <= num/2 ; i++)

{

if((num % i) == 0)

{

return false ;

}

}

return true ;

}

}

**Output :**

Enter a number to check if it is truly prime number or not : 209

209 is a non-prime number.

**Program :**

import java.io.\* ;

import java.util.Scanner ;

public class MatrixMultiplication

{

public static void main(String [] array)

{

System.out.print("Matrix Multiplication\n") ;

int r1, r2, c1, c2 ;

Scanner s = new Scanner(System.in) ;

System.out.print("For First matrix\nEnter no.of rows :\t") ;

r1 = s.nextInt() ;

System.out.print("\nEnter no.of columns :\t") ;

c1 = s.nextInt() ;

System.out.print("\nFor Second matrix\nEnter no.of rows :\t") ;

r2 = s.nextInt() ;

System.out.print("\nEnter no.of columns :\t") ;

c2 = s.nextInt() ;

if(c1 != r2)

{

System.out.println("\nMatrix multiplication is not possible.");

System.exit(0) ;

}

int i, j, k ;

int [][] firstMatrix = new int [r1][c1] ;

int [][] secondMarix = new int [r2][c2] ;

System.out.println("\nEnter the elements of First Matrix :") ;

for (i = 0 ; i < r1 ; i++)

{

for(j = 0 ; j < c1 ; j++)

{

firstMatrix[i][j] = s.nextInt() ;

}

}

System.out.println("Enter the elements of Second Matrix :");

for (i = 0 ; i < r2 ; i++)

{

for(j = 0 ; j < c2 ; j++)

{

secondMarix[i][j] = s.nextInt() ;

}

}

int [][] product = new int [r1][c2] ;

for(i = 0 ; i < r1 ; i++)

{

for(j = 0 ; j < c2 ; j++)

{

for(k = 0 ; k < c1 ; k++)

{

product[i][j] += firstMatrix[i][k] \* secondMarix[k][j] ;

}

}

}

System.out.println("Multiplication of given two matrix is :") ;

for(int row[] : product)

{

for(int column : row)

{

System.out.print(column + " ") ;

}

System.out.println();

}

}

}

**Output :**

Matrix Multiplication

For First matrix

Enter no.of rows : 2

Enter no.of columns : 3

For Second matrix

Enter no.of rows : 3

Enter no.of columns : 2

Enter the elements of First Matrix :

3

-2

5

3

0

4

Enter the elements of Second Matrix :

2

3

-9

0

0

4

Multiplication of given two matrix is :

24 29

6 25

**Program:**

import java.util.Scanner ;

public class EB

{

public static void main(String [] array)

{

double amt ;

Scanner s = new Scanner(System.in) ;

System.out.print("Enter your card no :\t") ;

int cno = s.nextInt() ;

System.out.print("\nEnter your name :\t") ;

String cname = s.next() ;

System.out.print("\nEnter your previous reading :\t") ;

int previous = s.nextInt() ;

System.out.print("\nEnter your current reading :\t") ;

int current = s.nextInt() ;

System.out.print("\n1.Domestic\n2.Commercial\nEnter your connection type :\t") ;

int type = s.nextInt() ;

double units = current - previous ;

double dunits = units - 100 ;

switch(type)

{

case 1:

if(units <= 100)

{

amt = units ;

}

else if(dunits <= 200 && dunits >= 101)

{

amt = 100 + (dunits \* 2.50) ;

}

else if(dunits <= 500 && dunits >= 201)

{

amt = 100 + (dunits \* 4) ;

}

else

{

amt = 100 + (dunits \* 6);

}

System.out.print("\nCustomer no :\t" + cno + "\nCustomer name :\t" + cname + "\nUnits :\t" + units + "\nBill Amt :\t" + amt ) ;

break ;

case 2:

if(units <= 100)

{

amt = units \* 2 ;

}

else if(dunits <= 200 && dunits >= 101)

{

amt = 100 + (dunits \* 4.50) ;

}

else if(dunits <= 500 && dunits >= 201)

{

amt = 100 + (dunits \* 6) ;

}

else

{

amt = 100 + (dunits \* 7) ;

}

System.out.print("\nCustomer no :\t" + cno + "\nCustomer name :\t" + cname + "\nUnits :\t" + units + "\nBill Amt :\t" + amt ) ;

break ;

default:

System.out.print("\nYour choice is invalid") ;

}

}

}

**Output :**

Enter your card no : 209

Enter your name : Tony\_Stark

Enter your previous reading : 601

Enter your current reading : 1205

1.Domestic

2.Commercial

Enter your connection type : 1

Customer no : 209

Customer name : Tony\_Stark

Units : 604.0

Bill Amt : 3124.0