

## Trabajo Práctico 4: Programación Orientada a Objetos II

1)



```
1 package empleadomain;
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() {
10         return id;
11     }
12     public void setId(int id) {
13         this.id = id;
14     }
15     public String getNombre() {
16         return nombre;
17     }
18     public void setNombre(String nombre) {
19         this.nombre = nombre;
20     }
21     public String getPuesto() {
22         return puesto;
23     }
24     public void setPuesto(String puesto) {
25         this.puesto = puesto;
26     }
27     public double getSalario() {
28         return salario;
29     }
30     public void setSalario(double salario) {
31         this.salario = salario;
32     }
33     public static int getTotalEmpleados() {
34         return totalEmpleados;
35     }
36     public static void setTotalEmpleados(int totalEmpleados) {
37         Empleado.totalEmpleados = totalEmpleados;
38     }
39     public Empleado(int id, String nombre, String puesto, double salario) {
40         Empleado.totalEmpleados++;
41         this.id = id;
42         this.nombre = nombre;
43         this.puesto = puesto;
44         this.salario = salario;
45     }
46     public Empleado(String nombre, String puesto) {
```

Output - EmpleadoMain (run) ×

```

Refactor Run Debug Profile Team Tools Window Help
EmpleadoMain - Apache NetBeans IDE 26
default config> 239,0/504,0MB Resume (pending events: 7)

EmpleadoMain.java x Empleado.java x
Source History
39 public Empleado(int id, String nombre, String puesto, double salario) {
40     Empleado.totalEmpleados++;
41     this.id = id;
42     this.nombre = nombre;
43     this.puesto = puesto;
44     this.salario = salario;
45 }
46 public Empleado(String nombre, String puesto) {
47     Empleado.totalEmpleados++;
48     this.id = Empleado.totalEmpleados;
49     this.nombre = nombre;
50     this.puesto = puesto;
51     this.salario = 650000.0;
52 }
53 public void actualizarSalario(double aumento) {
54     this.salario += ((salario * aumento) / 100);
55 }
56 public void actualizarSalario() {
57     actualizarSalario(15);
58 }
59 @Override
60 public String toString() {
61     return "Empleado(" + "id=" + id + ", nombre=" + nombre + ", puesto=" + puesto + ",salario=" + salario + ')';
62 }
63
64 public static int mostrarTotalEmpleados() {
65     return Empleado.totalEmpleados;
66 }
67 }
68
69

```

2)

```

EmpleadoMain.java x Empleado.java x
Source History
1 package empleadomain;
2
3 public class EmpleadoMain {
4     public static void main(String[] args) {
5         Empleado emp1 = new Empleado(1, "Empleado 1", "Vendedor", 650000);
6         Empleado emp2 = new Empleado(2, "Empleado 2", "Vendedor", 800000);
7         Empleado emp3 = new Empleado(3, "Empleado 3", "Encargado", 1250000);
8         Empleado emp4 = new Empleado("Empleado 4", "Repositor");
9         Empleado emp5 = new Empleado("Empleado 5", "Limpieza");
10        Empleado emp6 = new Empleado("Empleado 6", "Deposito");
11
12        emp1.actualizarSalario();
13        emp3.actualizarSalario(20);
14        emp5.actualizarSalario();
15
16        System.out.println(emp1.toString());
17        System.out.println(emp2.toString());
18        System.out.println(emp3.toString());
19        System.out.println(emp4.toString());
20        System.out.println(emp5.toString());
21        System.out.println(emp6.toString());
22
23        System.out.println("\nTotal empleados: " + Empleado.mostrarTotalEmpleados());
24    }
25 }
26

```

Output - EmpleadoMain (run) x

```

run:
Empleado{id=1, nombre=Empleado 1, puesto=Vendedor,salario=747500.0}
Empleado{id=2, nombre=Empleado 2, puesto=Vendedor,salario=800000.0}
Empleado{id=3, nombre=Empleado 3, puesto=Encargado,salario=1500000.0}
Empleado{id=4, nombre=Empleado 4, puesto=Repositor,salario=650000.0}
Empleado{id=5, nombre=Empleado 5, puesto=Limpieza,salario=747500.0}
Empleado{id=6, nombre=Empleado 6, puesto=Deposito,salario=650000.0}

Total empleados: 6
BUILD SUCCESSFUL (total time: 0 seconds)

```