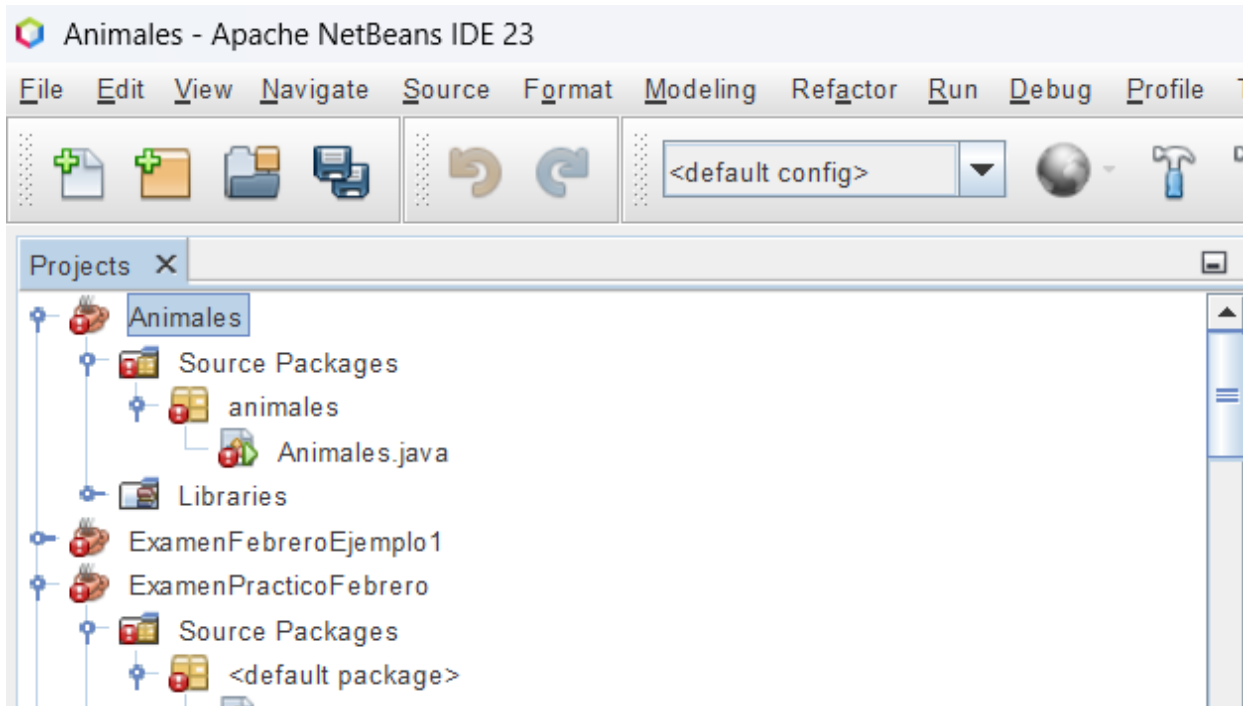


## Nombre:

Realizar las siguientes tareas, de forma que para cada una de ellas quede constancia de que la tarea ha sido realizada.

1. Se nos ha asignado la tarea de realizar la codificación de una clase que formará parte de un software de gestión para un zoológico. Para ellos debemos realizar las siguientes tareas:

1.1. Crear un nuevo proyecto con el nombre "*Animales*".



1.2. Crear una nueva clase llamada "*Pájaro*" que contendrá la información referente al cuidado del ave. La clase contendrá las siguientes variables:

- *nombre\_especie*: indicará el nombre científico de la especie del ave.
- *envergadura*: indicará la distancia entre las puntas de las alas del ave cuando aquellas están completamente abiertas, en centímetros.
- *peso*: peso en gramos del animal.

