

TALLER DE REFACTORING

INTEGRANTES:

-JEREMY ALEXANDER RAMIREZ GALEOTTI

-DIANA CAROLINA RAMIREZ NEIRA

-RONNY RAUL ANDRADE MORA

Contenido

[DUPLICATE CODE 3](#_Toc48249836)

[CLASE Estudiante 3](#_Toc48249837)

[Solución: Consolidate Conditional Expression 3](#_Toc48249838)

[TEMPORARY FIELD 4](#_Toc48249839)

[CLASE calcularSueldoProfesor 4](#_Toc48249840)

[Solución: Inline TEMP 4](#_Toc48249841)

[LONG PARAMETER LIST 5](#_Toc48249842)

[CLASE Estudiante 5](#_Toc48249843)

[Solución: Preserve Whole Object . 5](#_Toc48249844)

[INNAPROPIATE INTIMACY 6](#_Toc48249845)

[CLASE Ayudante 6](#_Toc48249846)

[Solución: Replace Delegation with Inheritance 6](#_Toc48249847)

[LAZY CLASS 8](#_Toc48249848)

[CLASE InformacionAdicionalProfesor 8](#_Toc48249849)

[Solución: Inline Class 8](#_Toc48249850)

[DATA CLASS 9](#_Toc48249851)

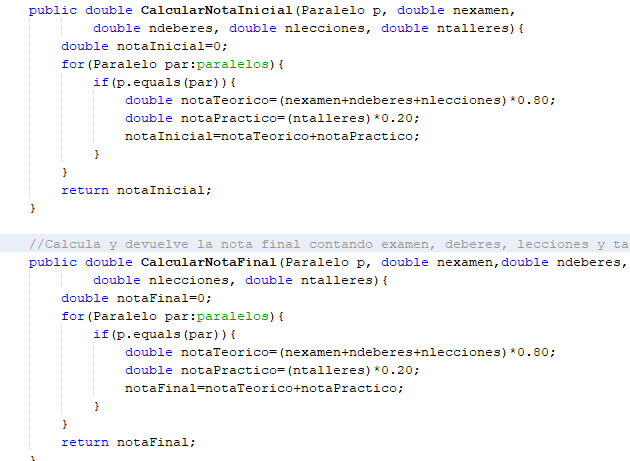
[CLASE Materia 9](#_Toc48249852)

[Solución: Encapsulate Field 9](#_Toc48249853)

# DUPLICATE CODE

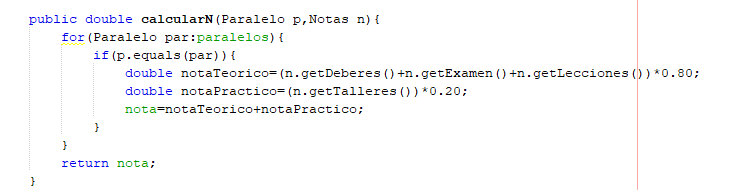
## CLASE Estudiante

Se encontró este tipo de code smell al notar que los métodos CalcularNotaInical y CalcularNotaFinal ejercen exactamente el mismo procedimiento. De seguir el smell presente el código sería demasiado largo, lo cual afectaría la mantenibilidad del software.



## Solución: Consolidate Conditional Expression

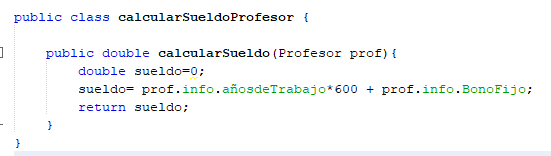
Se consolidó los condicionales de los 2 métodos en uno solo en un nuevo método.



# TEMPORARY FIELD

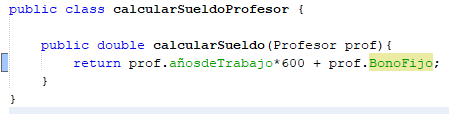
## CLASE calcularSueldoProfesor

Se encontró este tipo de code smell al detectar la creación de la variable ‘’sueldo’’ que solo se usará en ciertas ocasiones. De seguir el smell puede volver el código difícil de entender para algún otro programador o el mismo creador del código pasado un tiempo.



## Solución: Inline TEMP

Se realizó directamente el retorno del sueldo sin ser almacenado en una variable.



