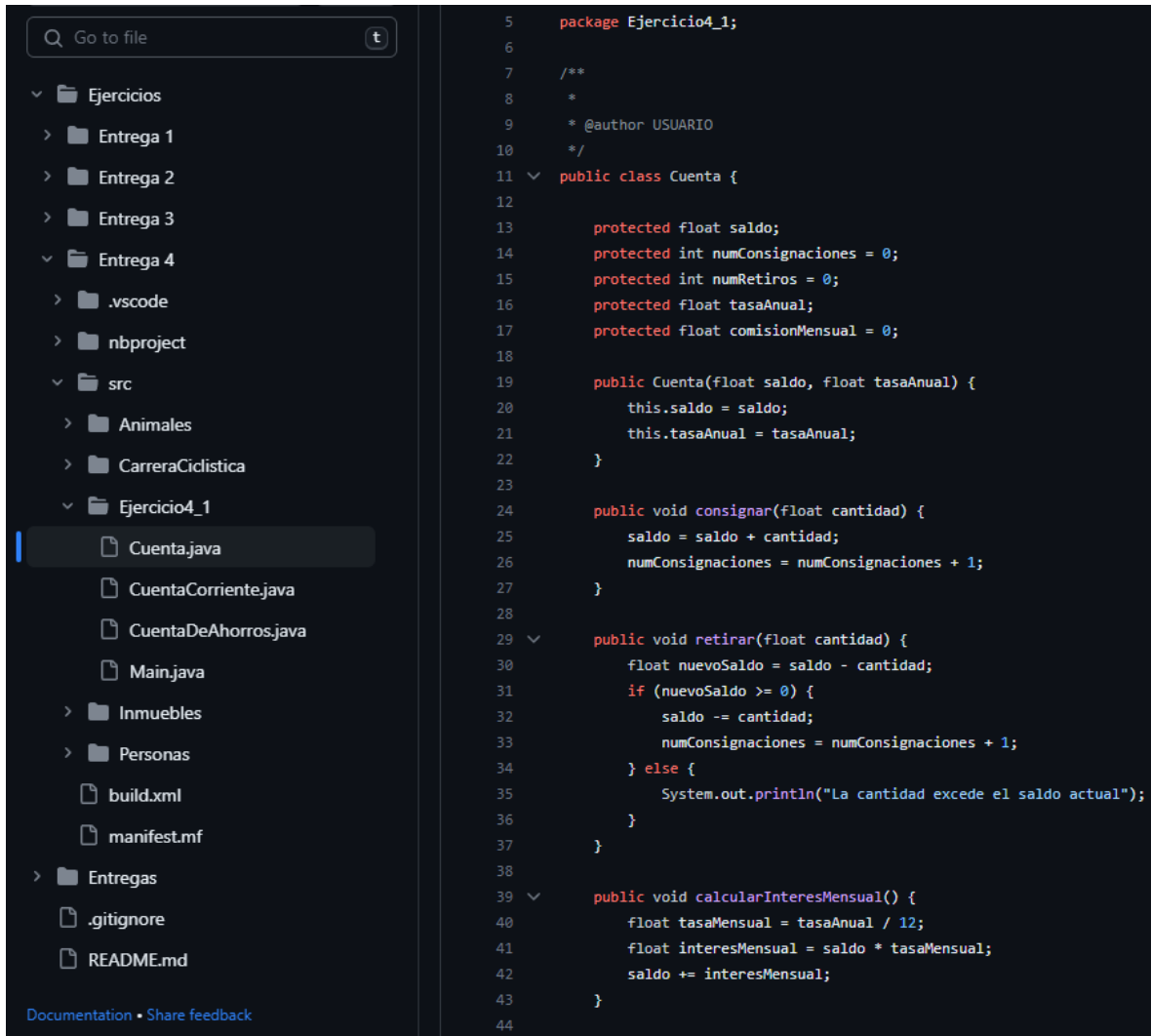


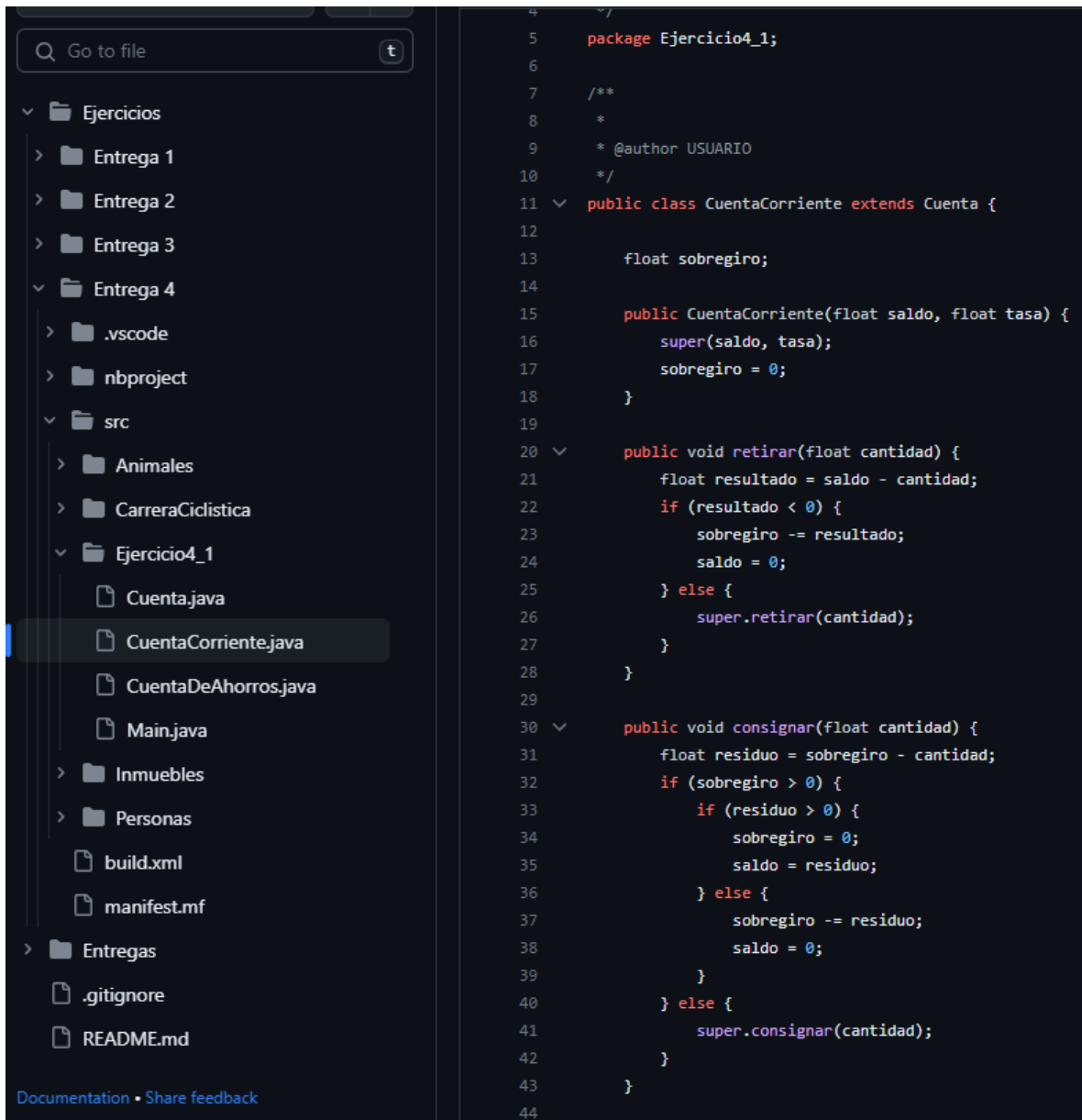
Entrega 4 POO 2023-2
Nombre: Mateo Sánchez
Correo: matsanchez@unal.edu.co
Profesor: Walter Hugo Arboleda Mazo
Link GitHub: <https://github.com/MateoS0/Entregas-POO-2023-2>



