**Q1. Write a Java program to calculate Sum & Average of an integer array.**

import java.util.\*;

// Author : Abhishek Sharma

public class a3q1 {

    public static void main(String args[]){

        Scanner sc = new Scanner (System.in);

        System.out.print("Enter the number of elements : ");

        int n = sc.nextInt();

        int s = 0;

        int[] array = new int[100];

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

        for (int i=0; i<n;i++){

            array[i] = sc.nextInt();

            s = s + array[i];

        }

        float avg = s/n;

        System.out.println("Sum of the elements : "+ s);

        System.out.println("Average of the elements : "+ avg);

    }

}

**Output :**

Enter the number of elements : 5

Enter the elements of the array :

2

3

5

4

6

Sum of the elements : 20

Average of the elements : 4.0

**Q2. Write a Java program to implement stack using array.**

import java.util.\*;

// Author : Abhishek Sharma

class Stack

{

    int top;

    int maxsize = 10;

    int[] arr = new int[maxsize];

    boolean isEmpty()

    {

        return (top < 0);

    }

    Stack()

    {

        top = -1;

    }

    boolean push (Scanner sc)

    {

        if(top == maxsize-1)

        {

            System.out.println("Overflow !!");

            return false;

        }

        else

        {

            System.out.println("Enter Value");

            int val = sc.nextInt();

            top++;

            arr[top]=val;

            System.out.println("Item pushed");

            return true;

        }

    }

    boolean pop ()

    {

        if (top == -1)

        {

            System.out.println("Underflow !!");

            return false;

        }

        else

        {

            top --;

            System.out.println("Item popped");

            return true;

        }

    }

    void display ()

    {

        System.out.println("Printing stack elements .....");

        for(int i = top; i>=0;i--)

        {

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

        }

    }

}

public class a3q2 {

public static void main(String[] args) {

    int choice=0;

    Scanner sc = new Scanner(System.in);

    Stack s = new Stack();

    System.out.println("\*\*\*\*\*\*\*\*\*Stack operations using array\*\*\*\*\*\*\*\*\*\n");

    System.out.println("\n------------------------------------------------\n");

    while(choice != 4)

    {

        System.out.println("\nChose one from the below options...\n");

        System.out.println("\n1.Push\n2.Pop\n3.Show\n4.Exit");

        System.out.println("\n Enter your choice \n");

        choice = sc.nextInt();

        switch(choice)

        {

            case 1:

            {

                s.push(sc);

                break;

            }

            case 2:

            {

                s.pop();

                break;

            }

            case 3:

            {

                s.display();

                break;

            }

            case 4:

            {

                System.out.println("Exiting....");

                System.exit(0);

                break;

            }

            default:

            {

                System.out.println("Please Enter valid choice ");

            }

        };

    }

}

}

**Output :**

\*\*\*\*\*\*\*\*\*Stack operations using array\*\*\*\*\*\*\*\*\*

------------------------------------------------

Chose one from the below options...

1.Push

2.Pop

3.Show

4.Exit

Enter your choice

1

Enter Value

5

Item pushed

Chose one from the below options...

1.Push

2.Pop

3.Show

4.Exit

Enter your choice

1

Enter Value

6

Item pushed

Chose one from the below options...

1.Push

2.Pop

3.Show

4.Exit

Enter your choice

1

Enter Value

7

Item pushed

Chose one from the below options...

1.Push

2.Pop

3.Show

4.Exit

Enter your choice

3

Printing stack elements .....

7

6

5

**Q3. Write a Java program to implement Queue using array.**

// Author : Abhishek Sharma

class Queue {

    private static int front, rear, capacity;

    private static int queue[];

    Queue(int c)

    {

        front = rear = 0;

        capacity = c;

        queue = new int[capacity];

    }

    static void queueEnqueue(int data)

    {

        if (capacity == rear) {

            System.out.printf("\nQueue is full\n");

            return;

        }

        else {

            queue[rear] = data;

            rear++;

        }

        return;

    }

    static void queueDequeue()

    {

        if (front == rear) {

            System.out.printf("\nQueue is empty\n");

            return;

        }

        else {

            for (int i = 0; i < rear - 1; i++) {

                queue[i] = queue[i + 1];