# LONG PARAMETER LIST

## CLASE Estudiante

Se encontró este tipo de code smell al notar que los parámetros que reciben los métodos CalcularNotaInicial y CalcularNotaFinal son más de 4. Es dificil entender las listas que se vuelven contradictorias y difíciles de usar, de continuar dicho smell, podría ocurrir un error al momento de utilizar el método.

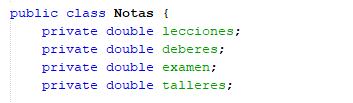




## Solución: [Preserve Whole Object](https://sourcemaking.com/refactoring/preserve-whole-object) .

Se procedió a crear una clase nueva ‘Notas’, donde se almacenan todos los tipos de notas, y se procedió a pasarla por un parámetro en el método ‘calcularN’, para disminuir la cantidad de ellos.

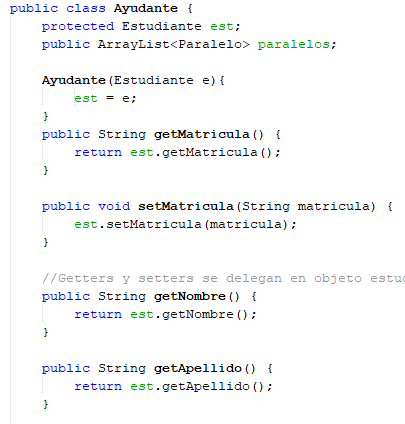




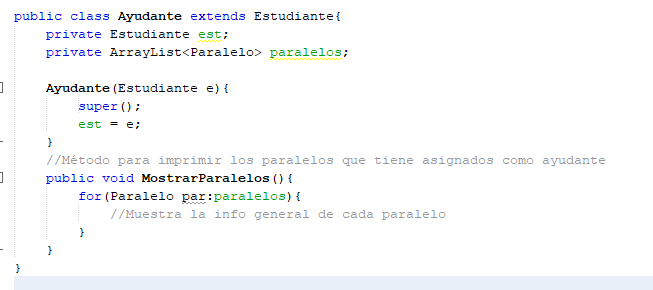
# INNAPROPIATE INTIMACY

## CLASE Ayudante

Se encontró este tipo de code smell al notar que la clase Ayudante utiliza atributos y métodos de otra clase (Estudiante). Al usar una clase una cantidad significativa de métodos de otra, tendrían demasiado acoplamiento lo cual haría que ambas sean difíciles de mantener, cambiar y comprender.



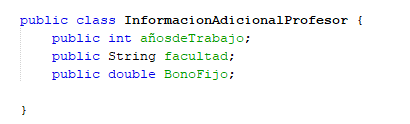
## Solución: Replace Delegation with Inheritance

Se extendió de la clase Estudiante, para poder acceder a los métodos necesarios. 

# LAZY CLASS

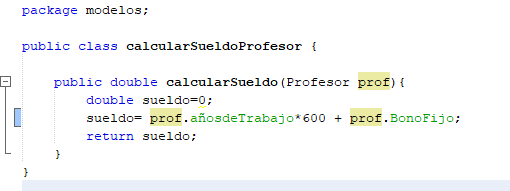
## CLASE InformacionAdicionalProfesor

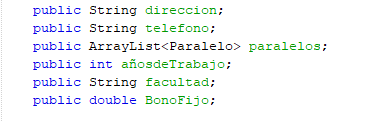
Se encontró este tipo de code smell al observar que los datos en InformaciónAdicionalProfesor no son de tan relevante importancia como para tener una clase propia. De continuar el smell, el software se volvería largo, difícil de entender y posiblemente costoso de mantener.



## Solución: [Inline Class](https://refactoring.guru/inline-class)

Se eliminó la clase InformacionAdicionalProfesor y los parámetros de dicha clase se los pasó a la clase Profesor.



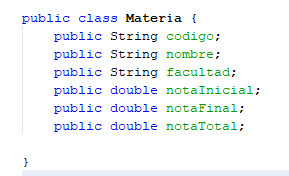


# DATA CLASS

## CLASE Materia

Se encontró este tipo de code smell al observar que la clase Materia solo posee atributos públicos.

Como esta clase no posee ninguna funcionalidad extra, de mantenerse el smell no podría funcionar de forma independiente con sus datos.



## Solución: [Encapsulate Field](https://refactoring.guru/encapsulate-field)

Se procedió a cambiar el parámetro de acceso a private , seguido de codificar los getters y setters correspondientes.

