

PROGRAMACIÓN II

Trabajo Práctico 4: Programación Orientada a Objetos II

Alumno: Matias Luna

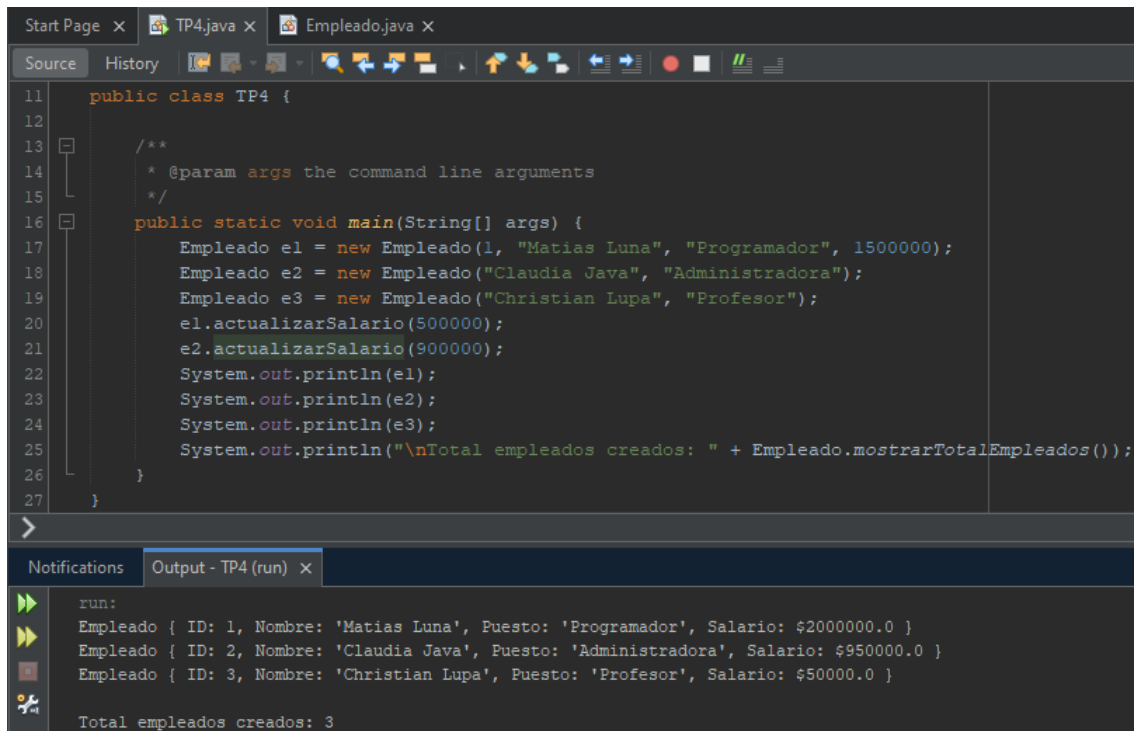
GitHub: <https://github.com/matiaslunaa?tab=repositories>

1. Implementar la clase Empleado aplicando todos los puntos anteriores.

```
Start Page x TP4.java x Empleado.java x
Source History
10  */
11  public class Empleado {
12      private int id;
13      private String nombre;
14      private String puesto;
15      private double salario;
16      private static int totalEmpleados = 0;
17
18      public Empleado(int id, String nombre, String puesto, double salario) {
19          this.id = id;
20          this.nombre = nombre;
21          this.puesto = puesto;
22          this.salario = salario;
23          totalEmpleados++;
24      }
25
26      public Empleado(String nombre, String puesto) {
27          this.id = ++totalEmpleados;
28          this.nombre = nombre;
29          this.puesto = puesto;
30          this.salario = 50000;
31      }
32
33      public void actualizarSalario(double porcentaje) {
34          this.salario += this.salario * (porcentaje / 100);
35      }
36      public void actualizarSalario(int cantidad) {
37          this.salario += cantidad;
38      }
39
40      public int getId() {
41          return id;
42      }
43      public void setNombre(String nombre) {
44          this.nombre = nombre;
45      }
46      public String getNombre() {
47          return nombre;
48      }
49      public void setPuesto(String puesto) {
50          this.puesto = puesto;
51      }
52      public String getPuesto() {
53          return puesto;
54      }
55      public void setSalario(double salario) {
56          this.salario = salario;
57      }
58      public double getSalario() {
59          return salario;
60      }
61      @Override
62      public String toString() {
63          return "Empleado { " +
64              "ID: " + id +
65              ", Nombre: '" + nombre + '\'' +
66              ", Puesto: '" + puesto + '\'' +
67              ", Salario: $" + salario +
68              " }";
69      }
70
71      public static int mostrarTotalEmpleados() {
72          return totalEmpleados;
73      }
74  }
```

2. 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().



The screenshot shows an IDE with two tabs: TP4.java and Empleado.java. The TP4.java tab is active, displaying the following code:

```
11 public class TP4 {
12
13     /**
14      * @param args the command line arguments
15      */
16     public static void main(String[] args) {
17         Empleado e1 = new Empleado(1, "Matias Luna", "Programador", 1500000);
18         Empleado e2 = new Empleado("Claudia Java", "Administradora");
19         Empleado e3 = new Empleado("Christian Lupa", "Profesor");
20         e1.actualizarSalario(500000);
21         e2.actualizarSalario(900000);
22         System.out.println(e1);
23         System.out.println(e2);
24         System.out.println(e3);
25         System.out.println("\nTotal empleados creados: " + Empleado.mostrarTotalEmpleados());
26     }
27 }
```

The output window shows the following results:

```
run:
Empleado { ID: 1, Nombre: 'Matias Luna', Puesto: 'Programador', Salario: $2000000.0 }
Empleado { ID: 2, Nombre: 'Claudia Java', Puesto: 'Administradora', Salario: $950000.0 }
Empleado { ID: 3, Nombre: 'Christian Lupa', Puesto: 'Profesor', Salario: $50000.0 }

Total empleados creados: 3
```