

# **GUIA DE N° 4 EJERCICIOS**

**PROFESORA:**

**LUISINA DE PAULA**

**ALUMNO:**

**MARTINEZ IVAN ALEJANDRO**



## Ejercicio N° 1:

# Vectores

Resolver los siguientes ejercicios en Lenguaje de Programación Java utilizando el IDE Netbeans.  
a) Llevar a cabo un programa que incluya un vector que almacene los siguientes nombres: Alejandra, Leonardo, Rosa, Guillermo, Gabriel, Daniel, Luisa y Ludmila. Realizar un recorrido del vector para cargar los datos y otro recorrido para mostrar los mismos por pantalla.

(SHIFT + f6)

Para ejecutar el archivo individual

```
package EjercicioN4;

public class EjercicioN4_ej1a {
    public static void main(String[] args) {
        System.out.println("");
        System.out.println("Ejercicio nº 1a - GUIA Ejercicios N4");
        System.out.println("*****");
        String nombres[] = { "Alejandra", "Leonardo", "Rosa", "Guillermo", "Gabriel", "Daniel", "Luisa",
        "Ludmila" };
        for (int i = 0; i < nombres.length; i++) {
            System.out.print((i + 1) + " " + nombres[i] + ", ");
        }
    }
}
```

The screenshot shows the NetBeans IDE interface. The top menu bar includes 'Start Page', 'File', 'Edit', 'Tools', 'View', 'Help', and a 'File' dropdown. Below the menu is a toolbar with various icons for file operations like new, open, save, cut, copy, paste, and search. The main area is the code editor, which displays the Java code for 'EjercicioN4\_ej1a'. The code uses System.out.println statements to print a header, a separator line, an array of names, and then prints each name with its index. The code editor has syntax highlighting and line numbers. To the right of the code editor is a vertical scroll bar. At the bottom of the screen is the output window, titled 'Output - EjercicioN4 (run)'. It shows the command 'run:' followed by the program's output: 'Ejercicio nº 1a - GUIA Ejercicios N4', a separator line, and the names '1 Alejandra, 2 Leonardo, 3 Rosa, 4 Guillermo, 5 Gabriel, 6 Daniel, 7 Luisa, 8 Ludmila'. Below this, it says 'BUILD SUCCESSFUL (total time: 0 seconds)'.



b) Realizar un programa que permita la carga por teclado de 10 números en un vector. Una vez cargados, se necesita que el programa determine cuál es el mayor y cuál es el menor de ellos.

```
package EjercicioN4;

import java.util.Scanner;

public class EjercicioN4_ej1b {
    public static void main(String[] args) {
        Scanner ingres = new Scanner(System.in);

        System.out.println("");
        System.out.println("");
        System.out.println("/*Ejercicio nº 1b - GUIA Ejercicios N4*");
        System.out.println("*****");
        int numeros[] = new int[10];

        for (int i = 0; i < numeros.length; i++) {
            System.out.println("Ingrese un numero que desea guardar en la posición " + (i + 1) + ":");
            numeros[i] = ingres.nextInt();
        }

        int max = numeros[0], min = numeros[0];
        for (int j = 0; j < numeros.length; j++) {
            if (min > numeros[j]) {
                min = numeros[j];
            }
            if (max < numeros[j]) {
                max = numeros[j];
            }
        }
        System.out.println("El número máximo, ingresado es: " + max + "\nEl numero mínimo es: " + min);
        ingres.close();
    }
}
```

**Output - EjercicioN4 (run)**

	run:  *Ejercicio nº 1b - GUIA Ejercicios N4* ***** Ingrese un numero que desea guardar en la posición 1: 315 Ingrese un numero que desea guardar en la posición 2: 165 Ingrese un numero que desea guardar en la posición 3: 021 Ingrese un numero que desea guardar en la posición 4: 105 Ingrese un numero que desea guardar en la posición 5: 999 Ingrese un numero que desea guardar en la posición 6: 106 Ingrese un numero que desea guardar en la posición 7: 105 Ingrese un numero que desea guardar en la posición 8: 648 Ingrese un numero que desea guardar en la posición 9: 1 Ingrese un numero que desea guardar en la posición 10: 3 El número máximo, ingresado es: 999 El numero mínimo es: 1 BUILD SUCCESSFUL (total time: 32 seconds)
--	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

