



PRACTICA 8

Facultad de ciencias de la computación
Ing. En ciencias de la computación

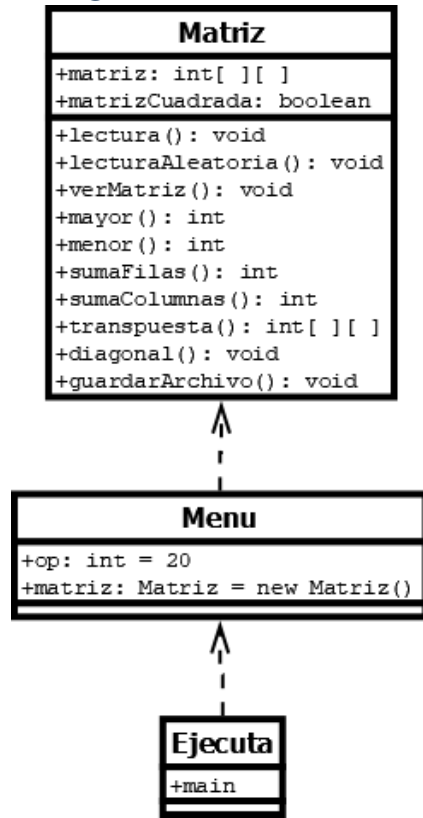


IRVYN XICALE CABRERA
201963582

Introducción

Esta practica nos ayuda a comprender el funcionamiento y creación de una matriz en el lenguaje java

Diagramas ULM



Código

Matriz.java

```
import java.io.*;
public class Matriz {
    // lectura
    InputStreamReader entrada = new InputStreamReader(System.in);
    BufferedReader flujoEntrada = new BufferedReader(entrada);
    String datoEntrada;

    // atributos
    int[][] matriz;
    boolean matrizCuadrada;

    // metodos
    public void lectura(){
        boolean validacion = false;
        int fila=0,columna=0;
        do{
            try{
                System.out.print("Ingresa el numero de filas de la matriz:");
                datoEntrada = flujoEntrada.readLine();
                fila=Integer.parseInt(datoEntrada);
                System.out.print("Ingresa el numero de columnas de la matriz:");
                datoEntrada = flujoEntrada.readLine();
                columna=Integer.parseInt(datoEntrada);
                if(fila==columna){
                    matrizCuadrada=true;
                }else
            }
        }
    }
}
```

```

        matrizCuadrada=false;
        matriz= new int[fila][columna];
        // si la lectura lee todo bien validacion cambia a verdadero
        validacion = true;
        // si los datos obtenidos son menores a 0 la validacion cambia a false
        if(fila<1 && columna<1){
            validacion=false;
            System.out.println("la matriz no permite numeros menores que 0");
        }
    }catch(IOException error){
        System.err.println("Error " + error.getMessage());
    }
}while(validacion == false);
try{
    for(int i=0; i<fila; i++){
        for(int j=0; j<columna; j++){
            System.out.print("ingresa el dato [" + (i+1) + "] [" + (j+1) + "]:");
            datoEntrada = flujoEntrada.readLine();
            matriz[i][j] = Integer.parseInt(datoEntrada);
        }
    }
}catch(IOException error){
    System.err.println("Error " + error.getMessage());
}
System.out.println("");
}

public void lecturaAleatoria(){
    int fila, columna;
    boolean validacion = false;

```

```

do{
    try{
        System.out.print("Ingresa el numero de filas de la matriz:");
        datoEntrada = flujoEntrada.readLine();
        fila = Integer.parseInt(datoEntrada);
        System.out.print("Ingresa el numero de columnas de la matriz:");
        datoEntrada = flujoEntrada.readLine();
        columna = Integer.parseInt(datoEntrada);
        matriz = new int[fila][columna];
        for(int i=0; i<fila; i++){
            for(int j=0; j<columna; j++){
                matriz[i][j] = (int)(Math.random()*100+1);
            }
        }
        if(fila == columna){
            matrizCuadrada=true;
        }else{
            matrizCuadrada = false;
            validacion=true;
        }catch(IOException error){
            System.err.println("Error " + error.getMessage());
        }
    }while(validacion==false);
    System.out.println("Matriz generada\n");
}

public void verMatriz(){
    for(int i=0;i<matriz.length;i++){
        for(int j=0;j<matriz[i].length;j++){
            System.out.print(matriz[i][j] + "\t");