The image shows a VS Code editor interface. On the left, the 'Explorer' sidebar displays a project structure with folders 'Ejercicios', 'Entrega 1', 'Entrega 2', 'Entrega 3', 'Entrega 4', '.vscode', 'nbproject', 'src', 'Animales', 'CarreraCiclistica', 'Ejercicio4_1', 'Inmuebles', 'Personas', and 'Entregas'. The file 'Cuenta.java' is selected under 'Ejercicio4_1'. The main editor area shows the code for 'Cuenta.java'.

```
5 package Ejercicio4_1;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public class Cuenta {
12
13     protected float saldo;
14     protected int numConsignaciones = 0;
15     protected int numRetiros = 0;
16     protected float tasaAnual;
17     protected float comisionMensual = 0;
18
19     public Cuenta(float saldo, float tasaAnual) {
20         this.saldo = saldo;
21         this.tasaAnual = tasaAnual;
22     }
23
24     public void consignar(float cantidad) {
25         saldo = saldo + cantidad;
26         numConsignaciones = numConsignaciones + 1;
27     }
28
29     public void retirar(float cantidad) {
30         float nuevoSaldo = saldo - cantidad;
31         if (nuevoSaldo >= 0) {
32             saldo -= cantidad;
33             numConsignaciones = numConsignaciones + 1;
34         } else {
35             System.out.println("La cantidad excede el saldo actual");
36         }
37     }
38
39     public void calcularInteresMensual() {
40         float tasaMensual = tasaAnual / 12;
41         float interesMensual = saldo * tasaMensual;
42         saldo += interesMensual;
43     }
44
```

```
45     public void extractoMensual() {
46         saldo -= comisionMensual;
47         calcularInteresMensual();
48     }
49
50 }
```



```
45     public void extractoMensual(){
46         super.extractoMensual();
47     }
48
49     public void imprimir() {
50         System.out.println("Saldo = $ " + saldo);
51         System.out.println("Cargo mensual = $ " + comisionMensual);
52         System.out.println("Número de transacciones = " +
53             (numConsignaciones + numRetiros));
54         System.out.println("Valor de sogregiro = $ " +
55             (numConsignaciones + numRetiros));
56         System.out.println();
57     }
58
59 }
```

Q Go to file t

Ejercicios

> Entrega 1

> Entrega 2

> Entrega 3

> Entrega 4

> .vscode

> nbproject

> src

> Animales

> CarreraCiclistica

> Ejercicio4_1

Cuenta.java

CuentaCorriente.java

CuentaDeAhorros.java

Main.java

> Inmuebles

> Personas

build.xml

manifest.mf

> Entregas

.gitignore

README.md

Documentation • Share feedback

```
5 package Ejercicio4_1;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public class CuentaDeAhorros extends Cuenta {
12
13     private boolean activa;
14
15     public CuentaDeAhorros(float saldo, float tasa) {
16         super(saldo, tasa);
17         activa = saldo >= 10000;
18     }
19
20     public void consignar(float cantidad) {
21         if (activa) {
22             super.consignar(cantidad);
23         }
24     }
25
26     public void retirar(float cantidad) {
27         if (activa) {
28             super.retirar(cantidad);
29         }
30     }
31
32     public void extractoMensual() {
33         if (numRetiros > 4) {
34             comisionMensual += (numRetiros - 4) * 1000;
35         }
36         super.extractoMensual();
37         activa = saldo >= 10000;
38     }
39
40     public void imprimir() {
41         System.out.println("Saldo = $ " + saldo);
42         System.out.println("Comisión mensual = $ " +
43             comisionMensual);
44         System.out.println("Número de transacciones = " +
```

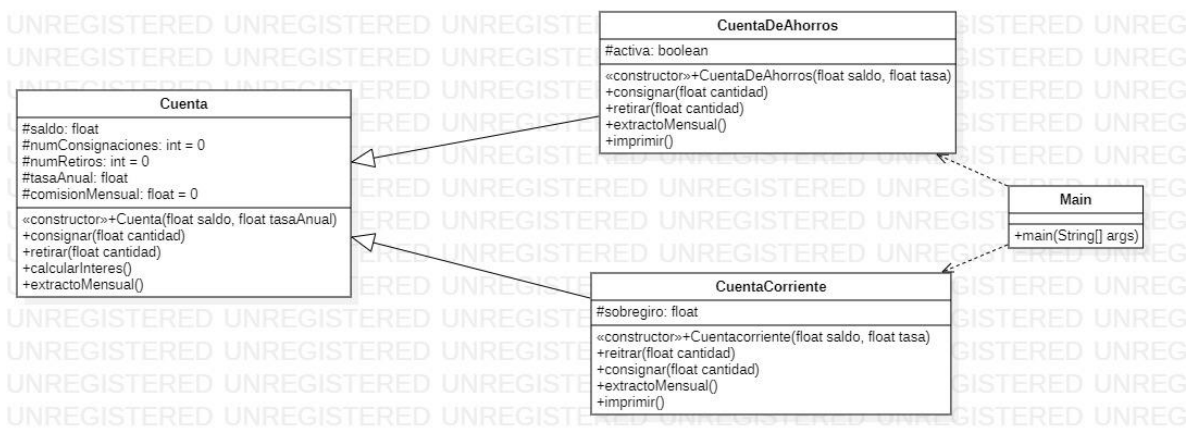
```
45         (numConsignaciones + numRetiros));
46         System.out.println();
47     }
48
49 }
```