            }

            if (rear < capacity)

                queue[rear] = 0;

            rear--;

        }

        return;

    }

    static void queueDisplay()

    {

        int i;

        if (front == rear) {

            System.out.printf("\nQueue is Empty\n");

            return;

        }

        for (i = front; i < rear; i++) {

            System.out.printf(" %d <-- ", queue[i]);

        }

        return;

    }

    static void queueFront()

    {

        if (front == rear) {

            System.out.printf("\nQueue is Empty\n");

            return;

        }

        System.out.printf("\nFront Element is: %d", queue[front]);

        return;

    }

}

public class a3q3 {

    // Driver code

    public static void main(String[] args)

    {

        // Create a queue of capacity 4

        Queue q = new Queue(4);

        // print Queue elements

        q.queueDisplay();

        // inserting elements in the queue

        q.queueEnqueue(20);

        q.queueEnqueue(30);

        q.queueEnqueue(40);

        q.queueEnqueue(50);

        // print Queue elements

        q.queueDisplay();

        // insert element in the queue

        q.queueEnqueue(60);

        // print Queue elements

        q.queueDisplay();

        q.queueDequeue();

        q.queueDequeue();

        System.out.printf("\n\nafter two node deletion\n\n");

        // print Queue elements

        q.queueDisplay();

        // print front of the queue

        q.queueFront();

    }

}

**Output :**

Queue is Empty

20 <-- 30 <-- 40 <-- 50 <--

Queue is full

20 <-- 30 <-- 40 <-- 50 <--

after two node deletion

40 <-- 50 <--

Front Element is: 40

**Q4. . Write a Java program to calculate Sum of two 2-dimensional arrays.**

// Author : Abhishek Sharma

import java.util.\*;

public class a3q4

{

    public static void main(String args[])

    {

        int row,col;

        Scanner sc=new Scanner(System.in);

        //Read number of rows and cols

        System.out.print("Input number of rows: ");

        row=sc.nextInt();

        System.out.print("Input number of rows: ");

        col=sc.nextInt();

        //declare two dimensional array (matrices)

        int a[][]=new int[row][col];    //for matrix 1

        int b[][]=new int[row][col];    //for matrix 2

        int c[][]=new int[row][col];    //for matrix 3

        //Read elements of Matrix a

        System.out.println("Enter elements of matrix a:");

        for(int i=0; i<row; i++)

        {

            for(int j=0; j<col; j++)

            {

                System.out.print("Element [" + (i+1) + "," + (j+1) + "] ? ");

                a[i][j]=sc.nextInt();

            }

        }

        //Read elements of Matrix b

        System.out.println("Enter elements of matrix b:");

        for(int i=0; i<row; i++)

        {

            for(int j=0; j<col; j++)

            {

                System.out.print("Element [" + (i+1) + "," + (j+1) + "] ? ");

                b[i][j]=sc.nextInt();

            }

        }

        //print matrix a

        System.out.println("Matrix a:");

        for(int i=0; i<row; i++){

            for(int j=0; j<col; j++){

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

            }

            System.out.print("\n");

        }

        //print matrix b

        System.out.println("Matrix b:");

        for(int i=0; i<row; i++){

            for(int j=0; j<col; j++){

                System.out.print(b[i][j] + "\t");

            }

            System.out.print("\n");

        }

        //adding matrices

        for(int i=0; i<row; i++){

            for(int j=0; j<col; j++){

                c[i][j]=a[i][j] + b[i][j];

            }

        }

        //print matrix b

        System.out.println(":: Final Matrix:");

        for(int i=0; i<row; i++){

            for(int j=0; j<col; j++){

                System.out.print(c[i][j] + "\t");

            }

            System.out.print("\n");

        }

    }

}

**Output :**

Input number of rows: 2

Input number of rows: 2

Enter elements of matrix a:

Element [1,1] ? 1

Element [1,2] ? 1

Element [2,1] ? 1

Element [2,2] ? 1

Enter elements of matrix b:

Element [1,1] ? 2

Element [1,2] ? 2

Element [2,1] ? 2

Element [2,2] ? 2

Matrix a:

1 1

1 1

Matrix b:

2 2

2 2

:: Final Matrix:

3 3

3 3

**Q5. Write a Java program to find the range of a 1D array.**

import java.io.\*;

