

M09 - PSP - UF1 - ACTIVIDAD 1 REPASO

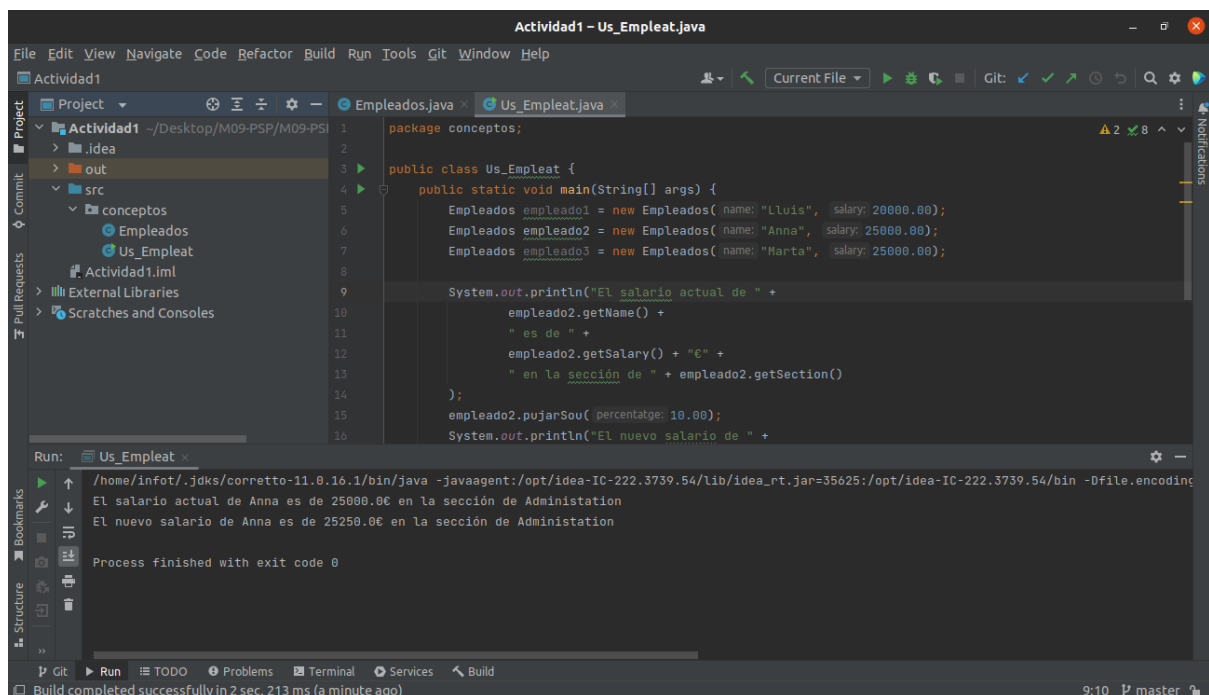
Què aconseguim declarant aquestes variables com a privades?

Al declarar las variables como privadas lo que hacemos es que su uso se limita únicamente a la clase, pudiendo únicamente interactuar con él añadiendo getters y setters.

Per què fem la declaració d'aquests mètodes com a públic?

Los creamos como públicos para poder acceder a ellos desde la clase principal.

Datos de empleado2 + su cambio de salario



The screenshot shows an IDE window titled 'Actividad1 - Us_Empleat.java'. The left sidebar displays a project structure for 'Actividad1' with a 'src' folder containing 'empleados' and 'Us_Empleat'. The main editor shows the following Java code:

```
1 package conceptos;
2
3 public class Us_Empleat {
4     public static void main(String[] args) {
5         Empleados empleado1 = new Empleados( "Lluís", salary: 20000.00);
6         Empleados empleado2 = new Empleados( "Anna", salary: 25000.00);
7         Empleados empleado3 = new Empleados( "Marta", salary: 25000.00);
8
9         System.out.println("El salario actual de " +
10             empleado2.getName() +
11             " es de " +
12             empleado2.getSalary() + "€" +
13             " en la sección de " + empleado2.getSection()
14         );
15         empleado2.pujarSou( porcentaje: 10.00);
16         System.out.println("El nuevo salario de " +
```