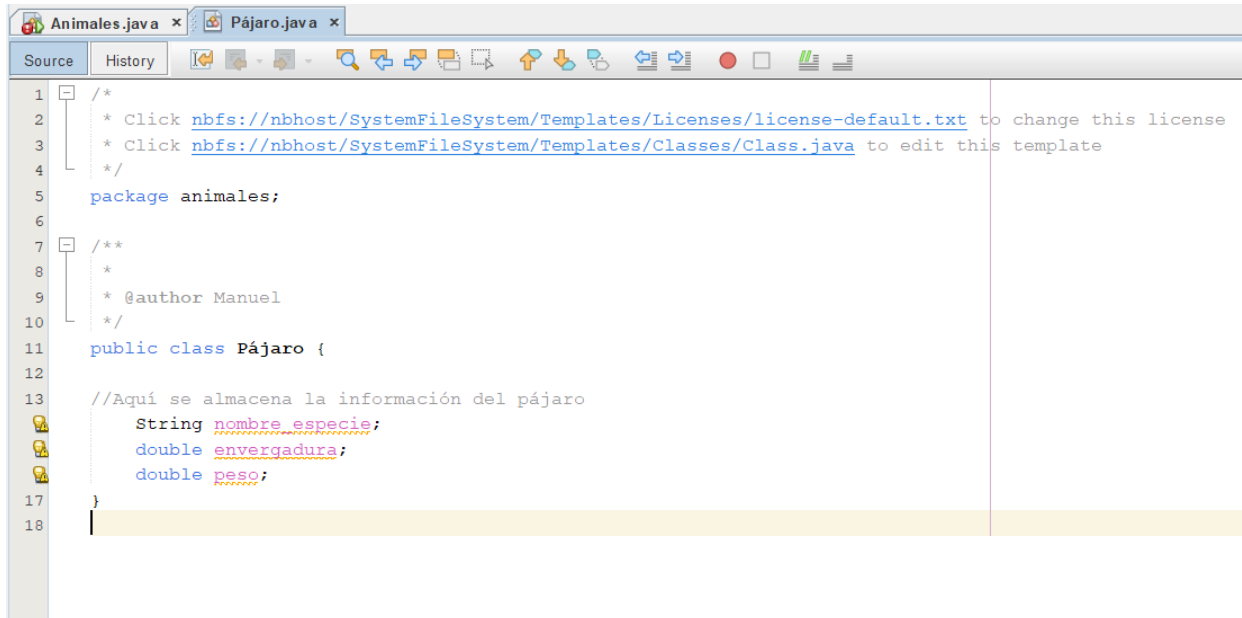
El código para la clase será el siguiente:

```
public class Pájaro {  
  
    //Aquí se almacena la información del pájaro  
  
    String nombre_especie;
```

