

## PROGRAMACIÓN II

### Trabajo Práctico 4: Programación Orientada a Objetos II

**Alumno:** Angelelli, Rodrigo Martin

**Comisión** N°7

**Link al repositorio:** [https://github.com/roangelelli/Programacion2\\_TUPAD\\_Angelelli\\_Rodrigo](https://github.com/roangelelli/Programacion2_TUPAD_Angelelli_Rodrigo)

#### **CASO PRACTICO:**

**Sistema de Gestión de Empleados :** Modelar una clase Empleado que represente a un trabajador en una empresa. Esta clase debe incluir constructores sobrecargados, métodos sobrecargados y el uso de atributos aplicando encapsulamiento y métodos estáticos para llevar control de los objetos creados.

#### **Clase Empleado – Atributos:**

- **int id:** Identificador único del empleado.
- **String nombre:** Nombre completo.
- **String puesto:** Cargo que desempeña.
- **double salario:** Salario actual.
- **static int totalEmpleados:** Contador global de empleados creados.

#### **Requerimientos:**

1. Uso de **this**: Utilizar **this** en los constructores para distinguir parámetros de atributos.
2. **Constructores sobrecargados:**
  - a. Uno que reciba todos los atributos como parámetros.
  - b. Otro que reciba solo nombre y puesto, asignando un id automático y un salario por defecto.
  - c. Ambos deben incrementar **totalEmpleados**.
3. **Métodos sobrecargados actualizarSalario:**
  - a. Uno que reciba un porcentaje de aumento.
  - b. Otro que reciba una cantidad fija a aumentar.
4. **Método toString():** Mostrar id, nombre, puesto y salario de forma legible.
5. **Método estático mostrarTotalEmpleados():** Retornar el total de empleados creados hasta el momento.
6. **Encapsulamiento en los atributos:**
  - a. Restringir el acceso directo a los atributos de la clase.
  - b. Crear los métodos Getters y Setters correspondientes.

### Tareas a realizar:

- Implementar la clase **Empleado** aplicando todos los puntos anteriores.
- Crear una clase de prueba con método main que:
  - Instancie varios objetos usando ambos constructores.
  - Aplique los métodos **actualizarSalario()** sobre distintos empleados.
  - Imprima la información de cada empleado con **toString()**.
  - Muestre el total de empleados creados con **mostrarTotalEmpleados()**.

### Clase Empleado:

```
1 package tp4_angelelli_rodrigo_poo;
2 public class Empleado {
3     private int id;
4     private String nombre;
5     private String puesto;
6     private double salario;
7     private static int totalEmpleados = 0;
8
9     public int getId() {...3 lines }
10    public void setId(int id) {...3 lines }
11    public String getNombre() {...3 lines }
12    public void setNombre(String nombre) {...3 lines }
13    public String getPuesto() {...3 lines }
14    public void setPuesto(String puesto) {...3 lines }
15    public double getSalario() {...3 lines }
16    public void setSalario(double salario) {...3 lines }
17    public static int getTotalEmpleados() {...3 lines }
18    public static void setTotalEmpleados(int totalEmpleados) {...3 lines }
19
20    //Constructor 1: Uno que reciba todos los atributos como parámetros.
21    public Empleado(int id, String nombre, String puesto, double salario){
22        Empleado.totalEmpleados++;
23        this.id = id;
24        this.nombre = nombre;
25        this.puesto = puesto;
26        this.salario = salario;
27    }
28
29    //Constructor 2: Otro que reciba solo nombre y puesto, asignando un id automático y un salario por defecto.
30    public Empleado(String nombre, String puesto){
31        Empleado.totalEmpleados++;
32        this.id = Empleado.totalEmpleados;
33        this.nombre = nombre;
34        this.puesto = puesto;
35        this.salario = 1000000;
36    }
37
38    //Metodo 1: Uno que reciba un porcentaje de aumento
39    public void actualizarSalario(int aumento){
40        this.salario += ((salario * aumento)/100);
41    }
42
43    //Metodo 2: Otro que reciba una cantidad fija a aumentar.
44    public void actualizarSalario(double aumento){
45        this.salario += aumento;
46    }
47 }
```

```

67
68 // Sobreescribimos el metodo toString() para mostrar la informacion del empleado
69
70 @Override
71 public String toString() {
72     return "Empleado{" + "id=" + id + ", nombre=" + nombre + ", puesto=" + puesto + ", salario=" + salario + '}';
73 }
74
75 public static int mostrarTotalEmpleados(){
76     return Empleado.totalEmpleados;
77 }
78
79 }

```

## Main :

```

1 package tp4_angelelli_rodrigo_poo;
2 public class TP4_ANGELELLI_RODRIGO_P00 {
3
4     public static void main(String[] args) {
5
6         Empleado emp1 = new Empleado(01, "Carlos", "Cajero", 1500000.00);
7         Empleado emp2 = new Empleado(02, "Roxana", "Encargada", 1600000.00);
8         Empleado emp3 = new Empleado("Marcela", "Cajera");
9         Empleado emp4 = new Empleado("Fabian", "Recepcionista");
10
11         System.out.println("----- DATOS INICIALES -----");
12         System.out.println(emp1.toString());
13         System.out.println(emp2.toString());
14         System.out.println(emp3.toString());
15         System.out.println(emp4.toString());
16         System.out.println("----- FIN DATOS INICIALES -----");
17
18         //Actualizamos los salarios
19
20         emp1.actualizarSalario(15); //Porcentaje
21         emp2.actualizarSalario(120000.00); //Monto fijo
22         emp3.actualizarSalario(20); //Porcentaje
23         emp4.actualizarSalario(250000.00); //Monto fijo
24
25         //Visualizamos si se actualizaron
26         System.out.println("----- DATOS ACTUALIZADOS -----");
27         System.out.println(emp1.toString());
28         System.out.println(emp2.toString());
29         System.out.println(emp3.toString());
30         System.out.println(emp4.toString());
31         System.out.println("----- FIN DATOS ACTUALIZADOS -----");
32
33         //Total de empleados
34         System.out.println("Total empleados:" + Empleado.mostrarTotalEmpleados());
35     }
36 }
37
38 }

```

## Consola:

```

run:
----- DATOS INICIALES -----
Empleado{id=1, nombre=Carlos, puesto=Cajero, salario=1500000.0}
Empleado{id=2, nombre=Roxana, puesto=Encargada, salario=1600000.0}
Empleado{id=3, nombre=Marcela, puesto=Cajera, salario=1000000.0}
Empleado{id=4, nombre=Fabian, puesto=Recepcionista, salario=1000000.0}
----- FIN DATOS INICIALES -----
----- DATOS ACTUALIZADOS -----
Empleado{id=1, nombre=Carlos, puesto=Cajero, salario=1725000.0}
Empleado{id=2, nombre=Roxana, puesto=Encargada, salario=1720000.0}
Empleado{id=3, nombre=Marcela, puesto=Cajera, salario=1200000.0}
Empleado{id=4, nombre=Fabian, puesto=Recepcionista, salario=1250000.0}
----- FIN DATOS ACTUALIZADOS -----
Total empleados:4
BUILD SUCCESSFUL (total time: 0 seconds)

```