```

```

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

public int mayor(){
    int mayor=matriz[0][0];
    for(int i=0;i<matriz.length;i++){
        for(int j=0;j<matriz[i].length;j++){
            if(mayor<matriz[i][j]){
                mayor=matriz[i][j];
            }
        }
    }
    return mayor;
}

public int menor(){
    int menor=matriz[0][0];
    for(int i=0;i<matriz.length;i++){
        for(int j=0;j<matriz[i].length;j++){
            if(menor>matriz[i][j]){
                menor=matriz[i][j];
            }
        }
    }
    return menor;
}

```

```

public int sumaFilas(boolean imprimir){
    int suma=0,sumaTotal=0;
    for(int i=0;i<matriz.length;i++){
        for(int j=0;j<matriz[i].length;j++){
            suma += matriz[i][j];
        }
        if(imprimir == true){
            System.out.println("la suma de la fila " + (i+1) + " es: " + suma);
        }
        sumaTotal += suma;
        suma = 0;
    }
    return sumaTotal;
}

public int sumaColumnas(boolean imprimir){
    int suma=0,sumaTotal=0;
    for(int j=0;j<matriz[0].length;j++){
        for(int i=0;i<matriz.length;i++){
            suma += matriz[i][j];
        }
        if(imprimir == true){
            System.out.println("la suma de la columna " + (j+1) + " es: " + suma);
        }
        sumaTotal += suma;
        suma = 0;
    }
    return sumaTotal;
}

```

```

public int[][] transpuesta(boolean imprimir){
    int[][] transpuesta= new int[matriz[0].length][matriz.length];
    for(int j=0;j<matriz[0].length;j++){
        for(int i=0;i<matriz.length;i++){
            transpuesta[j][i] = matriz[i][j];
        }
    }
    // imprime en consola
    if(imprimir == true){
        System.out.println("\tMATRIZ ORIGINAL");
        for(int i=0;i<matriz.length;i++){
            for(int j=0;j<matriz[i].length;j++){
                System.out.print(matriz[i][j] + "\t");
            }
            System.out.println("");
        }
        System.out.println("");
        System.out.println("*****");
        System.out.println("");
        System.out.println("MATRIZ TRANSPUESTA");
        for(int i=0;i<transpuesta.length;i++){
            for(int j=0;j<transpuesta[i].length;j++){
                System.out.print(transpuesta[i][j] + "\t");
            }
            System.out.println("");
        }
    }
    return transpuesta;
}

```



```

public void diagonal(){
    for(int i=0;i<matriz.length;i++){
        for(int j=0;j<matriz[i].length;j++){
            if (i==j){
                System.out.println("posicion [" + (i+1) + "][" + (j+1) + "]: " + matriz[i][j]);
            }
        }
    }
}

```

```

public void guardarArchivo(int[][] transpuesta){
    FileWriter fichero = null;
    PrintWriter inicio = null;
    try {
        fichero = new FileWriter("matriz.txt");
        inicio = new PrintWriter(fichero);

        inicio.println("\tMATRIZ ORIGINAL");
        for(int i=0;i<matriz.length;i++){
            for(int j=0;j<matriz[i].length;j++){
                inicio.print(matriz[i][j] + "\t");
            }
            inicio.println("");
        }
        inicio.println("");
        inicio.println("*****");
        inicio.println("");
        inicio.println("MATRIZ TRANSPUESTA");
        for(int i=0;i<transpuesta.length;i++){
            for(int j=0;j<transpuesta[i].length;j++){