*double envergadura;*

*double peso;*

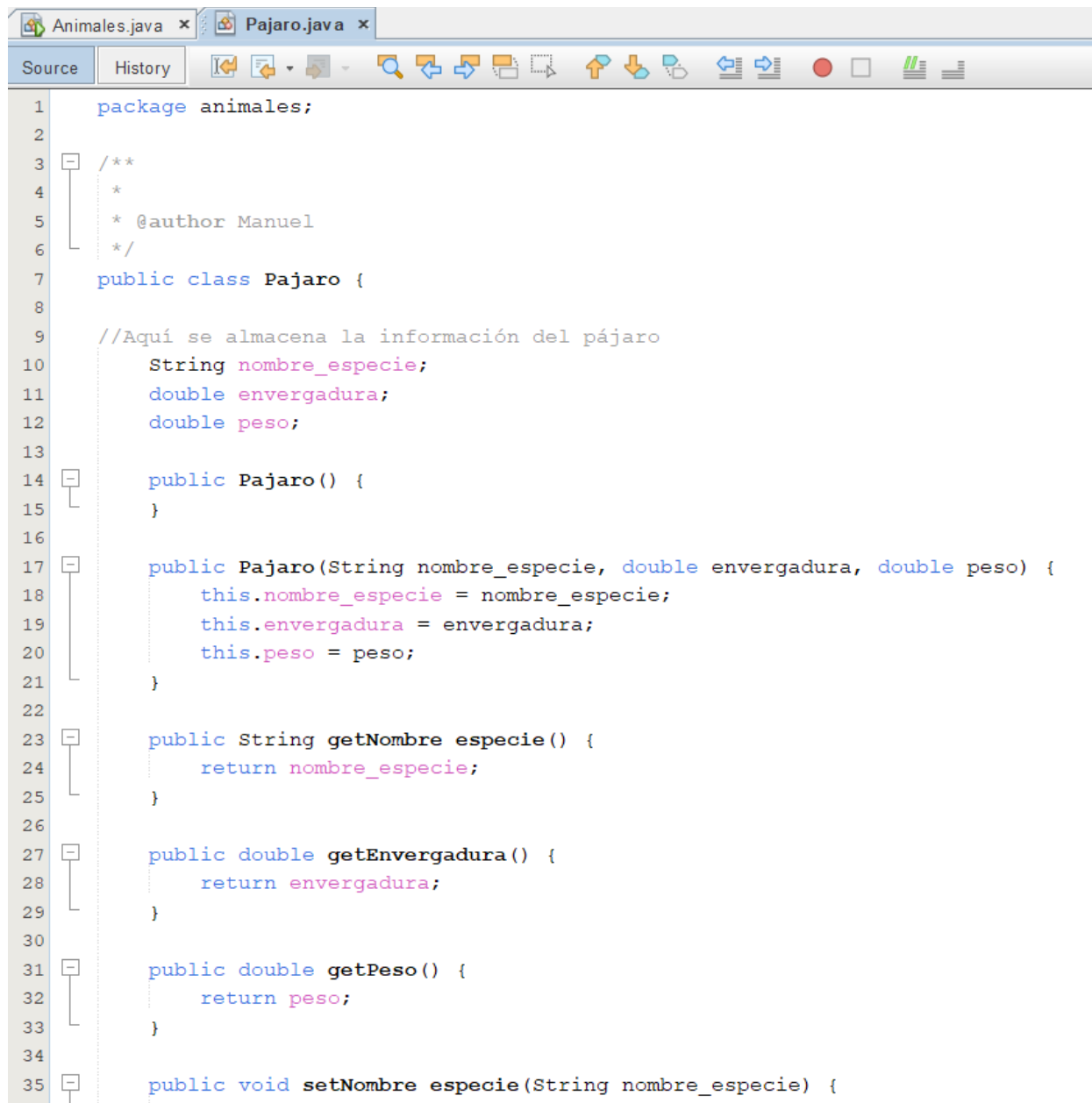
}



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

```
1  /*
2   * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
3   * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
4   */
5   package animales;
6
7   /**
8    *
9    * @author Manuel
10   */
11   public class Pájaro {
12
13       //Aquí se almacena la información del pájaro
14       String nombre especie;
15       double envergadura;
16       double peso;
17   }
18
```

1.3. Crear dos métodos constructores (uno sin parámetros y otro con todos los parámetros), además de los métodos getter & setter para todas las variables de clase. Para esta tarea se debe usar la generación de código automática.



```
1 package animales;
2
3 /**
4  *
5  * @author Manuel
6  */
7 public class Pajaro {
8
9     //Aquí se almacena la información del pájaro
10    String nombre_especie;
11    double envergadura;
12    double peso;
13
14    public Pajaro() {
15    }
16
17    public Pajaro(String nombre_especie, double envergadura, double peso) {
18        this.nombre_especie = nombre_especie;
19        this.envergadura = envergadura;
20        this.peso = peso;
21    }
22
23    public String getNombre especie() {
24        return nombre_especie;
25    }
26
27    public double getEnvergadura() {
28        return envergadura;
29    }
30
31    public double getPeso() {
32        return peso;
33    }
34
35    public void setNombre especie(String nombre_especie) {
```

1.4. Crear un método llamado imprimir que muestre los datos de un objeto de esta clase. El código fuente para este método será el siguiente:

```
public void imprimir () {
    System.out.println("ZOOLOGICO DE CUENCA – SECCIÓN DE AVES");
    System.out.println("Registro Sanitario No 48/38751");
    System.out.println("Para consultar el nombre del animal revise la etiqueta");
    System.out.println("Peso: "+this.peso);
    System.out.println("Envergadura: " + this.envergadura);
    System.out.println("Nombre de la especie: " + this.nombre_especie);
}
```

```

65 public void imprimir() {
66     System.out.println("ZOOLOGICO DE CUENCA - SECCIÓN DE AVES");
67     System.out.println("Registro Sanitario No 48/38751");
68     System.out.println("Para consultar el nombre del animal revise la etiqueta");
69     System.out.println("Peso: " + this.peso);
70     System.out.println("Envergadura: " + this.envergadura);
71     System.out.println("Nombre de la especie: " + this.especie);
72 }
73 }
74

```

1.5. Generar los comentarios *Javadoc* para nuestro proyecto, incluyendo comentarios para la clase y todos sus métodos. Comprobar mediante la herramienta "Analyze Javadoc" que no tenemos ningún método sin documentar y ningún error en la documentación.