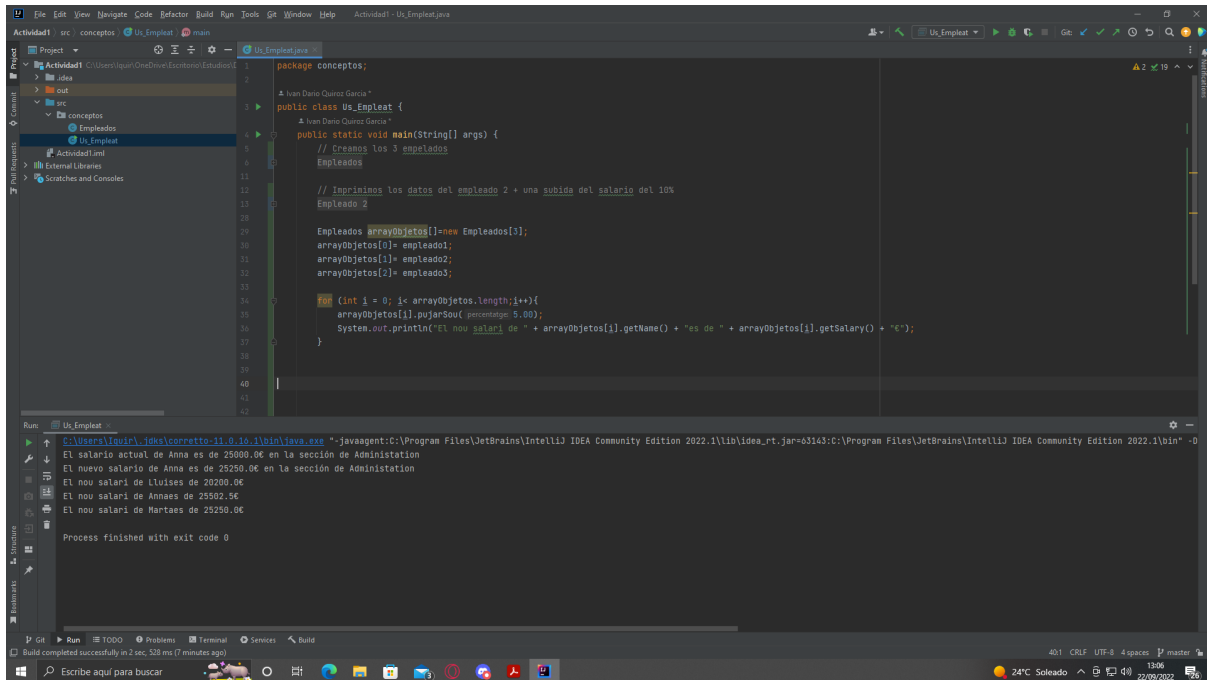
The bottom panel shows the execution output:

```
Run: Us_Empleat x
/home/infot/.jdk/corretto-11.0.16.1/bin/java -javaagent:/opt/idea-IC-222.3739.54/lib/idea_rt.jar=35625:/opt/idea-IC-222.3739.54/bin -Dfile.encoding=UTF-8
El salario actual de Anna es de 25000.00€ en la sección de Administration
El nuevo salario de Anna es de 25250.00€ en la sección de Administration
Process finished with exit code 0
```

The status bar at the bottom indicates 'Build completed successfully in 2 sec, 213 ms (a minute ago)' and the time is 9:10 on the master branch.

M09 - PSP - UF1 - ACTIVIDAD 1 REPASO

Creación de la Array + subida de sueldo



```
package conceptos;

import java.util.Scanner;

public class Ux_Emplsat {

    // Ivan Dami Queiroz Garcia
    // Creamos los 3 empleados
    Empleados

    // Imprimimos los datos del empleado 2 + una subida del salario del 10%
    Empleado 2

    Empleados arrayObjetos[] = new Empleados[3];
    arrayObjetos[0] = empleado1;
    arrayObjetos[1] = empleado2;
    arrayObjetos[2] = empleado3;

    for (int i = 0; i < arrayObjetos.length; i++) {
        arrayObjetos[i].pujarSou(percentage 5.00);
        System.out.println("El nou salari de " + arrayObjetos[i].getName() + " es de " + arrayObjetos[i].getSalary() + "€");
    }
}
```

Run: Ux_Emplsat - javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2022.1\lib\idea_rt.jar=63143:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2022.1\bin" -D

El salario actual de Anna es de 25000.0€ en la sección de Administration
El nuevo salario de Anna es de 25250.0€ en la sección de Administration
El nou salari de Lluïses de 20200.0€
El nou salari de Annaes de 25502.5€
El nou salari de Martaes de 25250.0€

Process finished with exit code 0

Com s’anomena aquesta propietat de la POO, mitjançant la qual podem utilitzar, per a un mètode de la subclasse, el mateix nom que el mètode de la superclasse? Com funciona aquesta propietat en la pràctica?

Polimorfismo. Consiste en la capacidad que tienen los objetos de una clase en ofrecer respuesta distinta e independiente en función de los parámetros

Es possible fer aquesta operació? Com s’anomena dintre del llenguatge de la POO?

Sí, es posible. Se llama Herencia.

He realizado hasta el apartado de “Visualitza, amb un “For Each” el nom, el sou i la secció dels 4 empleats i comprova que el mateix programa ja detecta si l’element de l’array és un empleat o un cap (Enllaçat dinàmic)”. Aunque lo he hecho con un “For” y el nombre de las clases no los he logrado poner del todo bien. (Mirar código)

M09 - PSP - UF1 - ACTIVIDAD 1 REPASO