Entrega 3
Entrega 4
.vscode
nbproject
src
Animales
CarreraCiclistica
Ejercicio4_1
Cuenta.java
CuentaCorriente.java
CuentaDeAhorros.java
Main.java
Inmuebles
Personas
build.xml
manifest.mf
Entregas
citigora

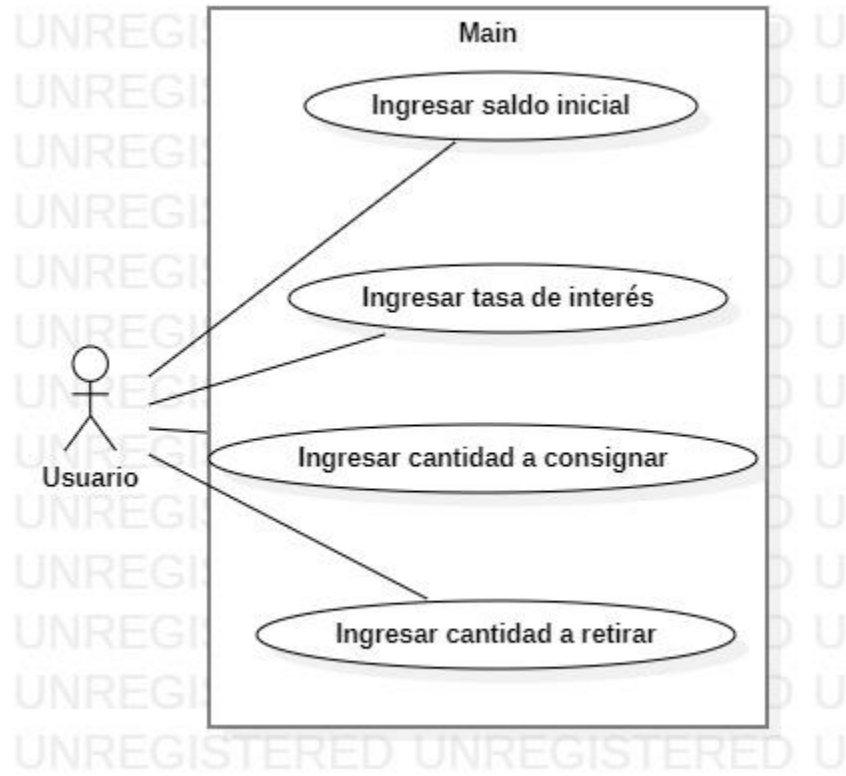
```

5 package Ejercicio4_1;
6
7 /**
8  *
9  * @author USUARIO
10 */
11
12 import java.util.*;
13 public class Main {
14     public static void main(String args[]) {
15         Scanner input = new Scanner(System.in);
16         System.out.println("Cuenta de ahorros");
17         System.out.println("Ingrese saldo inicial = $");
18         float saldoInicialAhorros = input.nextFloat();
19         System.out.print("Ingrese tasa de interés = ");
20         float tasaAhorros = input.nextFloat();
21         CuentaDeAhorros cuenta1 = new CuentaDeAhorros(saldoInicialAhorros, tasaAhorros);
22         System.out.print("Ingresar cantidad a consignar: $");
23         float cantidadDepositar = input.nextFloat();
24         cuenta1.consignar(cantidadDepositar);
25         System.out.print("Ingresar cantidad a retirar: $");
26         float cantidadRetirar = input.nextFloat();
27         cuenta1.retirar(cantidadRetirar);
28         cuenta1.extractoMensual();
29         cuenta1.imprimir();
30     }
31 }