c) Realizar un programa que permita la carga por teclado de 15 números en un vector. Una vez cargados, se necesita que el programa cuente e informe por pantalla cuántas veces se cargó el número 3 (si es que fue cargado alguna vez).

```
package EjercicioN4;

import java.util.Scanner;

public class EjercicioN4_ej1c {
    public static void main(String[] args) {
        Scanner ingres = new Scanner(System.in);
        System.out.println("");
        System.out.println("");
        System.out.println("*Ejercicio nº 1c - GUIA Ejercicios N4*");
        System.out.println("*****");

        int count3 = 0, j = 0, numeros15[] = new int[15];
        do {
            System.out.println("Ingrese un valor que desea guardar en la posición " + (j + 1) + ":");
            numeros15[j] = ingres.nextInt();
            if (numeros15[j] == 3) {
                count3 = count3 + 1;
            }
            j++;
        } while (j < numeros15.length);

        if (count3 == 0) {
            System.out.println("No se ha ingresado el numero 3.");
        } else {
            System.out.println("Hemos contado que ingresó " + count3 + " veces el numero 3.");
        }
        ingres.close();
    }
}
```

```
Output - EjercicioN4 (run) ×
run:
*Ejercicio nº 1c - GUIA Ejercicios N4*
*****
Ingrese un valor que desea guardar en la posición 1:
3150
Ingrese un valor que desea guardar en la posición 2:
3
Ingrese un valor que desea guardar en la posición 3:
3
Ingrese un valor que desea guardar en la posición 4:
1
Ingrese un valor que desea guardar en la posición 5:
2
Ingrese un valor que desea guardar en la posición 6:
4
Ingrese un valor que desea guardar en la posición 7:
5
Ingrese un valor que desea guardar en la posición 8:
6
Ingrese un valor que desea guardar en la posición 9:
8
Ingrese un valor que desea guardar en la posición 10:
3
Ingrese un valor que desea guardar en la posición 11:
3
Ingrese un valor que desea guardar en la posición 12:
3
Ingrese un valor que desea guardar en la posición 13:
3
Ingrese un valor que desea guardar en la posición 14:
3
Ingrese un valor que desea guardar en la posición 15:
3
Hemos contado que ingresó 8 veces el numero 3.
BUILD SUCCESSFUL (total time: 29 seconds)
```



d) Un empleado desea almacenar en un vector sus 12 sueldos del año. A partir de esa carga, necesita un programa que determine y muestre por pantalla la suma total de sus 12 sueldos y el promedio de sueldo que tuvo en el año.

```
package EjercicioN4;

import java.util.Scanner;

public class EjercicioN4_ej1d {
    public static void main(String[] args) {
        Scanner ingres = new Scanner(System.in);
        System.out.println("");
        System.out.println("");
        System.out.println("*Ejercicio nº 1d - GUIA Ejercicios N4*");
        System.out.println("*****");
        String meses[] = { "Enero", "Febrero", "Marzo", "Abril", "Mayo", "Junio", "Julio", "Agosto", "Septiembre",
                           "Octubre", "Noviembre", "Diciembre" };
        int i = 0;
        double suma = 0, sueldos[] = new double[12];

        do {
            System.out.println("Ingrese el sueldo correspondiente al mes de " + meses[i] + ":");
            sueldos[i] = ingres.nextDouble();
            suma = suma + sueldos[i];
            i++;
        } while (i < meses.length);
        ingres.close();
        System.out.println("La suma de su sueldo anual es: " + suma + " y el sueldo promedio es: " + (suma / 12));
    }
}
```

Output - EjercicioN4 (run) X

run:

\*Ejercicio nº 1d - GUIA Ejercicios N4\*

\*\*\*\*\*

Ingrese el sueldo correspondiente al mes de Enero:  
25000

Ingrese el sueldo correspondiente al mes de Febrero:  
30000

Ingrese el sueldo correspondiente al mes de Marzo:  
31000

Ingrese el sueldo correspondiente al mes de Abril:  
31000

Ingrese el sueldo correspondiente al mes de Mayo:  
35000

Ingrese el sueldo correspondiente al mes de Junio:  
35000

Ingrese el sueldo correspondiente al mes de Julio:  
36000

Ingrese el sueldo correspondiente al mes de Agosto:  
36000

Ingrese el sueldo correspondiente al mes de Septiembre:  
37000

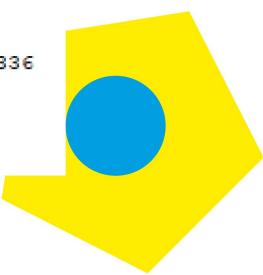
Ingrese el sueldo correspondiente al mes de Octubre:  
38000