```

```

        inicio.print(transpuesta[i][j] + "\t");
    }
    inicio.println("");
}
} catch (Exception e) {
    e.printStackTrace();
} finally {
    try {
        // Nuevamente aprovechamos el finally para
        // asegurarnos que se cierra el fichero.
        if (null != fichero)
            fichero.close();
    } catch (Exception e2) {
        e2.printStackTrace();
    }
}
}
}
}

```

Menu.java

```

import java.io.*;
public class Menu {
    // lectura
    InputStreamReader entrada = new InputStreamReader(System.in);
    BufferedReader flujoEntrada = new BufferedReader(entrada);
    String datoEntrada;

    // atributos

```

```

int op=20;
Matriz matriz = new Matriz();

// metodos
public void menu(){
    do{
        try{
            System.out.println("\tMenu\n0.Salir\n1.Leer los elementos de una Matriz\n2. Generar aleatoria-
mente los números en la matriz\n3. Visualizar elementos de la matriz\n4. Obtener el elemento mayor\n5. Obte-
ner el elemento menor\n6. Sumar todas las filas de la matriz\n7. Sumar todas las columnas de la matriz\n8. Su-
mar toda la matriz\n9. Obtener la transpuesta de una matriz\n10. Obtener la diagonal de una matriz\n11. Guar-
dar en un archivo");
            datoEntrada = flujoEntrada.readLine();
            op = Integer.parseInt(datoEntrada);
            switch(op){
                case 0:
                    System.out.println("Gracias por usar este programa :)");
                    break;
                case 1:
                    System.out.println("");
                    matriz.lectura();
                    System.out.println("");
                    break;
                case 2:
                    System.out.println("");
                    matriz.lecturaAleatoria();
                    System.out.println("");
                    break;
                case 3:
                    System.out.println("");

```

```
matriz.verMatriz();
System.out.println("");
break;
case 4:
System.out.println("");
System.out.println("el numero mayor es: " + matriz.mayor());
System.out.println("");
break;
case 5:
System.out.println("");
System.out.println("el numero menor es: " + matriz.menor());
System.out.println("");
break;
case 6:
System.out.println("");
matriz.sumaFilas(true);
System.out.println("");
break;
case 7:
System.out.println("");
matriz.sumaColumnas(true);
System.out.println("");
break;
case 8:
System.out.println("");
System.out.println("la suma de toda la matriz es " + matriz.sumaFilas(false));
System.out.println("");
break;
case 9:
System.out.println("");
matriz.transpuesta(true);
```

```

        System.out.println("");
        break;
        case 10:
            System.out.println("");
            matriz.diagonal();
            System.out.println("");
            break;
        case 11:
            System.out.println("");
            matriz.guardarArchivo(matriz.transpuesta(false));
            System.out.println("");
            break;
        default:
            System.out.println("Opcion invalida");
            op = 20;
    }
} catch(IOException error)
{
    System.err.println("Error " + error.getMessage());
}
}while(op!=0);
}
}