PEGA AQUÍ UNA CAPTURA DE LA SOLUCIÓN

1.6. Una vez comprobado que nuestra documentación está preparada (sin errores en la salida del *Analyzer*), generar la documentación *Javadoc* para el proyecto.

PEGA AQUÍ UNA CAPTURA DE LA SOLUCIÓN

1.7. Generar un primer commit con el comentario "v1 - Creación clase Pájaro". Aseguraos que se incluyen todos los ficheros del proyecto.

PEGA AQUÍ UNA CAPTURA DE LA SOLUCIÓN

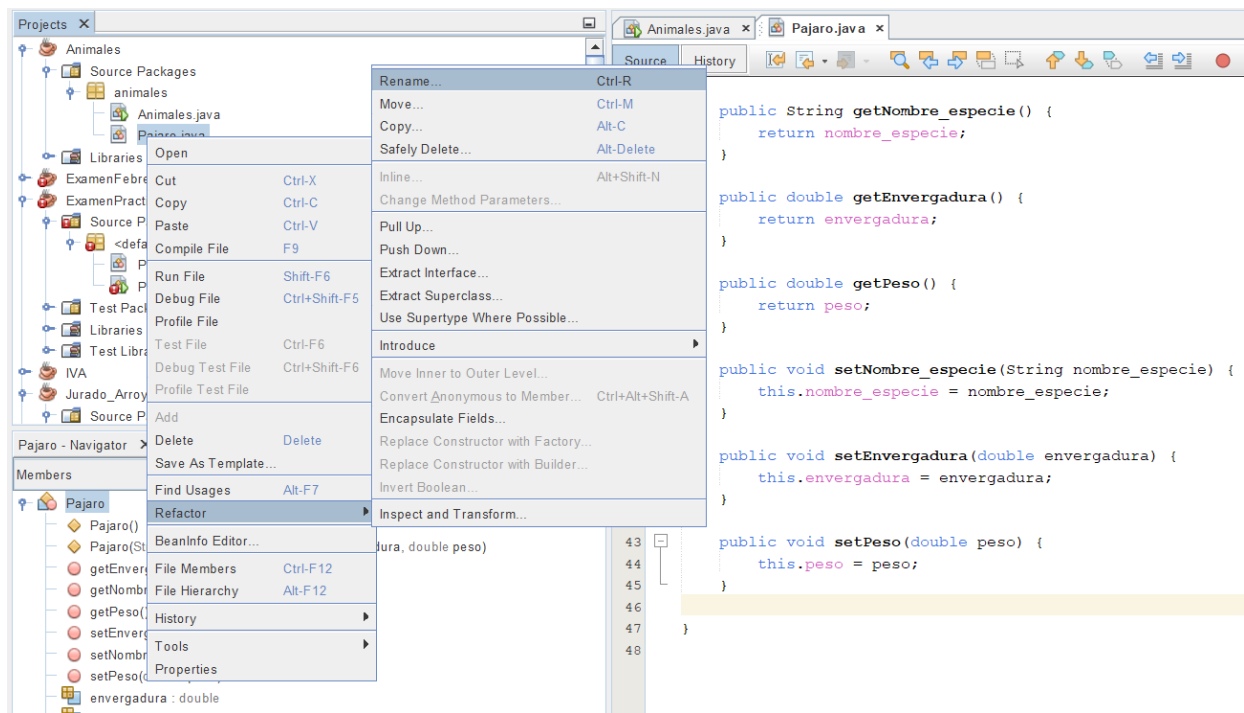
```

35 public void setNombre_especie(String nombre_especie) {
36     this.nombre_especie = nombre_especie;
37 }
38
39 public void setEnvergadura(double envergadura) {
40     this.envergadura = envergadura;
41 }
42
43 public void setPeso(double peso) {
44     this.peso = peso;
45 }
46
47 }
48

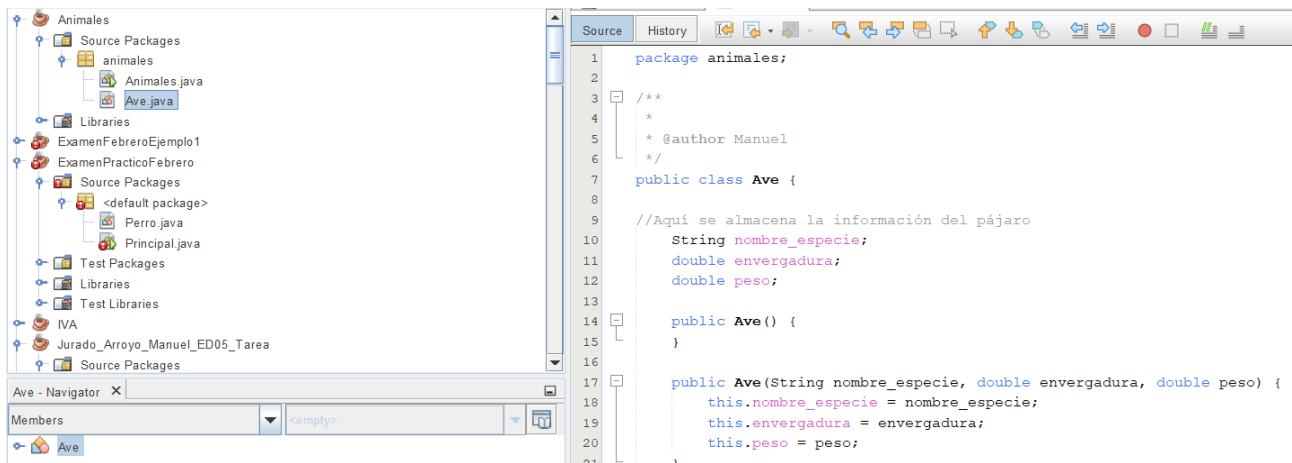
```