```



cuenta1: CuentaDeAhorros
+numConsignaciones = 0
+numRetiros = 0
+comisionMensual = 0



```
5 package Inmuebles;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public class Apartaestudio extends Apartamento{
12
13     protected static double valorArea = 1500000;
14
15     public Apartaestudio(int identificadorInmobiliario, int area, String direccion, int numHabitaciones, int numBaños) {
16         super(identificadorInmobiliario, area, direccion, numHabitaciones, numBaños);
17     }
18
19     void Imprimir() {
20         super.Imprimir();
21         System.out.println();
22     }
23
24 }
```

```
5 package Inmuebles;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public class Apartamento extends InmuebleVivienda{
12
13     public Apartamento(int identificadorInmobiliario, int area, String direccion, int numHabitaciones, int numBaños) {
14         super(identificadorInmobiliario, area, direccion, numHabitaciones, numBaños);
15     }
16
17     void Imprimir() {
18         super.Imprimir();
19     }
20
21 }
```

```
> CarreraCiclistica
> Ejercicio4_1
  > Inmuebles
    Apartaestudio.java
    Apartamento.java
    ApartamentoFamiliar.java
    Casa.java
    CasaConjuntoCerrado.java
    CasaIndependiente.java
    CasaRural.java
    CasaUrbana.java
    Inmueble.java
    InmuebleVivienda.java
    Local.java
    LocalComercial.java

5 package Inmuebles;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public class ApartamentoFamiliar extends Apartamento {
12
13     protected static double valorArea = 2000000;
14     protected int valorAdministracion;
15
16     public ApartamentoFamiliar(int identificadorInmobiliario, int area, String direccion, int numHabitaciones, int numBaños, int valorAdministracion) {
17         super(identificadorInmobiliario, area, direccion, numHabitaciones, numBaños);
18         this.valorAdministracion = valorAdministracion;
19     }
20
21     void Imprimir() {
22         super.Imprimir();
23         System.out.println("Valor de la administración = $" + valorAdministracion);
24         System.out.println();
25     }
26
27 }
```

```
> Ejercicio4_1
  > Inmuebles
    Apartaestudio.java
    Apartamento.java
    ApartamentoFamiliar.java
    Casa.java
    CasaConjuntoCerrado.java
    CasaIndependiente.java
    CasaRural.java
    CasaUrbana.java
    Inmueble.java
    InmuebleVivienda.java
    Local.java

5 package Inmuebles;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public class Casa extends InmuebleVivienda {
12
13     protected int numPisos;
14
15     public Casa(int identificadorInmobiliario, int area, String direccion, int numHabitaciones, int numBaños, int numPisos) {
16         super(identificadorInmobiliario, area, direccion, numHabitaciones, numBaños);
17         this.numPisos = numPisos;
18     }
19
20     void Imprimir() {
21         super.Imprimir();
22         System.out.println("Número de pisos = " + numPisos);
23     }
24
25 }
```

```
5 package Inmuebles;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public class CasaConjuntoCerrado extends CasaUrbana {
12
13     protected static double valorArea = 2500000;
14     protected int valorAdministracion;
15     protected boolean tienePiscina;
16     protected boolean tieneCamposDeportivos;
17
18     public CasaConjuntoCerrado(int identificadorInmobiliario, int area, String direccion, int numHabitaciones, int numBaños, int numPisos, int valorAdministracion, boolean t
19         super(identificadorInmobiliario, area, direccion, numHabitaciones, numBaños, numPisos);
20         this.valorAdministracion = valorAdministracion;
21         this.tienePiscina = tienePiscina;
22         this.tieneCamposDeportivos = tieneCamposDeportivos;
23     }
24
25     void Imprimir() {
26         super.Imprimir();
27         System.out.println("Valor de la administración = " + valorAdministracion);
28         System.out.println("¿tiene piscina? = " + tienePiscina);
29         System.out.println("¿tiene campos deportivos? = " + tieneCamposDeportivos);
30     }
31 }
32
33 }
```

```
CasaUrbana {
+000000;

PS;

ficadorInmobiliario, int area, String direccion, int numHabitaciones, int numBaños, int numPisos, int valorAdministracion, boolean tienePiscina, boolean tieneCamposDeportivos) {
area, direccion, numHabitaciones, numBaños, numPisos);
ministracion;

CamposDeportivos;

ministración = " + valorAdministracion);
? = " + tienePiscina);
deportivos? = " + tieneCamposDeportivos);
}
```

```
4
5 package Inmuebles;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public class CasaIndependiente extends CasaUrbana {
12
13     protected static double valorArea = 3000000;
14
15     public CasaIndependiente(int identificadorInmobiliario, int area, String direccion, int numHabitaciones, int numBaños, int numPisos) {
16         super(identificadorInmobiliario, area, direccion, numHabitaciones, numBaños, numPisos);
17     }
18
19     void Imprimir() {
20         super.Imprimir();
21         System.out.println();
22     }
23
24 }
```



```
5 package Inmuebles;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public class CasaRural extends Casa {
12
13     protected static double valorArea = 1500000;
14     protected int distanciaCabera;
15     protected int altitud;
16
17     public CasaRural(int identificadorInmobiliario, int area, String direccion, int numHabitaciones, int numBaños, int numPisos, int distanciaCabera, int altitud) {
18         super(identificadorInmobiliario, area, direccion, numHabitaciones, numBaños, numPisos);
19         this.distanciaCabera = distanciaCabera;
20         this.altitud = altitud;
21     }
22
23     void Imprimir() {
24         super.Imprimir();
25         System.out.println("Distancia la cabecera municipal = " + numHabitaciones + "km.");
26         System.out.println("Altitud sobre el nivel del mar = " + altitud + "metros.");
27         System.out.println();
28     }
29
30 }
```

```
5 package Inmuebles;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public class CasaUrbana extends Casa {
12
13     public CasaUrbana(int identificadorInmobiliario, int area, String direccion, int numHabitaciones, int numBaños, int numPisos) {
14         super(identificadorInmobiliario, area, direccion, numHabitaciones, numBaños, numPisos);
15     }
16
17     void Imprimir() {
18         super.Imprimir();
19     }
20
21 }
```

```
5 package Inmuebles;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public class Inmueble {
12
13     protected int identificadorInmobiliario;
14     protected int area;
15     protected String direccion;
16     protected double precioVenta;
17
18     public Inmueble(int identificadorInmobiliario, int area, String direccion) {
19         this.identificadorInmobiliario = identificadorInmobiliario;
20         this.area = area;
21         this.direccion = direccion;
22     }
23
24     double calcularPrecioVenta(double valorArea) {
25         precioVenta = area * valorArea;
26         return precioVenta;
27     }
28
29     void Imprimir() {
30         System.out.println("Identificador inmobiliario = " +
31             identificadorInmobiliario);
32         System.out.println("Área = " + area);
33         System.out.println("Dirección = " + direccion);
34         System.out.println("Precio de venta = $" + precioVenta);
35     }
36
37 }
```

```
5 package Inmuebles;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public class InmuebleVivienda extends Inmueble {
12
13     protected int numHabitaciones;
14     protected int numBaños;
15
16     public InmuebleVivienda(int identificadorInmobiliario, int area, String direccion, int numHabitaciones, int numBaños) {
17         super(identificadorInmobiliario, area, direccion);
18         this.numHabitaciones = numHabitaciones;
19         this.numBaños = numBaños;
20     }
21
22     void Imprimir() {
23         super.Imprimir();
24         System.out.println("Número de habitaciones = " + numHabitaciones);
25         System.out.println("Número de baños = " + numBaños);
26     }
27
28 }
```

```
5 package Inmuebles;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public class Local extends Inmueble {
12
13     enum tipo {INTERNO, CALLE};
14     protected tipo tipoLocal;
15
16     public Local(int identificadorInmobiliario, int area, String direccion, tipo tipoLocal) {
17         super(identificadorInmobiliario, area, direccion);
18         this.tipoLocal = tipoLocal;
19     }
20
21     void Imprimir() {
22         super.Imprimir();
23         System.out.println("Tipo de local = " + tipoLocal);
24     }
25
26 }
```