```

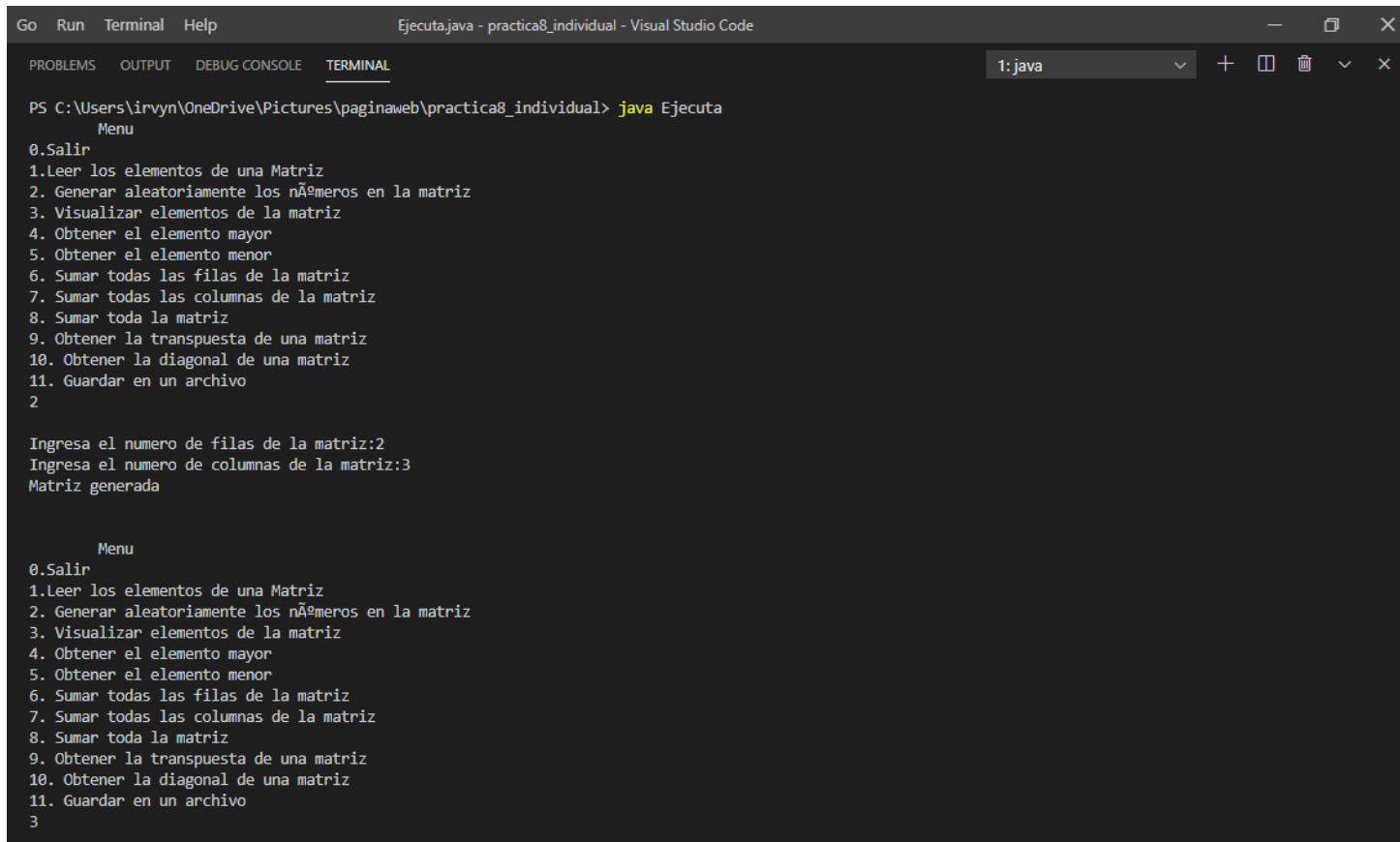
Ejecuta.java

```

public class Ejecuta {
    public static void main(String[] args){
        Menu menu = new Menu();
        menu.menu();
    }
}