1.8. Cambiar el nombre a la clase Pájaro, asignando el nuevo nombre Ave. El cambio de nombre debe aplicarse también a los comentarios de la clase, si los hubiese.

PEGA AQUÍ UNA CAPTURA DE LA SOLUCIÓN







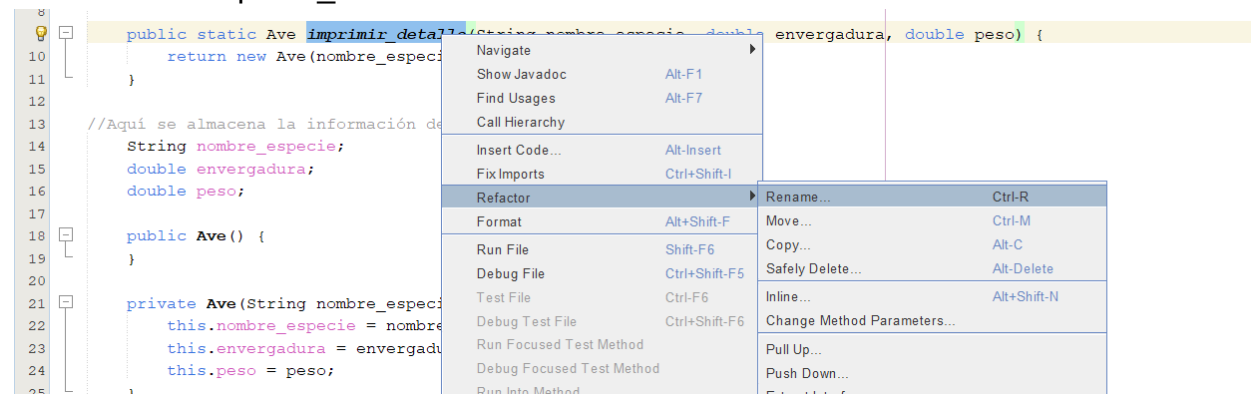
1.9. Extraer un método con las líneas que imprimen los datos del objeto Ave (es decir, las tres últimas líneas del método). El nuevo método se llamará imprimir\_detalle.

```

21 private Ave(String nombre_especie, double envergadura, double peso) {
22     this.nombre_especie = nombre_especie;
23     this.envergadura = envergadura;
24     this.peso = peso;
25 }

