

Apunte Actividad 3

static y toString() en acción

1. El método toString()

Ya vimos que toString() sirve para mostrar el estado interno de un objeto.

Ahora lo usaremos en ejemplos prácticos.

Ejemplo: Clase Producto

```
public class Producto {  
  
    private String nombre;  
    private double precio;  
  
    public Producto(String nombre, double precio) {  
        this.nombre = nombre;  
        this.precio = precio;  
    }  
  
    @Override  
    public String toString() {  
        return "Producto [nombre=" + nombre + ", precio=" + precio + "]";  
    }  
}  
  
Producto p1 = new Producto("Café", 1200.5);  
Producto p2 = new Producto("Azúcar", 850.0);  
  
System.out.println(p1); // Producto [nombre=Café, precio=1200.5]  
System.out.println(p2); // Producto [nombre=Azúcar, precio=850.0]
```

Ahora cada vez que imprimimos el objeto, obtenemos una descripción clara.

2. Miembros estáticos (static)

Cuando usamos static, estamos definiendo **elementos de la clase**, no de cada objeto.

2.1 Atributos estáticos

Un atributo static se comparte entre todos los objetos de la clase.

```
public class Producto {  
  
    private String nombre;  
    private double precio;  
    public static int contadorProductos = 0;  
  
    public Producto(String nombre, double precio) {  
        this.nombre = nombre;  
        this.precio = precio;  
        contadorProductos++; // cada vez que se crea un producto, aumenta  
    }  
  
    @Override  
    public String toString() {  
        return "Producto [nombre=" + nombre + ", precio=" + precio + "];"  
    }  
}  
  
public class Main {  
  
    public static void main(String[] args) {  
        Producto p1 = new Producto("Café", 1200.5);  
        Producto p2 = new Producto("Azúcar", 850.0);  
  
        System.out.println("Productos creados: " + Producto.contadorProductos);  
        // Productos creados: 2  
    }  
}
```

2.2 Métodos estáticos

Un método static se puede invocar **sin necesidad de instanciar la clase**.

```
public class Utilidad {  
  
    public static double convertirADolares(double pesos) {  
        return pesos / 1000; // ejemplo con tasa fija  
    }  
}  
  
public class Main {  
  
    public static void main(String[] args) {  
        double valor = Utilidad.convertirADolares(5000);  
        System.out.println("Equivalente en dólares: " + valor);  
    }  
}
```

Esto es muy útil en **clases de utilidad** o cuando una acción no depende de un objeto particular.