```

```
}  
}
```



```
Go Run Terminal Help Ejecuta.java - practica8_individual - Visual Studio Code  
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1:java  
PS C:\Users\irvyn\OneDrive\Pictures\paginaweb\practica8_individual> java Ejecuta  
Menu  
0.Salir  
1.Leer los elementos de una Matriz  
2. Generar aleatoriamente los números en la matriz  
3. Visualizar elementos de la matriz  
4. Obtener el elemento mayor  
5. Obtener el elemento menor  
6. Sumar todas las filas de la matriz  
7. Sumar todas las columnas de la matriz  
8. Sumar toda la matriz  
9. Obtener la transpuesta de una matriz  
10. Obtener la diagonal de una matriz  
11. Guardar en un archivo  
2  
  
Ingresa el numero de filas de la matriz:2  
Ingresa el numero de columnas de la matriz:3  
Matriz generada  
  
Menu  
0.Salir  
1.Leer los elementos de una Matriz  
2. Generar aleatoriamente los números en la matriz  
3. Visualizar elementos de la matriz  
4. Obtener el elemento mayor  
5. Obtener el elemento menor  
6. Sumar todas las filas de la matriz  
7. Sumar todas las columnas de la matriz  
8. Sumar toda la matriz  
9. Obtener la transpuesta de una matriz  
10. Obtener la diagonal de una matriz  
11. Guardar en un archivo  
3
```

```
Go Run Terminal Help Ejecuta.java - practica8_individual - Visual Studio Code

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: java + [ ] [ ] [ ] [ ] [ ] [ ]

11. Guardar en un archivo
3

19      55      95
13      19      27

Menu

0.Salir
1.Leer los elementos de una Matriz
2. Generar aleatoriamente los números en la matriz
3. Visualizar elementos de la matriz
4. Obtener el elemento mayor
5. Obtener el elemento menor
6. Sumar todas las filas de la matriz
7. Sumar todas las columnas de la matriz
8. Sumar toda la matriz
9. Obtener la transpuesta de una matriz
10. Obtener la diagonal de una matriz
11. Guardar en un archivo
4

el numero mayor es: 95

Menu

0.Salir
1.Leer los elementos de una Matriz
2. Generar aleatoriamente los números en la matriz
3. Visualizar elementos de la matriz
4. Obtener el elemento mayor
5. Obtener el elemento menor
6. Sumar todas las filas de la matriz
7. Sumar todas las columnas de la matriz
8. Sumar toda la matriz
9. Obtener la transpuesta de una matriz
10. Obtener la diagonal de una matriz
11. Guardar en un archivo
5
```


```
Go Run Terminal Help Ejecuta.java - practica8_individual - Visual Studio Code

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
5
el numero menor es: 13

Menu
0.Salir
1.Leer los elementos de una Matriz
2. Generar aleatoriamente los números en la matriz
3. Visualizar elementos de la matriz
4. Obtener el elemento mayor
5. Obtener el elemento menor
6. Sumar todas las filas de la matriz
7. Sumar todas las columnas de la matriz
8. Sumar toda la matriz
9. Obtener la transpuesta de una matriz
10. Obtener la diagonal de una matriz
11. Guardar en un archivo
6

la suma de la fila 1 es: 169
la suma de la fila 2 es: 59

Menu
0.Salir
1.Leer los elementos de una Matriz
2. Generar aleatoriamente los números en la matriz
3. Visualizar elementos de la matriz
4. Obtener el elemento mayor
5. Obtener el elemento menor
6. Sumar todas las filas de la matriz
7. Sumar todas las columnas de la matriz
8. Sumar toda la matriz
9. Obtener la transpuesta de una matriz
10. Obtener la diagonal de una matriz
11. Guardar en un archivo
7
```

```
Go Run Terminal Help Ejecuta.java - practica8_individual - Visual Studio Code

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: java
7

la suma de la columna 1 es: 32
la suma de la columna 2 es: 74
la suma de la columna 3 es: 122

Menu

0.Salir
1.Leer los elementos de una Matriz
2. Generar aleatoriamente los números en la matriz
3. Visualizar elementos de la matriz
4. Obtener el elemento mayor
5. Obtener el elemento menor
6. Sumar todas las filas de la matriz
7. Sumar todas las columnas de la matriz
8. Sumar toda la matriz
9. Obtener la transpuesta de una matriz
10. Obtener la diagonal de una matriz
11. Guardar en un archivo
8

la suma de toda la matriz es 228

Menu

0.Salir
1.Leer los elementos de una Matriz
2. Generar aleatoriamente los números en la matriz
3. Visualizar elementos de la matriz
4. Obtener el elemento mayor
5. Obtener el elemento menor
6. Sumar todas las filas de la matriz
7. Sumar todas las columnas de la matriz
8. Sumar toda la matriz
9. Obtener la transpuesta de una matriz
10. Obtener la diagonal de una matriz
11. Guardar en un archivo
9
```

```
Go Run Terminal Help Ejecuta.java - practica8_individual - Visual Studio Code

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: java

9

      MATRIZ ORIGINAL
19    55    95
13    19    27

*****

MATRIZ TRANSPUESTA
19    13
55    19
95    27

      Menu
0.Salir
1.Leer los elementos de una Matriz
2. Generar aleatoriamente los números en la matriz
3. Visualizar elementos de la matriz
4. Obtener el elemento mayor
5. Obtener el elemento menor
6. Sumar todas las filas de la matriz
7. Sumar todas las columnas de la matriz
8. Sumar toda la matriz
9. Obtener la transpuesta de una matriz
10. Obtener la diagonal de una matriz
11. Guardar en un archivo
10

posicion [1][1]:19
posicion [2][2]:19
```

```
Go Run Terminal Help      matriz.txt - practica8_individual - Visual Studio Code
Ejecuta.java  matriz.txt x Menu.java  Matriz.java
matriz.txt
1  MATRIZ ORIGINAL
2  19  55  95
3  13  19  27
4
5  *****
6
7  MATRIZ  TRANSPUESTA
8  19  13
9  55  19
10 95  27
11
```