```

1.10. Cambiar el nombre al método imprimir de la clase Ave y asignarle el nuevo nombre imprimir\_cabecera.

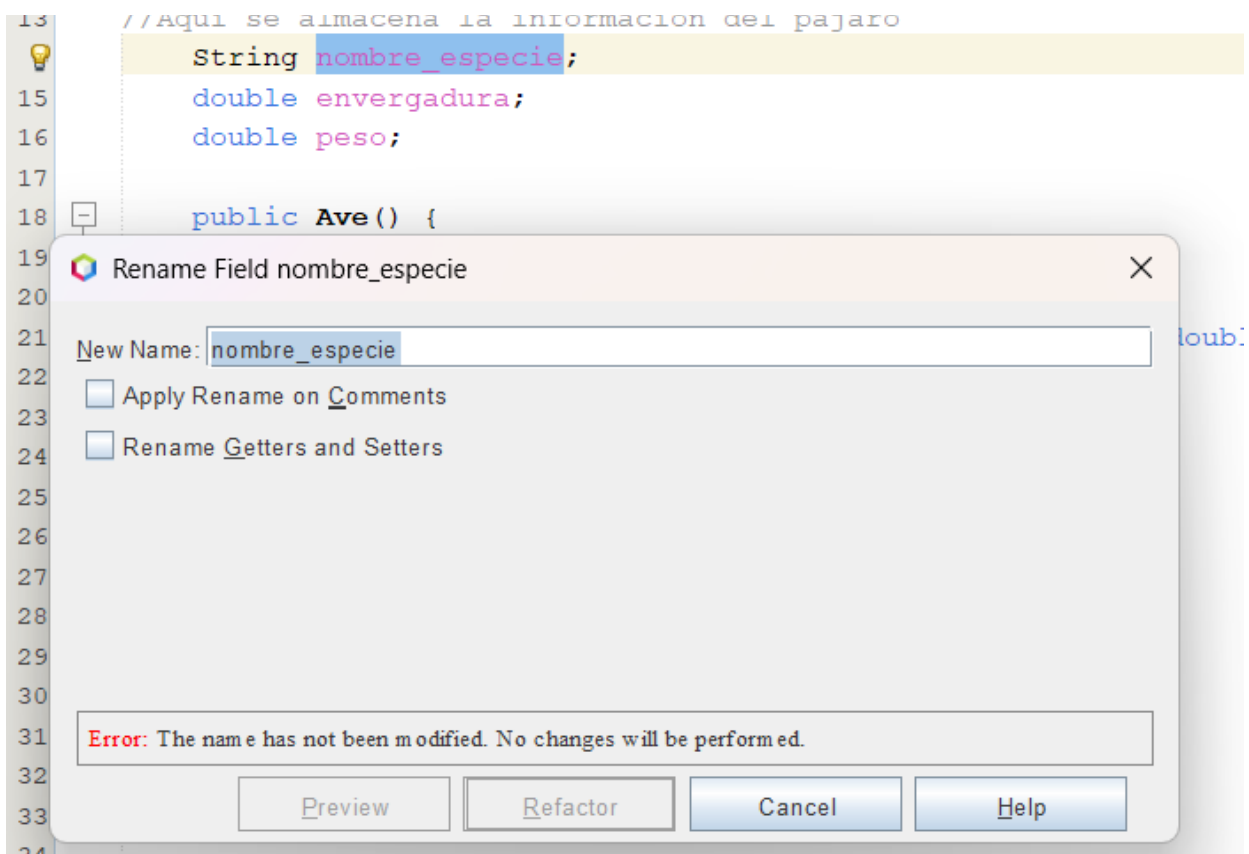


```

8 public static Ave imprimir_detalle(String nombre_especie, double envergadura, double peso) {
9     return new Ave(nombre_especie, envergadura, peso);
10 }
11
12

