

Open ▾ +

```
CountPandN.java x High.java x Tpl.java x Untitled Document 1 x Ecal.java x sortlist.java

import java.util.Scanner;
//Q2.

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

        int positives = 0;
        int negatives = 0;
        int count = 0;
        double total = 0;
        System.out.print("Enter an integer, the input ends if it is 0: ");
        int num = input.nextInt();

        if (num == 0) {
            System.out.println("No numbers are entered except 0");
            System.exit(1);
        }

        while (num != 0) {
            if (number > 0)
                positives++;
            else
                negatives++;

            total += num;
            count++;
            num = input.nextInt();
        }

        double average = total / count;
        System.out.println(
            "The number of positive is " + positives +
            "\nThe number of negatives is " + negatives +
            "\nThe total is total " + total +
            "\nThe average is " + average);
    }
}
```

```
Open ▾
```

CountPandN.java x High.java x Tpl.java x Untitled Document 1 x Ecal.java x sortlist.java

```
import java.util.Scanner;
//Q3
public class HIgn {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);

        int highestScore = 0;
        String highestScoreName = "";
        System.out.print("Enter the number of students: ");
        int numberOfStudents = input.nextInt();
        System.out.println("Enter each student's name and score");
        for (int i = 0; i < numberOfStudents; i++) {
            System.out.print(
                "Student: " + (i + 1) +
                "\n  Name: ");
            String name = input.next();
            System.out.print(
                "  Score: ");
            int score = input.nextInt();
            if (score > highestScore) {
                highestScore = score;
                highestScoreName = name;
            }
        }
        System.out.println("Student with the highest score: " + highestScoreName);
    }
}
```

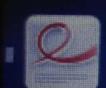
Tpl.java (~/) - gedit



Open ▾



```
CountPandN.java x    High.java x    Tpl.java x    Untitled Document 1 x    Ecal.java x
public class Tpl {
//Q4|
public static void main(String[] args) {
    int count = 1;
    for (int i = 100; i <= 200; i++) {
        if (i % 6 == 0 ^ i % 5 == 0) {
            System.out.print((count++ % 10 != 0) ? i + " " : i + "\n");
        }
    }
}
```





Open ▾ New

CountPandN.java x High.java x Tpl.java x Ecal.java x

```
import java.util.*;
//Q5.
class Ecal
{
    public static void main()
    {
        System.out.print("Enter the value of limit n : ");
        Scanner sc=new Scanner(System.in);
        int n,i;
        n=sc.nextInt();
        double S=0,f=0;
        for(i=1;i<n;i++)
        {
            f=fact(i);
            S=S+(1/f);
        }
        System.out.println("The sum =" +S);
    }

    public static double fact(int num)
    {
        int f=1,j;
        for(j=1;j<=num;j++)
        {
            f=f*j;
        }
        return(f);
    }
}
```

```
import java.util.Scanner;
//Q7.

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

        int[] values = new int[100];
        int input;
        int count = 0;
        System.out.print("Enter the integers between 1 and 100: ");
        do
        {
            input = in.nextInt();
            values[count] = input;
            count += 1;
        }
        while (input != 0);

        countOccurrence(values);
    }

    public static void countOccurrence(int[] list) {
        for (int i = 1; i <= 100; i++) {
            int count = 0;
            for (int j = 0; j < list.length - 1; j++) {
                if (list[j] == i)
                    count += 1;
            }
            if (count != 0)
                System.out.printf("%d occurs %d %s%n",
                    i, count, count > 1 ? "times" : "time");
        }
    }
}
```

sumE.java (~/) - gedit

Open ▾

```
Cou High.java x Tpl.java x Ecal.java x sortlist.java x occ.java x
public class sumE {
//Q8

    public static void main(String[] args) {
        System.out.print("Enter a 3 X 4 matrix: ");
        Scanner input = new Scanner(System.in);

        double[][] matrix = new double[3][4];
        for (int i = 0; i < matrix.length; i++)
            for (int j = 0; j < matrix[i].length; j++)
                matrix[i][j] = input.nextDouble();

        for (int i = 0; i < matrix[0].length; i++) {
            System.out.println("Sum of the elements at column" + i + " is " + sumColumn(matrix, i));
        }
    }

    public static double sumColumn(double[][] m, int columnIndex) {
        double total = 0;

        for (int i = 0; i < m.length; i++) {
            total += m[i][columnIndex];
        }
        return total;
    }

    public static void displayMatrix(double[][] matrix) {
        for (int row = 0; row < matrix.length; row++) {

            for (int column = 0; column < matrix[row].length; column++) {
                System.out.printf("%5.0f ", matrix[row][column]);
            }
            System.out.printf("\n");
        }
    }
}
```