Conclusión

Esta practica nos ayudo a comprender la programación orientada a objetos

Hoja de firmas

RÚBRICA HOJA DE EVALUACIÓN DE PRÁCTICAS

MATRICULA: 201963582 FECHA: 24/01/2021

NOMBRE: Xicale Cabrera Irvyn No. PRACTICA: 1 INDIVIDUAL (x) COLABORATIVA ()

CRITERIOS	Deficiente	Suficiente	Bueno	Excelente	Calificación Obtenida	
	1-5.9	6-7.9	8-9	9.1-10	%	Puntos
CONOCIMIENTO TEÓRICO 20%	Conocimiento deficiente de los fundamentos teóricos de POO y no puede aplicarlos en el diseño de clases.	Conocimiento confuso de los fundamentos teóricos de POO y el diseño de las clases y relaciones es incompleto.	Conocimiento claro de los fundamentos teóricos POO pero requiere mejorar el modelado de las clases, sus métodos y sus relaciones entre clases.	Dominio del Conocimiento de los fundamentos teóricos POO y puede aplicarlos de forma completa en el modelado de las clases, métodos y todas las relaciones entre clase.		
EJECUCIÓN DE LA PRÁCTICA 30%	No puede realizar la práctica ya que desconoce el entorno de trabajo y desarrollo de la práctica en lenguaje UML y Java.	Realiza la práctica de forma incompleta ya que desconoce el entorno de trabajo del lenguaje UML y Java	Realiza la práctica pero requiere mejorar en el manejo del entorno de trabajo del lenguaje de programación(sintaxis y semántica)	Realiza la práctica de forma correcta y completa, demuestra dominio del entorno de trabajo del lenguaje de programación (sintaxis y semántica).		
SOLUCIÓN DE LA PRÁCTICA 40%	No puede generar las soluciones o programas a los problemas planteados ya que no posee el dominio teórico y práctico del modelado y el lenguaje de programación.	Propone soluciones confusas o programas incompletos a los problemas planteados, ya que carece del dominio del modelado y lenguaje de programación	Genera soluciones con poca profundidad y los programas no están orientados de acuerdo a los problemas solicitados, por lo cual no tiene un dominio profundo de la temática y del Lenguaje.	Genera soluciones con profundidad y los programas son correctos de acuerdo a los problemas planteados, por lo cual demuestra un dominio de la temática y del Lenguaje de Programación.		

ACTITUD DE APRENDER Y COLABORAR EN EQUIPO DE TRABAJO 10%	No posee una actitud proactiva para un aprendizaje autónomo y no le gusta participar y trabajar en equipo.	Posee una actitud propositiva para un aprendizaje autónomo, participa pero no le gusta trabajar en equipo.	Posee una actitud propositiva logrando un aprendizaje autónomo, colaborativo, le gusta trabajar en equipo pero requiere mejorar su participación y portaciones de forma profunda.	Posee una actitud proactiva logrando un aprendizaje autónomo, participa con propuestas concretas y profundas, le gusta trabajar en equipo y asume su responsabilidad dentro para lograr éxito del equipo		
Total						

Firma del Alumn@

Vo. Bo. Docente