```

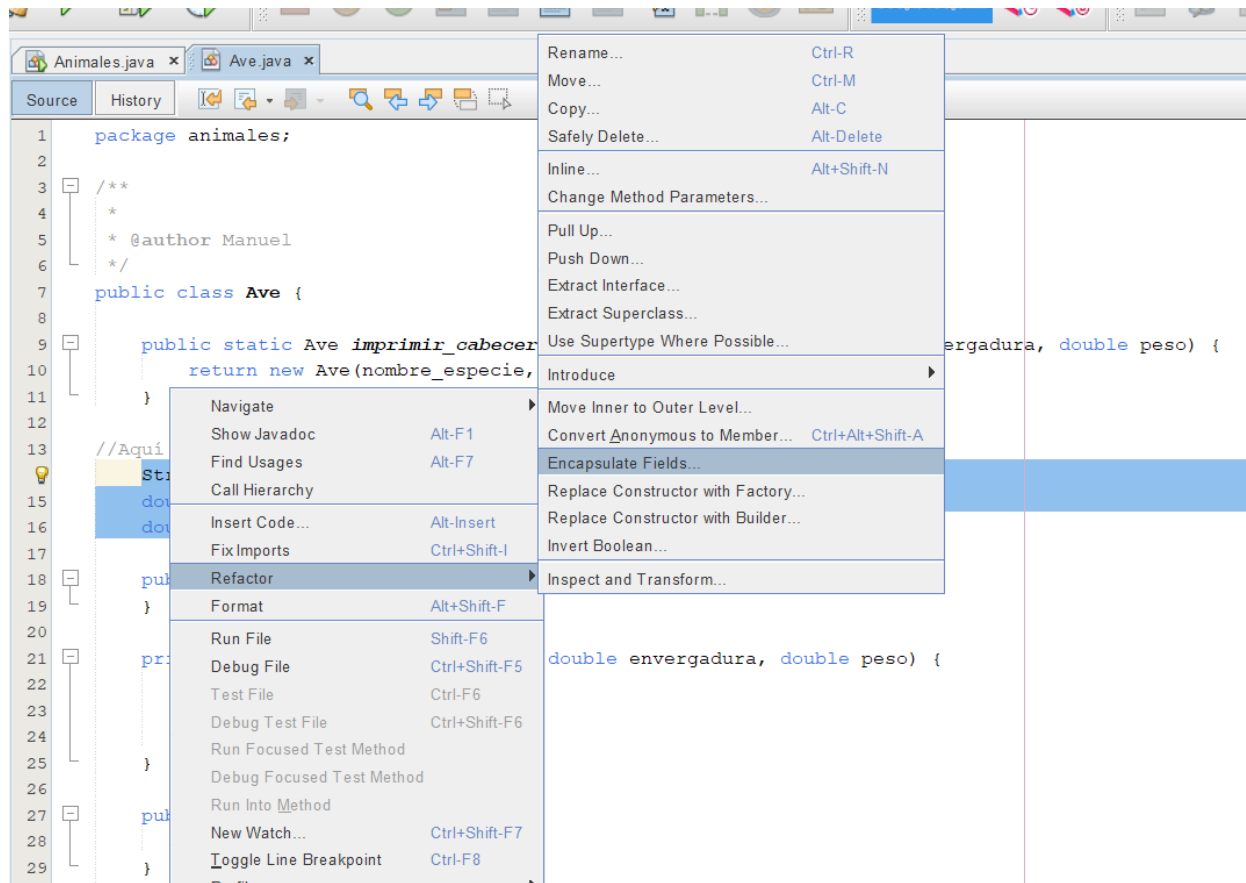
1.11. Cambiar el nombre a la variable nombre\_especie para que pase a llamarse especie, de forma que el cambio aplique también a los métodos creados.



```
13 //Aquí se almacena la información del pájaro
14 String especie;
15 double envergadura;
16 double peso;
17
18 public Ave() {
19 }
20
21 private Ave(String nombre_especie, double envergadura, double peso) {
22     this.especie = nombre_especie;
23     this.envergadura = envergadura;
24     this.peso = peso;
25 }
26
27 public String getNombre especie() {
28     return especie;
29 }
30
31 public double getEnvergadura() {
32     return envergadura;
33 }
34
35 public double getPeso() {
```

1.12. Encapsular todas las variables de clase de la clase Ave.





```

12 public String getEspecie() {
13     return especie;
14 }
15
16 /**
17  * @param especie the especie to set
18  */
19 public void setEspecie(String especie) {
20     this.especie = especie;
21 }
22
23 public static Ave imprimir_cabecera(String nombre_especie, double envergadura, double peso) {
24     return new Ave(nombre_especie, envergadura, peso);
25 }
26
27 //Aquí se almacena la información del pájaro
28 private String especie;
29 private double envergadura;
30 private double peso;
31
32 public Ave() {
33 }
34