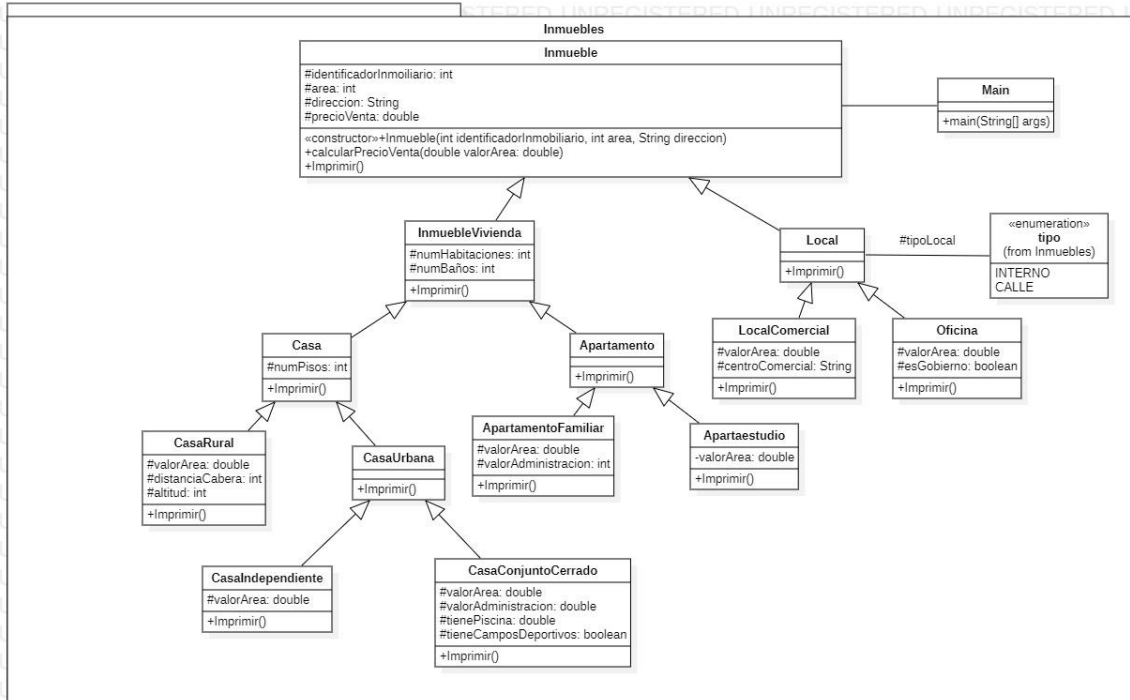
```
5 package Inmuebles;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public class LocalComercial extends Local {
12
13     protected static double valorArea = 3000000;
14     protected String centroComercial;
15
16     public LocalComercial(int identificadorInmobiliario, int area, String direccion, tipo tipoLocal, String centroComercial) {
17         super(identificadorInmobiliario, area, direccion, tipoLocal);
18         this.centroComercial = centroComercial;
19     }
20
21     void Imprimir() {
22         super.Imprimir();
23         System.out.println("Centro comercial = " + centroComercial);
24         System.out.println();
25     }
26
27 }
```

```
> CarreraCiclistica
> Ejercicio4_1
  > Inmuebles
    Apartaestudio.java
    Apartamento.java
    ApartamentoFamiliar.java
    Casa.java
    CasaConjuntoCerrado.java
    CasaIndependiente.java
    CasaRural.java
    CasaUrbana.java
    Inmueble.java
    InmuebleVivienda.java
    Local.java
    LocalComercial.java
    Main.java
    Oficina.java
  > Personas

4  */
5  package Inmuebles;
6
7  /**
8   *
9   * @author USUARIO
10  */
11  public class Oficina extends Local {
12
13      protected static double valorArea = 3500000;
14      protected boolean esGobierno;
15
16      public Oficina(int identificadorInmobiliario, int area, String direccion, tipo tipolocal, boolean esGobierno) {
17          super(identificadorInmobiliario, area, direccion, tipoLocal);
18          this.esGobierno = esGobierno;
19      }
20
21      void Imprimir() {
22          super.Imprimir();
23          System.out.println("Es oficina gubernamental = " + esGobierno);
24          System.out.println();
25      }
26
27  }
```

```
> CarreraCiclistica
> Ejercicio4_1
  > Inmuebles
    Apartaestudio.java
    Apartamento.java
    ApartamentoFamiliar.java
    Casa.java
    CasaConjuntoCerrado.java
    CasaIndependiente.java
    CasaRural.java
    CasaUrbana.java
    Inmueble.java
    InmuebleVivienda.java
    Local.java
    LocalComercial.java
    Main.java
  > Personas

4  */
5  package Inmuebles;
6
7  /**
8   *
9   * @author USUARIO
10  */
11  public class Main {
12
13      public static void main(String[] args) {
14          ApartamentoFamiliar apto1 = new ApartamentoFamiliar(103067,120,"Avenida Santander 45-45",3,2,200000);
15          System.out.println("Datos apartamento");
16          apto1.calcularPrecioVenta(aptol.valorArea);
17          apto1.Imprimir();
18
19          System.out.println("Datos apartament");
20          Apartaestudio aptestudio1 = new Apartaestudio(123554,50,"Avenida Caracas 30-15",1,1);
21          aptestudio1.calcularPrecioVenta(aptestudio1.valorArea);
22          aptestudio1.Imprimir();
23      }
24
25  }
```