// Author : Abhishek Sharma

class a3q5 {

static float getMin(float arr[], int n)

{

    float res = arr[0];

    for (int i = 1; i < n; i++)

        res = Math.min(res, arr[i]);

    return res;

}

static float getMax(float arr[], int n)

{

    float res = arr[0];

    for (int i = 1; i < n; i++)

        res = Math.max(res, arr[i]);

    return res;

}

static void findRangeAndCoefficient(float arr[], int n)

{

    float max = getMax(arr, n);

    float min = getMin(arr, n);

    float range = max - min;

    System.out.println("Range : " + range );

}

    public static void main (String[] args) {

    float arr[] = { 5, 10, 15, 20, 56, 54 };

    int n = arr.length;

    findRangeAndCoefficient(arr, n);

    }

}

**Output :**

Range : 51.0

**Q6. Write a Java program to search an element in an array.**

import java.util.Scanner;

// Author : Abhishek Sharma

class a3q6

{

  public static void main(String args[])

  {

    int c, n, search, array[];

    Scanner in = new Scanner(System.in);

    System.out.println("Enter number of elements");

    n = in.nextInt();

    array = new int[n];

    System.out.println("Enter those " + n + " elements");

    for (c = 0; c < n; c++)

      array[c] = in.nextInt();

    System.out.println("Enter value to find");

    search = in.nextInt();

    for (c = 0; c < n; c++)

    {

      if (array[c] == search)     /\* Searching element is present \*/

      {

         System.out.println(search + " is present at location " + (c + 1) + ".");

          break;

      }

   }

   if (c == n)  /\* Element to search isn't present \*/

      System.out.println(search + " isn't present in array.");

  }

}

**Output :**

Enter number of elements

5

Enter those 5 elements

5

6

77

58

100

Enter value to find

77

77 is present at location 3.

**Q7. Write a Java program to find the sum of even numbers in an integer array.**

import java.util.Scanner;

// Author : Abhishek Sharma

public class a3q7 {

    private static Scanner sc;

    public static void main(String[] args)

    {

        int Size, i, EvenSum = 0;

        sc = new Scanner(System.in);

        System.out.print(" Please Enter Number of elements in an array : ");

        Size = sc.nextInt();

        int [] a = new int[Size];

        System.out.print(" Please Enter " + Size + " elements of an Array  : ");

        for (i = 0; i < Size; i++)

        {

            a[i] = sc.nextInt();

        }

        for(i = 0; i < Size; i++)

        {

            if(a[i] % 2 == 0)

            {

                EvenSum = EvenSum + a[i];

            }

        }

        System.out.println("\n The Sum of Even Numbers in this Array = " + EvenSum);

    }

}

**Output :**

Please Enter Number of elements in an array : 5

Please Enter 5 elements of an Array : 56

12

45

97

100

The Sum of Even Numbers in this Array = 168

**Q8. Write a Java program to find the sum of diagonal elements in a 2D array.**

import java.io.\*;

// Author : Abhishek Sharma

public class a3q8 {

    static void printDiagonalSums(int [][]mat,

                                        int n)

    {

        int principal = 0, secondary = 0;

        for (int i = 0; i < n; i++) {

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

                // Condition for principal

                // diagonal

                if (i == j)

                    principal += mat[i][j];

                // Condition for secondary

                // diagonal

                if ((i + j) == (n - 1))

                    secondary += mat[i][j];

            }

        }

        System.out.println("Principal Diagonal : "

                                    + principal);

        System.out.println("Secondary Diagonal : "

                                    + secondary);

    }

    // Driver code

    static public void main (String[] args)

    {

        int [][]a = { { 1, 2, 3, 4 },

                    { 5, 6, 7, 8 },

                    { 1, 2, 3, 4 },

                    { 5, 6, 7, 8 } };

        printDiagonalSums(a, 4);

    }

}

**Output :**

Principal Diagonal : 18

Secondary Diagonal : 18

**Q9. Reverse the elements in an array of integers without using a second array.**

import java.util.Arrays;

// Author : Abhishek Sharma

public class a3q9 {

    public int[] reverse(int [] array){

        if(array == null || array.length <= 1){

            System.out.println("Invalid array.");