```

1.13. Generar una segunda versión con el comentario "v2 - Primera Entrega".

```
MINGW64:/c:/Users/Manuel/OneDrive/Desktop/Examen Mayo/examen-endes-...
Manuel@DESKTOP-CJVI0DK MINGW64 ~/OneDrive/Desktop/Examen Mayo/examen-endes-temas-4-5-y-6-Manuel-Jurado-Arroyo/Parte 1 (main)
$ git commit -m "v2 - Primera Entrega"
[main 3a7172d] v2 - Primera Entrega
14 files changed, 2053 insertions(+)
create mode 100644 "Parte 1/~lock.Parte 1 - Refactorizaci\303\263n.odt#"
create mode 100644 Parte 1/Animales/build.xml
create mode 100644 Parte 1/Animales/build/classes/.netbeans_automatic_build
create mode 100644 Parte 1/Animales/build/classes/.netbeans_update_resources
create mode 100644 Parte 1/Animales/build/classes/animales/Animales.class
create mode 100644 Parte 1/Animales/build/classes/animales/Ave.class
create mode 100644 Parte 1/Animales/manifest.mf
create mode 100644 Parte 1/Animales/nbproject/build-impl.xml
create mode 100644 Parte 1/Animales/nbproject/genfiles.properties
create mode 100644 Parte 1/Animales/nbproject/private/private.properties
create mode 100644 Parte 1/Animales/nbproject/project.properties
create mode 100644 Parte 1/Animales/nbproject/project.xml
create mode 100644 Parte 1/Animales/src/animales/Animales.java
create mode 100644 Parte 1/Animales/src/animales/Ave.java
Manuel@DESKTOP-CJVI0DK MINGW64 ~/OneDrive/Desktop/Examen Mayo/examen-endes-temas-4-5-y-6-Manuel-Jurado-Arroyo/Parte 1 (main)
$ |
```

1.14. Mostrar las diferencias de los ficheros entre las versiones v1 y v2.  
*Dados los problemas que me encontré en mitad del examen a causa del LibreOffice no realicé el commit anterior*

```
Manuel@DESKTOP-CJVI0DK MINGW64 ~/OneDrive/Desktop/Examen Mayo/examen-endes-temas-4-5-y-6-Manuel-Jurado-Arroyo/Parte 1 (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
(use "git push" to publish your local commits)

Untracked files:
  (use "git add <file>..." to include in what will be committed)
  ../Parte 3/~lock.Parte 3 - Diagramas UML.odt#

nothing added to commit but untracked files present (use "git add" to track)
Manuel@DESKTOP-CJVI0DK MINGW64 ~/OneDrive/Desktop/Examen Mayo/examen-endes-temas-4-5-y-6-Manuel-Jurado-Arroyo/Parte 1 (main)
$
```

•

1.15. Comprobar en el browser de repositorios de Git que la nueva versión ha sido creada.

```
Manuel@DESKTOP-CJVI0DK MINGW64 ~/OneDrive/Desktop/Examen Mayo/examen-endes-temas-4-5-y-6-Manuel-Jurado-Arroyo/Parte 1 (main)
$ git log
commit 3a7172d55e8d0c96e791f80d4cb8071b267d162a (HEAD -> main)
Author: Manuel Jurado Arroyo <mjara2015@hotmail.com>
Date: Tue May 20 12:59:24 2025 +0200

    v2 - Primera Entrega

commit f5b436f42d4b9bd0cb1a4f0b824f590ea1a83738 (origin/main, origin/HEAD)
Author: github-classroom[bot] <66690702+github-classroom[bot]@users.noreply.github.com>
Date: Tue May 20 10:09:49 2025 +0000

    add deadline

commit 2d16ca5d301032e4cfa460ea98316a8f2df39c24
Author: github-classroom[bot] <66690702+github-classroom[bot]@users.noreply.github.com>
Date: Mon May 19 17:31:56 2025 +0000

    Initial commit
Manuel@DESKTOP-CJVI0DK MINGW64 ~/OneDrive/Desktop/Examen Mayo/examen-endes-temas-4-5-y-6-Manuel-Jurado-Arroyo/Parte 1 (main)
$
```