

## Refactorización

En esta practica vamos a aprender a Refactorizar mediante algunos de los métodos de refactorización explicados en clase.

Os dejo en el Moodle la clase Trabajador en la que podemos observar que un trabajador esta formado por los siguientes campos:

```
public String n;//nombre
public int e;//edad
public int cat;//categoria
public int ant;//antigüedad
public double sue;//sueldo
```

¿Os parecen adecuados los nombres escogidos para cada uno de los atributos del trabajador?

1) Debéis renombrar cada parámetro por el nombre que aparece en el comentario situado a su lado.

2) Una vez tengamos renombrados los atributos vamos a proceder a tabular de manera automática el código, ya que es más fácil comprender un código si esta tabulado.

3) Una vez tenemos los atributos definidos de manera correcta, quiero que todos atributos estén encapsulados. Aplicaremos esta encapsulación de manera automática.

4) Por ultimo, vemos que dentro del Main existen unas formulas para calcular el sueldo del trabajador, como podemos observar en la siguiente imagen:

```
public static void main(String[] args) {

    Trabajador t1 = new Trabajador();

    // Calculo del sueldo del trabajador
    if (t1.cat == 0 && t1.ant == 0) {
        t1.sue = t1.sue + t1.sue * 0.15 + 150;
    } else if (t1.cat == 0 && t1.ant == 1) {
        t1.sue = t1.sue + t1.sue * 0.15 + 300;
    } else if (t1.cat == 0 && t1.ant == 2) {
        t1.sue = t1.sue + t1.sue * 0.15 + 600;
    } else if (t1.cat == 1 && t1.ant == 0) {
        t1.sue = t1.sue + t1.sue * 0.35 + 150;
    } else if (t1.cat == 1 && t1.ant == 1) {
        t1.sue = t1.sue + t1.sue * 0.35 + 300;
    } else if (t1.cat == 1 && t1.ant == 2) {
        t1.sue = t1.sue + t1.sue * 0.35 + 600;
    } else if (t1.cat == 2 && t1.ant == 0) {
        t1.sue = t1.sue + t1.sue * 0.6 + 150;
    } else if (t1.cat == 2 && t1.ant == 1) {
        t1.sue = t1.sue + t1.sue * 0.6 + 300;
    } else {
        t1.sue = t1.sue + t1.sue * 0.6 + 600;
    }

}
```

Esto no es adecuado realizarlo de esta manera, como solución a este problema os propongo encapsular todas esos cálculos dentro de una nueva función. Esto se denomina extraer método y el propio IDE nos ayuda a realizarlo de manera automática. Vamos a crear la acción llamada calculaSuelo(Trabajador t1).

Adaptaremos el código de manera manual para que nos muestre por pantalla el calculo del sueldo del trabajador, añadiendo en el main:

```
System.out.println("El sueldo del trabajador es: " +  
getSuelo());
```

Vamos a proceder a probar el código completo creando el siguiente trabajador:

```
Trabajador t1 = new Trabajador("Pepe",29,1,2,830);
```

Salida por consola: El sueldo del trabajador es: 1720.5

Ya hemos mejorado el código de la clase de manera automática.