Ingrese el sueldo correspondiente al mes de Noviembre:  
38000

Ingrese el sueldo correspondiente al mes de Diciembre:  
40000

La suma de su sueldo anual es: 412000.0 y el sueldo promedio es: 34333.333333333336

BUILD SUCCESSFUL (total time: 32 seconds)



e) En tres vectores diferentes se guardan los datos de temperaturas máximas y mínimas de 5 ciudades de Misiones. En el primer vector se guardan los nombres de las ciudades, en el segundo las temperaturas mínimas alcanzadas y en el tercero las temperaturas máximas alcanzadas en la última semana. Se necesita un programa que permita la carga de las ciudades, sus temperaturas mínimas y máximas; además, deberá poder informar por pantalla cual fue la ciudad con la temperatura más baja y cual con la temperatura más alta (dando a conocer al mismo tiempo la cantidad de grados).

```
package EjercicioN4;

import java.util.Scanner;

public class EjercicioN4_ej1e {
    public static void main(String args[]) {
        Scanner ingres = new Scanner(System.in);

        String ciudMnes[] = new String[5];

        double max = 0, min = 0;
        double tempMin[] = new double[5];
        double tempMax[] = new double[5];

        System.out.println("");
        System.out.println("");
        System.out.println("*Ejercicio nº 1e - GUIA Ejercicios N4*");
        System.out.println("*****");

        int i , pmin = 0, pmax = 0;

        for (i = 0; i < ciudMnes.length; i++) {
            System.out.print("Indique la ciudad: ");
            ciudMnes[i] = ingres.nextLine();
            ingres.reset();
        }

        i = 0;
        do{
            System.out.print("Indique Temperatura Máxima en " + ciudMnes[i] + ": ");
            tempMax[i] = ingres.nextDouble();

            if (tempMax[i] > max) {
                max = tempMax[i];
                pmax = i;
            }
            ingres.reset();
            System.out.print("Indique Temperatura Mínima: ");
            tempMin[i] = ingres.nextDouble();

            if (tempMin[i] < min) {
                min = tempMin[i];
                pmin = i;
            }
            i++;
        }while(i < ciudMnes.length);

        ingres.close();

        System.out.println("CIUDAD | MAX | MIN");
        for (int j = 0; j < ciudMnes.length; j++) {
            System.out.println(ciudMnes[j] + " " + tempMax[j] + " " + tempMin[j]);
        }
        System.out.println("La ciudad con temperatura mínima es " + ciudMnes[pmin] + " " + tempMin[pmin]
            + "\nLa ciudad con temperatura máxima es " + ciudMnes[pmax] + " " + tempMax[pmax]);
    }
}
```

Output - EjercicioN4 (run) X

run:

\*Ejercicio n° 4 - GUIA Ejercicios N4\*

\*\*\*\*\*

Indique la ciudad: Posadas

Indique la ciudad: Montecarlos

Indique la ciudad: Obera

Indique la ciudad: Iguaz{u

Indique la ciudad: Alem

Indique Temperatura MÃ;xima en Posadas: 59

Indique Temperatura MÃ;nima: 10

Indique Temperatura MÃ;xima en Montecarlos: 30

Indique Temperatura MÃ;nima: -5

Indique Temperatura MÃ;xima en Obera: 27

Indique Temperatura MÃ;nima: 15

Indique Temperatura MÃ;xima en Iguaz{u: 40

Indique Temperatura MÃ;nima: -6

Indique Temperatura MÃ;xima en Alem: 30

Indique Temperatura MÃ;nima: 20

CIUDAD | MAX | MIN

CIUDAD	MAX	MIN
Posadas	59.0	10.0
Montecarlos	30.0	-5.0
Obera	27.0	15.0
Iguaz{u	40.0	-6.0
Alem	30.0	20.0