        }

        for (int i = 0; i < array.length / 2; i++) {

            int temp = array[i];

            array[i] = array[array.length - 1 - i];

            array[array.length - 1 - i] = temp;

        }

        return array;

    }

    public static void main(String[] args){

        a3q1 arrayReverse = new a3q1();

        int[] input = {1, 2, 3, 4, 5, 6, 7, 8};

        System.out.println("Original array" + Arrays.toString(input));

        System.out.println("Reversed array" + Arrays.toString(arrayReverse.reverse(input)));

    }

}

**Output :**

Original array[1, 2, 3, 4, 5, 6, 7, 8]

Reversed array[8, 7, 6, 5, 4, 3, 2, 1]

**Q10. Write a Java program to enter n elements in an array and find smallest number among them.**

import java.util.Scanner;

// Author : Abhishek Sharma

public class a3q10{

    public static int getSmallest(int[] a, int total){

    int temp;

    for (int i = 0; i < total; i++)

            {

                for (int j = i + 1; j < total; j++)

                {

                    if (a[i] > a[j])

                    {

                        temp = a[i];

                        a[i] = a[j];

                        a[j] = temp;

                    }

                }

            }

           return a[0];

    }

    public static void main(String args[]){

    Scanner sc = new Scanner (System.in);

    System.out.print("Enter the no. of elements : ");

    int z = sc.nextInt();

    int[] array = new int[100];

    System.out.print("Enter the elements : ");

    for (int i = 0; i<z;i++){

        array[i] = sc.nextInt();

    }

    System.out.println("Smallest: "+getSmallest(array,z));

    }

}

**Output :**

Enter the no. of elements : 5

Enter the elements : 69

58

41

23

55

Smallest: 23

**Q11. Write Java program to find the sum of all odd numbers in a 2D array.**

import java.util.Scanner;

// Author : Abhishek Sharma

public class a3q11 {

    private static Scanner sc;

    public static void main(String[] args)

    {

        int OddSum = 0;

        sc = new Scanner(System.in);

        System.out.print(" Please Enter Number of rows in an array : ");

        int row = sc.nextInt();

        System.out.print(" Please Enter Number of columns in an array : ");

        int col = sc.nextInt();

        int [][] a = new int[row][col];

        System.out.print(" Please Enter elements of an Array  : ");

        for (int j=0;j<row;j++){

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

            {

                a[i][j] = sc.nextInt();

                if (a[i][j] %2 != 0){

                    OddSum = OddSum + a[i][j];

                }

            }

        }

        System.out.println("\n The Sum of Odd Numbers in this Array = " + OddSum);

    }

}

**Output :**

Please Enter Number of rows in an array : 2

Please Enter Number of columns in an array : 2

Please Enter elements of an Array : 3

4

6

7

The Sum of Odd Numbers in this Array = 10

**Q12. Write a Java program to print transpose of matrix.**

// Author : Abhishek Sharma

public class a3q12{

    public static void main(String args[]){

    int original[][]={{1,3,4},{2,4,3},{3,4,5}};

    int transpose[][]=new int[3][3];

    for(int i=0;i<3;i++){

    for(int j=0;j<3;j++){

    transpose[i][j]=original[j][i];

    }

    }

    System.out.println("Printing Matrix without transpose:");

    for(int i=0;i<3;i++){

    for(int j=0;j<3;j++){

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

    }

    System.out.println();

    }

    System.out.println("Printing Matrix After Transpose:");

    for(int i=0;i<3;i++){

    for(int j=0;j<3;j++){

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

    }

    System.out.println();

    }

    }

}

**Output :**

Printing Matrix without transpose:

1 3 4

2 4 3

3 4 5

Printing Matrix After Transpose:

1 2 3

3 4 4

4 3 5

**Q13. Write a Java program to check whether a given matrix is sparse or not.**

// Author : Abhishek Sharma

import java.util.Scanner;

public class a3q13

{

    public static void main(String args[])

    {

    int i, j, zero = 0, count = 0;

    int array[][] = new int[10][10];

    System.out.println("Enter total rows and columns: ");

    Scanner s = new Scanner(System.in);

    int row = s.nextInt();

    int column = s.nextInt();

    System.out.println("Enter matrix:");

        for(i = 0; i < row; i++)