locate.java (~/) - gedit

```
Open ▾ +
```

CountPandN.java x High.java x Tpl.java x Ecal.java x sortlist.java x occ.java x sumE.j

```
public class locate {
//Q9

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);
        System.out.printf("Enter the number of rows and columns of the array: ");
        int row = input.nextInt();
        int column = input.nextInt();
        double[][] m = new double[row][column];

        System.out.println("Enter the array below: ");
        for (int i = 0; i < row; i++)
            for (int k = 0; k < column; k++)
                m[i][k] = input.nextDouble();

        int[] location = locateLargest(m);
        System.out.printf("The location of the largest element is at (%d, %d)\n", location[0], location[1]);
    }

    public static int[] locateLargest(double[][] a) {

        int[] location = new int[] { 0, 0 };
        double largest = a[0][0];
        for (int i = 0; i < a.length; i++) {

            for (int k = 0; k < a[i].length; k++) {
                if (a[i][k] > largest) {
                    location[0] = i;
                    location[1] = k;
                    largest = a[i][k];
                }
            }
        }
    }
}
```

Java Tab Win

sortlist.java (~/) - gedit

```
Open ▾ +
```

CountPandN.java x High.java x Tpl.java x Ecal.java x sortlist.java

```
import java.util.Scanner;
//Q6

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

        System.out.print("Enter list1: ");
        int[] list1 = new int[input.nextInt()];
        for (int i = 0; i < list1.length; i++) {
            list1[i] = input.nextInt();
        }

        System.out.print("Enter list2: ");
        int[] list2 = new int[input.nextInt()];
        for (int i = 0; i < list2.length; i++) {
            list2[i] = input.nextInt();
        }

        int[] list3 = merge(list1, list2);

        System.out.print("The merged list is");
        for (int e: list3) {
            System.out.print(" " + e);
        }
        System.out.println();
    }
}
```

```
        System.out.println();
    }

public static int[] merge(int[] list1, int[] list2) {
    int[] list3 = new int[list1.length + list2.length];

    for (int i = 0; i < list1.length; i++)
        list3[i] = list1[i];
    for (int i = 0, j = list1.length; i < list2.length; i++, j++) {
        list3[j] = list2[i];
    }
    sort(list3);

    return list3;
}
public static void sort(int[] list) {
    for (int i = 0; i < list.length - 1; i++) {
        int min = list[i];
        int minIndex = i;

        for (int j = i + 1; j < list.length; j++) {
            if (list[j] < min) {
                min = list[j];
                minIndex = j;
            }
        }

        if (minIndex != i) {
            list[minIndex] = list[i];
            list[i] = min;
        }
    }
}
```

```
Open ▾ +
```

CountPandN.java x High.java x Tpl.java x Ecal.java x

```
public class time {
//Q10.
    public static void main(String[] args) {
        Time t = new Time(555550000);
        Time time2 = new Time();
        System.out.printf("%d:%d:%d%n%n",
            time2.getHour(),
            time2.getMinute(),
            time2.getSecond());

        System.out.printf("%d:%d:%d%n%n",
            time.getHour(),
            time.getMinute(),
            time.getSecond());

        time2.setTime(2432423534L);
        System.out.printf("%d:%d:%d%n%n",
            time2.getHour(),
            time2.getMinute(),
            time2.getSecond());
    }
}

class Time {
```

```
CountPandN.java x      High.java x      Tpl.java x      Ecal.java x      sortlist.java x
class Time {
    private int mHour;
    private int mMinute;
    private int mSecond;
    private long mTime;

    public Time() {
        mTime = System.currentTimeMillis();
    }

    public Time(long time) {
        mTime = time;
    }
    public Time(int hour, int minute, int second) {
        mHour = hour;
        mMinute = minute;
        mSecond = second;
    }

    public void setTime(long elapsedTime) {
        mTime = elapsedTime;
    }

    public int getHour() {
        return (int)(mTime / (1000 * 60 * 60)) % 24;
    }

    public int getMinute() {
        return (int)(mTime / (1000 * 60)) % 60;
    }

    public int getSecond() {
        return (int)(mTime / 1000) % 60;
    }
}
```