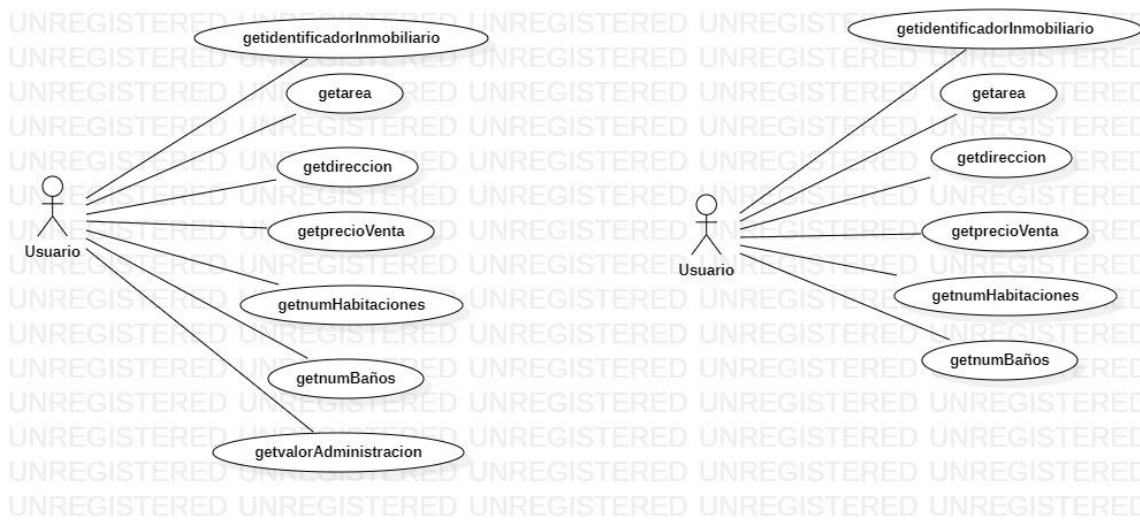


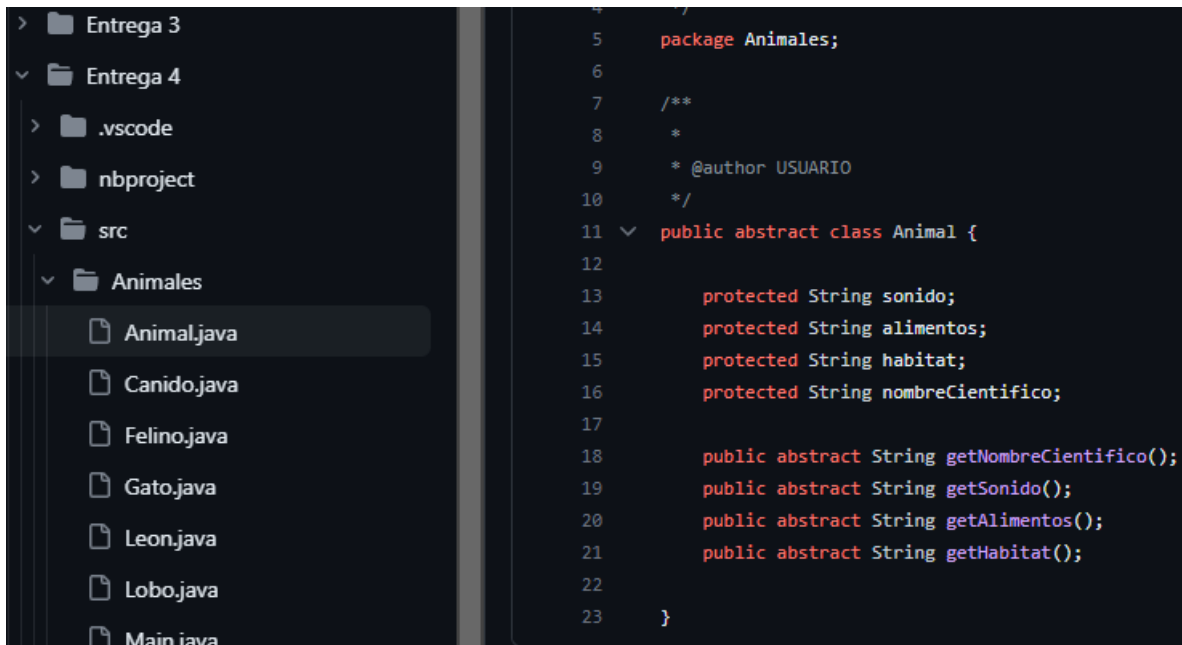
apto1: ApartamentoFamiliar

+identificadorInmobiliario = 103067
 +area = 120
 +direccion = "Avenida Santander 45-45"
 +numHabitaciones = 3
 +numBaños = 2
 +valorAdministracion = 200000

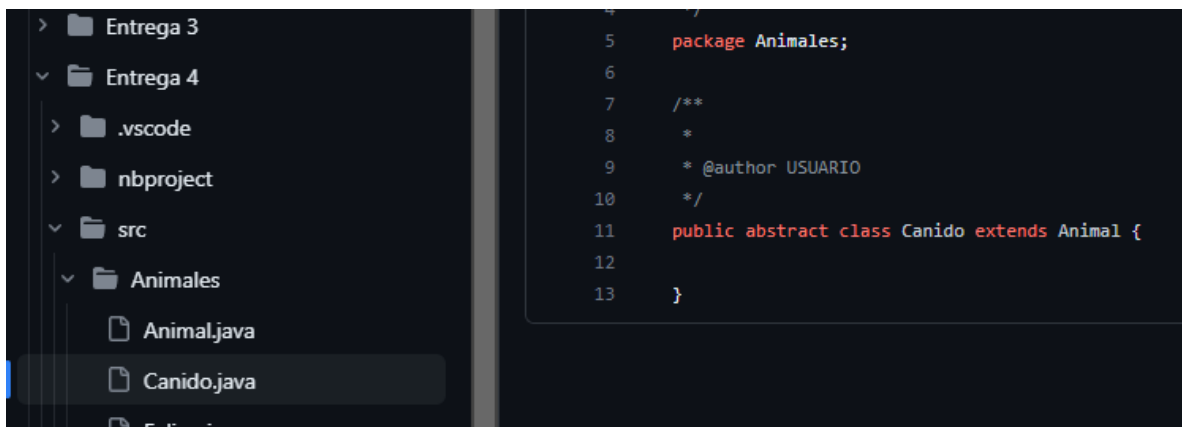
aptestudio1: Apartaestudio

+identificadorInmobiliario = 12354
 +area = 50
 +direccion = "Avenida Caracas 30-15"
 +numHabitaciones = 1
 +numBaños = 2





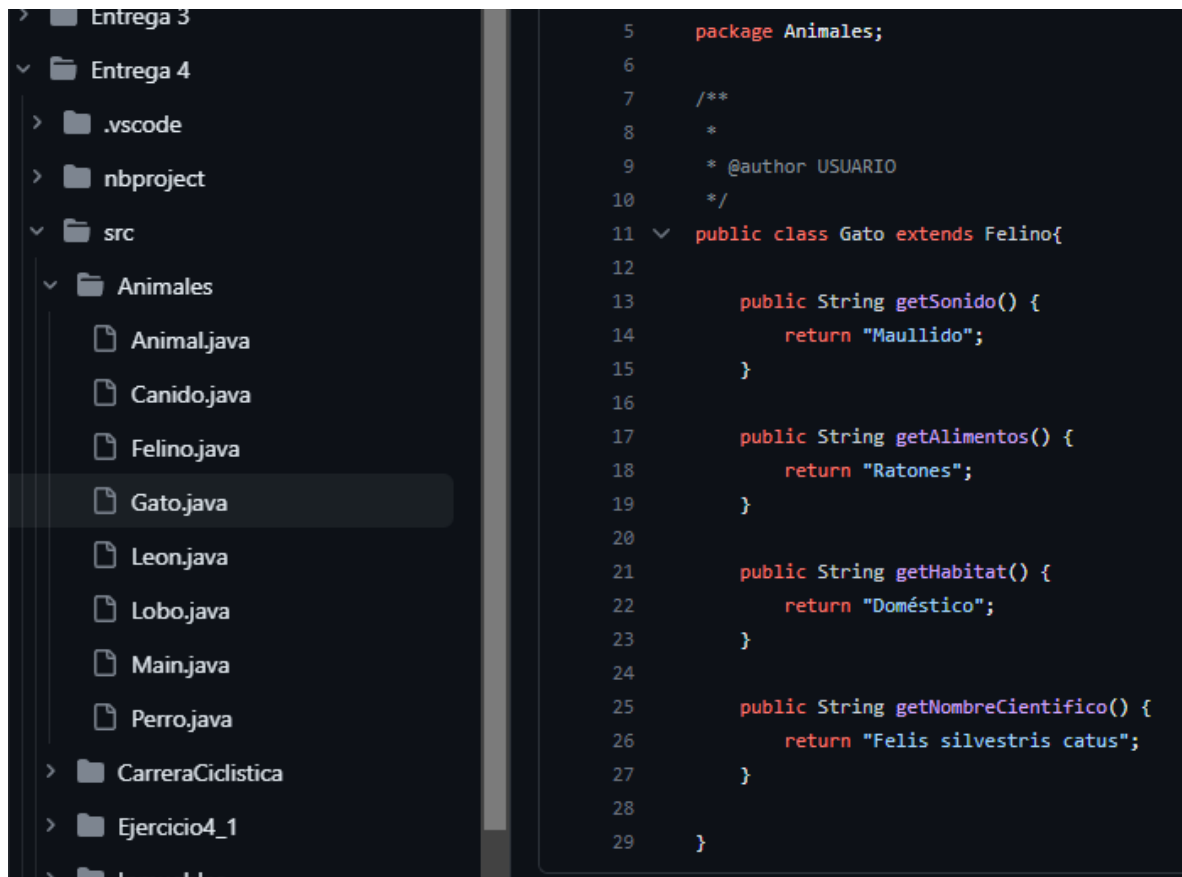
```
4 //
5 package Animales;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public abstract class Animal {
12
13     protected String sonido;
14     protected String alimentos;
15     protected String habitat;
16     protected String nombreCientifico;
17
18     public abstract String getNombreCientifico();
19     public abstract String getSonido();
20     public abstract String getAlimentos();
21     public abstract String getHabitat();
22
23 }
```