La ciudad con temperatura mÃ;nima es Iguaz{u -6.0

La ciudad con temperatura mÃ;xima es Posadas 59.0

BUILD SUCCESSFUL (total time: 1 minute 28 seconds)



# Matrices

Resolver los siguientes ejercicios en Lenguaje de Programación Java utilizando el IDE Netbeans.  
a) Llevar a cabo un programa que permite cargar completamente con números 5 una matriz de 4x5 (4 filas, 5 columnas).



b) Llevar a cabo un programa que permita cargar mediante teclado una matriz de 3 x 3 (3 filas, 3 columnas) con razas de perros. Una vez cargada la misma, será necesario realizar un recorrido por la misma y mostrar los datos cargados por pantalla.

```
package EjercicioN4;

import java.util.Scanner;

public class EjercicioN4_ej2b {
    public static void main(String[] args) {
        Scanner ingredt = new Scanner(System.in);

        String razasCan[][] = new String[3][3];

        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {
                System.out.println("Ingrese una raza canina: (Colum " + (i + 1) + " Fila " + (j + 1) + ")/9");
                razasCan[j][i] = ingredt.nextLine();
            }
        }

        ingredt.close();
        System.out.println("Columna 1 " + " Columna 2 " + " Columna 3 ");
        for (int j = 0; j < 3; j++) {
            System.out.println(razasCan[j][0] + " " + razasCan[j][1] + " " + razasCan[j][2]);
        }

    }
}
```

The screenshot shows the 'Output' window of an IDE during the execution of the 'EjercicioN4 (run)' program. The window title is 'Output - EjercicioN4 (run)'. The output text is as follows:

```
run:
Ingrese una raza canina: (Colum 1 Fila 1)/9
dog1
Ingrese una raza canina: (Colum 1 Fila 2)/9
dog2
Ingrese una raza canina: (Colum 1 Fila 3)/9
dog3
Ingrese una raza canina: (Colum 2 Fila 1)/9
dog4
Ingrese una raza canina: (Colum 2 Fila 2)/9
dog5
Ingrese una raza canina: (Colum 2 Fila 3)/9
dog6
Ingrese una raza canina: (Colum 3 Fila 1)/9
dog7
Ingrese una raza canina: (Colum 3 Fila 2)/9
dog8
Ingrese una raza canina: (Colum 3 Fila 3)/9
dog9
Columna 1   Columna 2   Columna 3
dog1   dog4   dog7
dog2   dog5   dog8
dog3   dog6   dog9
BUILD SUCCESSFUL (total time: 34 seconds)
```

c) En una tabla de 4 filas y 4 columnas se guardan las notas de 4 alumnos de secundario. Cada fila corresponde a las notas y al promedio de cada alumno.

- Se necesita un programa que permita a un profesor cargar en las 3 primeras posiciones de cada fila las notas del alumno y que en la última columna se calculen los promedios.
- Una vez realizados los cálculos, se desea mostrar las 3 notas de cada alumno y el promedio correspondiente recorriendo la matriz.

```
package EjercicioN4;

import java.util.Scanner;

public class EjercicioN4_ej2c {
    public static void main(String[] args) {
        Scanner ingresoDt = new Scanner(System.in);

        double notas[][] = new double[4][4];

        for (int i = 0; i < 4; i++) {
            for (int j = 0; j < 3; j++) {
                System.out.println("Ingrese la nota" + (j+1) + " del Alumno" + (i+1));
                notas[i][j] = ingresoDt.nextDouble();
            }
            notas[i][3] = (notas[i][0]+notas[i][1]+notas[i][2])/(notas.length-1);
        }

        ingresoDt.close();

        System.out.println("Alumno Nota1 Nota2 Nota3 Promedio");
        for (int i = 0; i < notas.length; i++) {
            System.out.println((i+1) + " " + notas[i][0] + " " + notas[i][1] + " " + notas[i][2] + " " + notas[i][3]);
        }
    }
}
```

Alumno	Nota1	Nota2	Nota3	Promedio
1	6.0	7.0	9.0	7.333333333333333
2	10.0	3.0	6.0	6.333333333333333
3	8.0	8.0	9.0	8.333333333333334
4	4.0	6.0	8.0	6.0

BUILD SUCCESSFUL (total time: 25 seconds)

MUCHAS  
GRACIAS