        {

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

            {

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

                 System.out.print(" ");

             }

        }

        for(i = 0; i < row; i++)

        {

        for(j = 0; j < column; j++)

        {

                if(array[i][j] == 0)

            {

                zero++;

            }

            else

                {

                    count++;

                }

       }

       }

       if(zero>count)

       {

           System.out.println("the matrix is sparse matrix");

       }

       else

       {

           System.out.println("the matrix is not a sparse matrix");

       }

    }

}

**Output :**

Enter total rows and columns:

2 2

Enter matrix:

1

0

5

0

the matrix is not a sparse matrix

**Q14. Write a Java program to count the prime numbers in an array.**

import java.util.Arrays;

import java.util.Vector;

// Author : Abhishek Sharma

class a3q14

{

    static int primeCount(int arr[], int n)

    {

        int max\_val = Arrays.stream(arr).max().getAsInt();

        Boolean[] prime = new Boolean[max\_val + 1];

        for (int i = 0; i < max\_val + 1; i++)

        {

            prime[i] = true;

        }

        prime[0] = false;

        prime[1] = false;

        for (int p = 2; p \* p <= max\_val; p++)

        {

            if (prime[p] == true)

            {

                for (int i = p \* 2; i <= max\_val; i += p)

                {

                    prime[i] = false;

                }

            }

        }

        int count = 0;

        for (int i = 0; i < n; i++)

        {

            if (prime[arr[i]])

            {

                count++;

            }

        }

        return count;

    }

    public static void main(String[] args)

    {

        int arr[] = {1, 2, 3, 4, 5, 6, 7};

        int n = arr.length;

        System.out.println(primeCount(arr, n));

    }

}

**Output :**

4

**Q15. Write a Java program to find second highest element of an array.**

import java.util.\*;

// Author : Abhishek Sharma

class a3q15{

static void print2largest(int arr[],

                        int arr\_size)

{

int i, first, second;

if (arr\_size < 2)

{

    System.out.printf(" Invalid Input ");

    return;

}

Arrays.sort(arr);

for (i = arr\_size - 2; i >= 0; i--)

{

    if (arr[i] != arr[arr\_size - 1])

    {

    System.out.printf("The second largest " +

                        "element is %d\n", arr[i]);

    return;

    }

}

System.out.printf("There is no second " +

                    "largest element\n");

}

// Driver code

public static void main(String[] args)

{

int arr[] = {12, 35, 1, 10, 34, 1};

int n = arr.length;

print2largest(arr, n);

}

}

**Output :**

The second largest element is 34

**Q16. Write a Java program which counts the non-zero elements in an integer array.**

import java.util.\*;

// Author : Abhishek Sharma

class a3q16{

    public static void main(String args[]) {

        Scanner sc = new Scanner (System.in);

        System.out.print("Enter the no. of elements : ");

        int n = sc.nextInt();

        int count = 0;

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

        int [] arr = new int[100];

        for (int i=0;i<n;i++){

            arr[i] = sc.nextInt();

            if (arr[i]!=0){

                count++;

            }

        }

        System.out.println("No. of non-zero elements  : " + count);

    }

}

**Output :**

Enter the no. of elements : 5

Enter the elements :

1

0

0

3

0

No. of non-zero elements : 2

**Q17. Write a Java program to merge two float arrays.**

import java.util.Arrays;

// Author : Abhishek Sharma

public class a3q17

{

public static void main(String[] args)

{

float[] firstArray = {23f,45f,12f,78f,4f,90f,1f};

float[] secondArray = {77f,11f,45f,88f,32f,56f,3f};

int fal = firstArray.length;

int sal = secondArray.length;

float[] result = new float[fal + sal];

System.arraycopy(firstArray, 0, result, 0, fal);

System.arraycopy(secondArray, 0, result, fal, sal);

System.out.println(Arrays.toString(result));

}

}

**Output :**

[23.0, 45.0, 12.0, 78.0, 4.0, 90.0, 1.0, 77.0, 11.0, 45.0, 88.0, 32.0, 56.0, 3.0]

**Q18. Write a Java program where elements of two integer arrays get added index wise and get stored into a third array.**

import java.util.\*;

// Author : Abhishek Sharma

class a3q18{

    public static void main(String args[]) {

        Scanner sc = new Scanner (System.in);