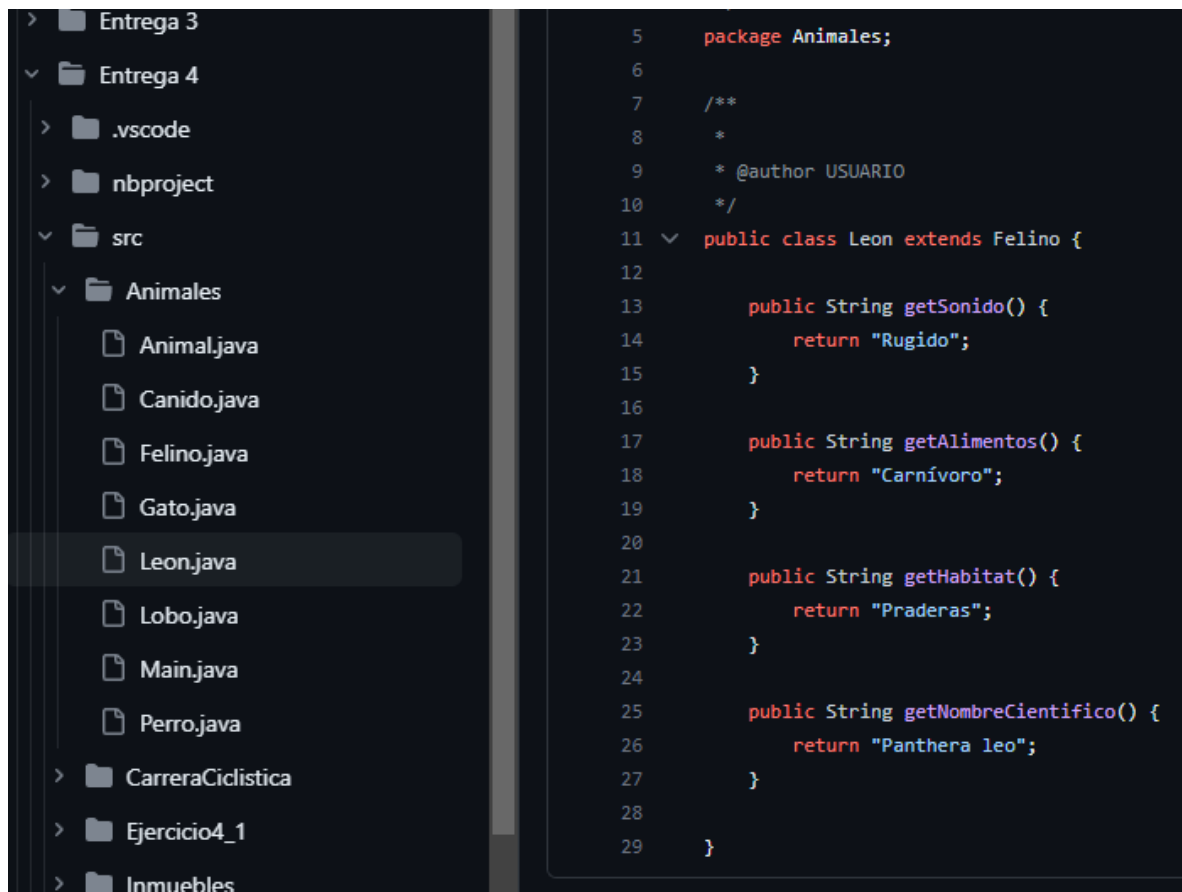


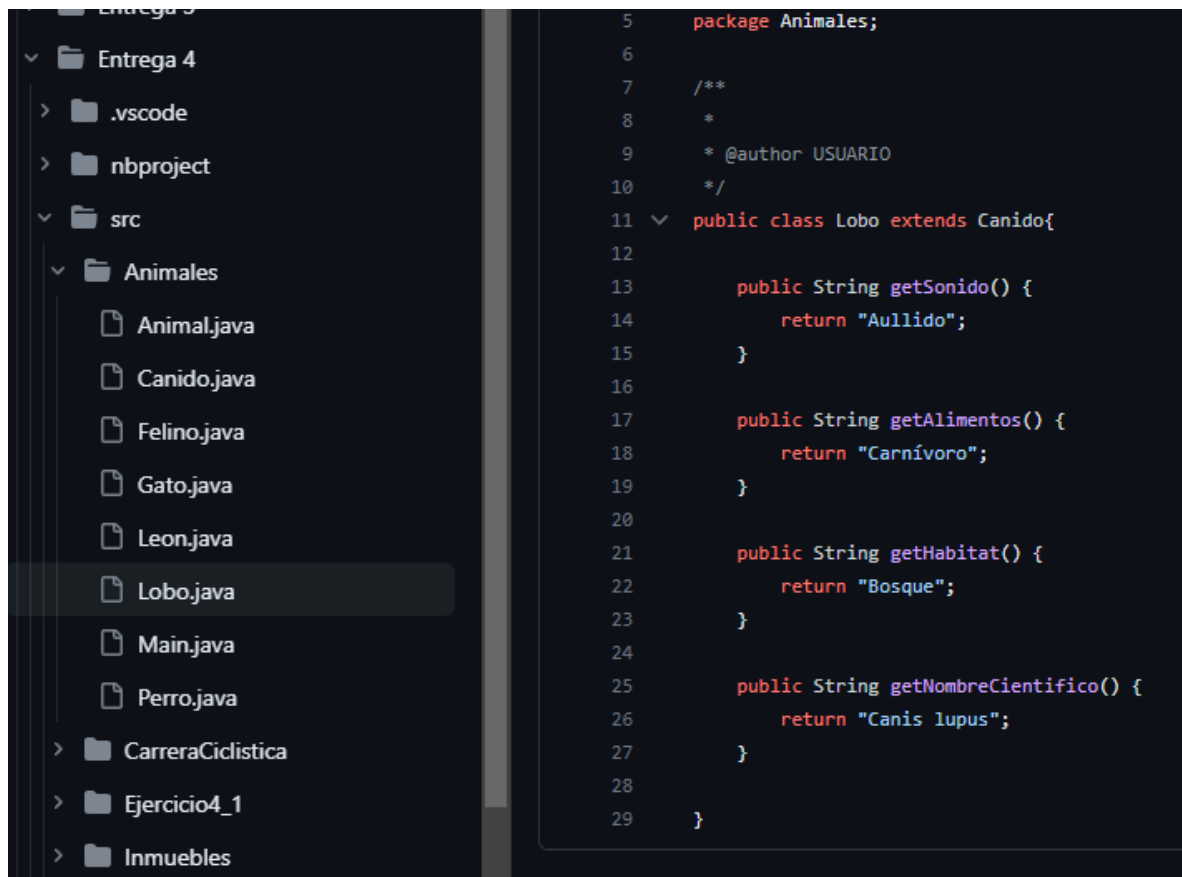
```
4 //
5 package Animales;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public abstract class Canido extends Animal {
12
13 }
```

