

LABORATORIO de COMPUTACIÓN I

Variables estáticas

STATIC:

Sirve para crear miembros que pertenecen a la clase, y no miembros locales a los métodos.

En ocasiones es necesario o conveniente generar elementos que tomen un mismo valor para cualquier número de métodos sin la necesidad de pasarlos como parámetros, para que otros métodos conozcan su existencia.

Dos aspectos característicos de utilizar el calificador static en un elemento Java son los siguientes:

- **De** una variable *static* sólo existe una instancia.
- Todos los miembros definidos *static* deben ser utilizados en una estructura *static*.

Importante: se debe considerar muy necesaria la existencia de la variable estática, ya que permanecerá en memoria **todo el tiempo que el programa se encuentre en ejecución**. Las variables locales a los métodos se crearán con la ejecución del mismo, y se prepararán para ser eliminadas cuando el método que las declaró, deje de funcionar.

Ejemplos utilizando variables y métodos static

```
public class NewClass {
    public static String texto;
    public static void main(String[] args) {
        texto = JOptionPane.showInputDialog("Escriba ...");
        longitud();
        System.out.println("En mayúscula es: "+mayus());
    }

    public static void longitud() {
        System.out.println("El texto tiene "+texto.trim().length()+" caracteres.");
    }

    public static String mayus() {
        return texto.trim().toUpperCase();
    }
}
```

```

public class NoEstaticas {
    public static void main(String[] args) {
        int[][] m = new int[5][5];
        llenarMatriz(m);
        mostrarMatriz(m);
        System.out.print("elegido-> "+m[1][2]);
        if (pares(1,2,m)) System.out.println(": Es par ");
        else System.out.println(": Es impar ");
    }

    public static void llenarMatriz(int[][] ma){ // recibe m que aquí es ma
        for (int i=0;i<ma.length;i++){
            for (int j=0;j<ma[0].length;j++){
                ma[i][j]=(int) (Math.random()*100);
            }
        }
    }

    public static void mostrarMatriz(int[][] otra){ //recibe m que aquí es otra
        for (int i=0;i<otra.length;i++){
            for (int j=0;j<otra[0].length;j++){
                System.out.print(otra[i][j]+" ");
            }
            System.out.println("");
        }
    }

    public static boolean pares(int f,int c,int[][] otramas ){ //recibe tres
        boolean es =false;
        if ((int)otramas[f][c]%2 == 0) es = true;
        return es;
    }
}

```

```

public class Estaticas {
    static int[][] m = new int[5][5];
    public static void main(String[] args) {
        llenarMatriz();
        mostrarMatriz();
        System.out.println("elegido: "+m[1][2]);
        if (pares(1,2)) System.out.println("Es par ");
    }

    public static void llenarMatriz(){ //NO recibe parámetros
        for (int i=0;i<m.length;i++){
            for (int j=0;j<m[0].length;j++){
                m[i][j]=(int) (Math.random()*100);
            }
        }
    }

    public static void mostrarMatriz(){ //NO recibe parámetros
        for (int i=0;i<m.length;i++){
            for (int j=0;j<m[0].length;j++){
                System.out.print(m[i][j]+" ");
            }
            System.out.println("");
        }
    }

    public static boolean pares(int f,int c){ //recibe parámetros
        boolean es =false;
        if ((int)m[f][c]%2 == 0) es = true;
        return es;
    }
}

```