        System.out.print("Enter the no. of elements of the 1st Array : ");

        int n1 = sc.nextInt();

        int[] arr1 = new int[100];

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

        for (int i = 0; i< n1;i++){

            arr1[i] = sc.nextInt();

        }

        System.out.print("Enter the no. of elements of the 2nd Array : ");

        int n2 = sc.nextInt();

        int[] arr2 = new int[100];

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

        for (int i = 0; i< n2;i++){

            arr2[i] = sc.nextInt();

        }

        int[] result = new int[100];

        for (int i = 0; i<n1;i++){

            result[i] = arr1[i]+arr2[i];

        }

        System.out.print("The resultant array : ");

        for (int i = 0; i<n1;i++){

            System.out.print(result[i]);

            System.out.print(" ");

        }

    }

}

**Output :**

Enter the no. of elements of the 1st Array : 3

Enter the elements :

1

2

3

Enter the no. of elements of the 2nd Array : 3

Enter the elements :

1

2

3

The resultant array : 2 4 6

**Q19. Write a Java program to multiply two matrices.**

import java.util.Scanner;

// Author : Abhishek Sharma

class a3q19

{

   public static void main(String args[])

   {

      int m, n, p, q, sum = 0, c, d, k;

      Scanner in = new Scanner(System.in);

      System.out.println("Enter the number of rows and columns of first matrix");

      m = in.nextInt();

      n = in.nextInt();

      int first[][] = new int[m][n];

      System.out.println("Enter elements of first matrix");

      for (c = 0; c < m; c++)

         for (d = 0; d < n; d++)

            first[c][d] = in.nextInt();

      System.out.println("Enter the number of rows and columns of second matrix");

      p = in.nextInt();

      q = in.nextInt();

      if (n != p)

         System.out.println("The matrices can't be multiplied with each other.");

      else

      {

         int second[][] = new int[p][q];

         int multiply[][] = new int[m][q];

         System.out.println("Enter elements of second matrix");

         for (c = 0; c < p; c++)

            for (d = 0; d < q; d++)

               second[c][d] = in.nextInt();

         for (c = 0; c < m; c++)

         {

            for (d = 0; d < q; d++)

            {

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

               {

                  sum = sum + first[c][k]\*second[k][d];

               }

               multiply[c][d] = sum;

               sum = 0;

            }

         }

         System.out.println("Product of the matrices:");

         for (c = 0; c < m; c++)

         {

            for (d = 0; d < q; d++)

               System.out.print(multiply[c][d]+"\t");

            System.out.print("\n");

         }

      }

   }

}

**Output :**

Enter the number of rows and columns of first matrix

3 3

Enter elements of first matrix

1 2 3

2 3 4

1 4 6

Enter the number of rows and columns of second matrix

3 3

Enter elements of second matrix

1 1 1

0 1 0

1 0 1

Product of the matrices:

4 3 4

6 5 6

7 5 7

**Q20. Write a Java program to subtract two matrices.**

import java.util.Scanner;

// Author : Abhishek Sharma

public class a3q20

{

   public static void main(String args[])

   {

       int i, j;

       System.out.print("Enter the roes and columns of matrices : ");

       Scanner scan = new Scanner(System.in);

       i = scan.nextInt();

       j = scan.nextInt();

       int mat1[][] = new int[i][j];

       int mat2[][] = new int[i][j];

       int mat3[][] = new int[i][j];

       System.out.print("Enter Matrix 1 Elements : ");

       for(i=0; i<3; i++)

       {

           for(j=0; j<3; j++)

           {

               mat1[i][j] = scan.nextInt();

           }

       }

       System.out.print("Enter Matrix 2 Elements : ");

       for(i=0; i<3; i++)

       {

           for(j=0; j<3; j++)

           {

               mat2[i][j] = scan.nextInt();

           }

       }

       System.out.print("Subtracting Matrices (i.e. Matrix1 - Matrix2)...\n");

       for(i=0; i<3; i++)

       {

           for(j=0; j<3; j++)

           {

               mat3[i][j] = mat1[i][j] - mat2[i][j];

           }

       }

       System.out.print("Result of Matrix1 - Matrix2 is :\n");

       for(i=0; i<3; i++)

       {

           for(j=0; j<3; j++)

           {

               System.out.print(mat3[i][j]+ " ");

           }

           System.out.println();

       }

   }

}

**Output :**

Enter the roes and columns of matrices : 3 3

Enter Matrix 1 Elements : 9 9 9

6 6 6

3 3 3

Enter Matrix 2 Elements : 6 6 6

3 3 3

0 0 0

Subtracting Matrices (i.e. Matrix1 - Matrix2)...

Result of Matrix1 - Matrix2 is :

3 3 3

3 3 3

3 3 3

**Q21. Write a Java program to find duplicate elements in a 1D array and find their frequency of occurrence.**

import java.util.\*;

// Author : Abhishek Sharma

class a3q21{

public static void findCounts(int arr[], int n)

{

    int hash[] = new int[n];

    Arrays.fill(hash, 0);

    int i = 0;

    while (i < n)

    {

        hash[arr[i] - 1]++;

        i++;

    }

    System.out.println("\nDuplicate elements -> Occurrence");

    for(i = 0; i < n; i++)

    {

        if (hash[i]>1){

        System.out.println((i + 1) + " -> " +

                           hash[i]);

        }

    }

}

public static void main(String []args)

{

    Scanner sc = new Scanner (System.in);

    System.out.print("Enter the no. of elements : ");

    int n = sc.nextInt();

    int[] arr = new int[n];

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

    for (int i=0; i<n; i++){

        arr[i] = sc.nextInt();

    }

    findCounts(arr, n);

}

}

**Output :**

Enter the no. of elements : 10

Enter the elements of the array :

1 5 6 4 7 7 8 5 9 9

Duplicate elements -> Occurrence

5 -> 2

7 -> 2

9 -> 2

**Q22. Write a Java program to print every alternate number of a given array.**

import java.util.\*;

// Author : Abhishek Sharma

import java.io.\*;

class a3q22{

static void printAlter(int[] arr, int N)

{

    for(int currIndex = 0; currIndex < N; currIndex++)

    {

        if (currIndex % 2 == 0)

        {

            System.out.print(arr[currIndex] + " ");

        }

    }

}

public static void main(String[] args)

{

    Scanner sc = new Scanner (System.in);

    System.out.print("Enter the no. of elements : ");

    int N = sc.nextInt();

    int[] arr = new int[N];

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

    for (int i=0; i<N; i++){

        arr[i] = sc.nextInt();

    }

    printAlter(arr, N);

}

}

**Output :**

Enter the no. of elements : 6

Enter the elements of the array :

1 2 3 4 5 6

1 3 5

**Q23. . Given are two one-dimensional arrays A & B, which are sorted in ascending order. Write a Java program to merge them into single sorted array C that contains every item from arrays A & B, in ascending order**

import java.util.\*;

import java.lang.\*;

import java.io.\*;

// Author : Abhishek Sharma

class a3q23

{

    public static void mergeArrays(int[] arr1, int[] arr2, int n1,

                                int n2, int[] arr3)

    {

        int i = 0, j = 0, k = 0;

        while (i<n1 && j <n2)

        {

            if (arr1[i] < arr2[j])

                arr3[k++] = arr1[i++];

            else

                arr3[k++] = arr2[j++];

        }

        while (i < n1)

            arr3[k++] = arr1[i++];

        while (j < n2)

            arr3[k++] = arr2[j++];

    }

    public static void main (String[] args)

    {

        Scanner sc = new Scanner (System.in);

        System.out.print("Enter the no. of elements of the first array (A) : ");

        int n1 = sc.nextInt();

        System.out.print("Enter the no. of elements of the second array (B) : ");

        int n2 = sc.nextInt();

        int[] arr1 = new int[n1];

        int[] arr2 = new int[n2];

        System.out.print("Enter the sorted elements of the first array (A) : ");

        for (int i=0;i<n1;i++){

            arr1[i] = sc.nextInt();

        }

        System.out.print("Enter the sorted elements of the second array (B) : ");

        for (int i=0;i<n2;i++){

            arr2[i] = sc.nextInt();

        }

        int[] arr3 = new int[n1+n2];

        mergeArrays(arr1, arr2, n1, n2, arr3);

        System.out.print("\nArray after merging (C) in sorted manner : ");

        for (int i=0; i < n1+n2; i++)

            System.out.print(arr3[i] + " ");

    }

}

**Output :**

Enter the no. of elements of the first array (A) : 4

Enter the no. of elements of the second array (B) : 6

Enter the sorted elements of the first array (A) : 1 4 8 9

Enter the sorted elements of the second array (B) : 2 4 7 9 15 24

Array after merging (C) in sorted manner : 1 2 4 4 7 8 9 9 15 24

**Q24. Write a Java program to show 0-arguments constructor.**

import java.util.\*;

import java.lang.\*;

import java.io.\*;

// Author : Abhishek Sharma

class Main {

    int i;

    private Main() {

      i = 5;

      System.out.println("Constructor is called");

    }

    public static void main(String[] args) {

      // calling the constructor without any parameter

      Main obj = new Main();

      System.out.println("Value of i: " + obj.i);

    }

}

**Output :**

Constructor is called

Value of i: 5

**Q25. Write a Java program to show parameterized constructor.**

class Main {

    String languages;

    Main(String lang) {

      languages = lang;

      System.out.println(languages + " Programming Language");

    }

    public static void main(String[] args) {

        Main obj1 = new Main("Java");

        Main obj2 = new Main("Python");

        Main obj3 = new Main("C");

    }

  }

**Output :**

Java Programming Language

Python Programming Language

C Programming Language

**Q26. Write a class, Grader, which has an instance variable, score, an appropriate constructor and appropriate methods. A method, letterGrade() that returns the letter grade as O/E/A/B/C/F.**

**Now write a demo class to test the Grader class by reading a score from the user, using it to create a Grader object after validating that the value is not negative and is not greater than 100. Finally, call the letterGrade() method to get and print the grade.**

import java.util.Scanner;

// Author : Abhishek Sharma

class Grader {

    public int s;

    public int marks;

    public int m;

    public static void letterGrade(int s){

        if (s>90 && s<=100)

            System.out.println("You Have Obtained : Grade O");

        else if(s>80 && s<=90)

            System.out.println("You Have Obtained : Grade E");

        else if (s>70 && s<=80)

            System.out.println("You Have Obtained : Grade A");

        else if (s>60 && s<= 70)

            System.out.println("You Have Obtained : Grade B");

        else if (s>50 && s<= 60)

            System.out.println("You Have Obtained : Grade C");

        else

            System.out.println("You Have Obtained : Grade F");

    }

    public static void demo(int marks){

        if (marks>=0 && marks<=100){

            int m = marks;

            letterGrade(m);

        }

        else {

            System.out.println("Invalid Input Given! Pls Check");

        }

    }

    public static void main(String[] args) {

        int score;

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter the marks obtained : ");

        score = sc.nextInt();

        demo(score);

    }

  }

**Output :**

Enter the marks obtained : 85

You Have Obtained : Grade E

Enter the marks obtained : 102

Invalid Input Given! Pls Check

Enter the marks obtained : -85

Invalid Input Given! Pls Check

**Q27. Write a class, Commission, which has an instance variable, sales; an appropriate constructor; and a method, commission() that returns the commission.**

**Now write a demo class to test the Commission class by reading a sale from the user, using it to create a Commission object after validating that the value is not negative. Finally, call the commission() method to get and print the commission. If the sales are negative, your demo should print the message “Invalid Input”.**

import java.util.Scanner;

// Author : Abhishek Sharma

class Commission {

    public int s;

    public int marks;

    public int m;

    public static void commission(int s){

        double count = s\*100.00;

        System.out.println("Your sales amount is : "+s);

        System.out.println("As per the regulation per sale provides Rs. 100.00");

        System.out.println("Your Commission amount is : Rs."+count);

    }

    public static void demo(int marks){

        if (marks>=0){

            int m = marks;

            commission(m);

        }

        else {

            System.out.println("Invalid Input Given! Pls Check");

        }

    }

    public static void main(String[] args) {

        int sales;

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter the sales amount : ");

        sales = sc.nextInt();

        demo(sales);

    }

  }

**Output :**

Enter the sales amount : 68

Your sales amount is : 68

As per the regulation per sale provides Rs. 100.00

Your Commission amount is : Rs.6800.0