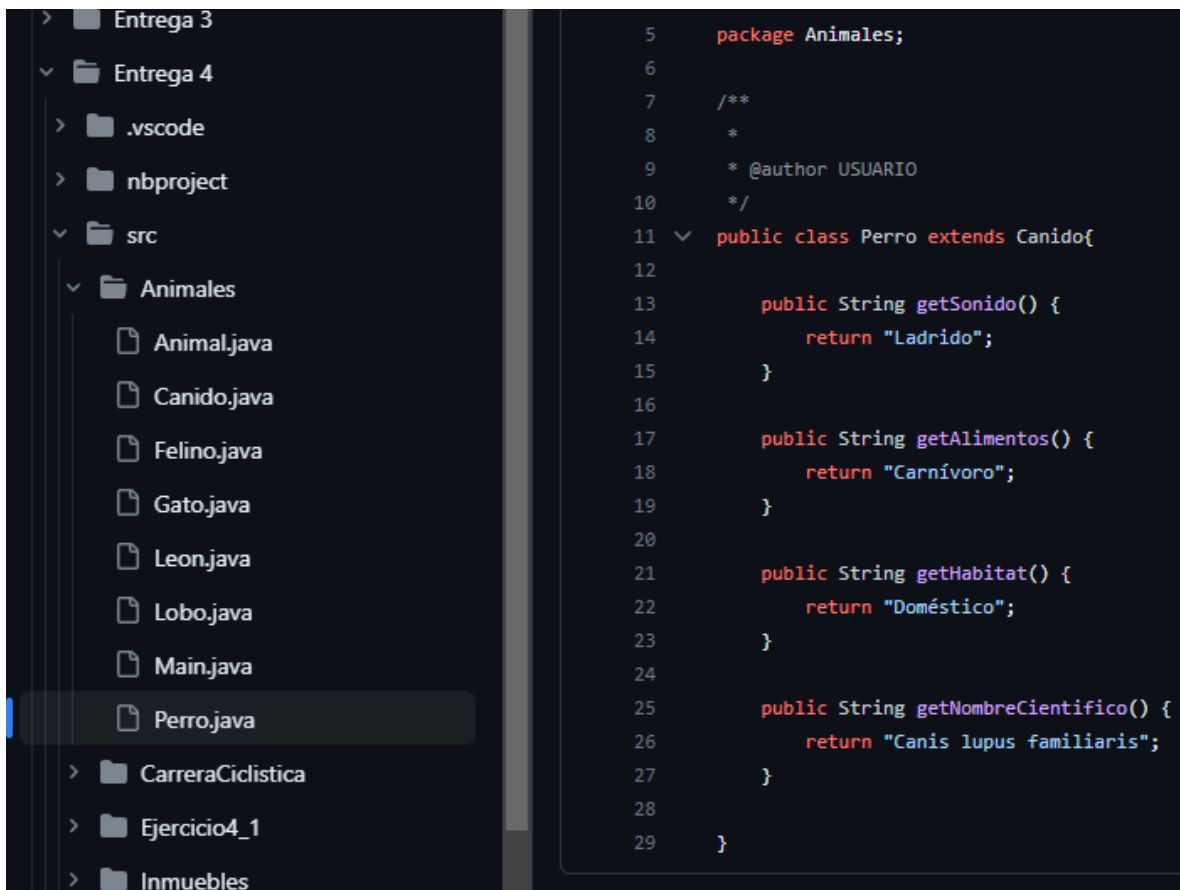


```
4 //
5 package Animales;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public abstract class Felino extends Animal {
12
13 }
```






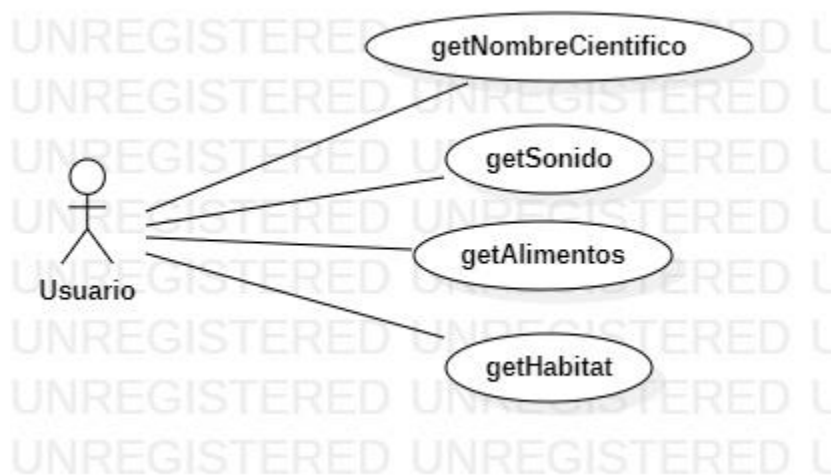
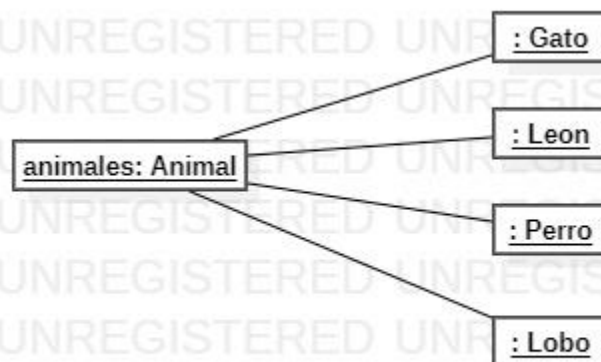
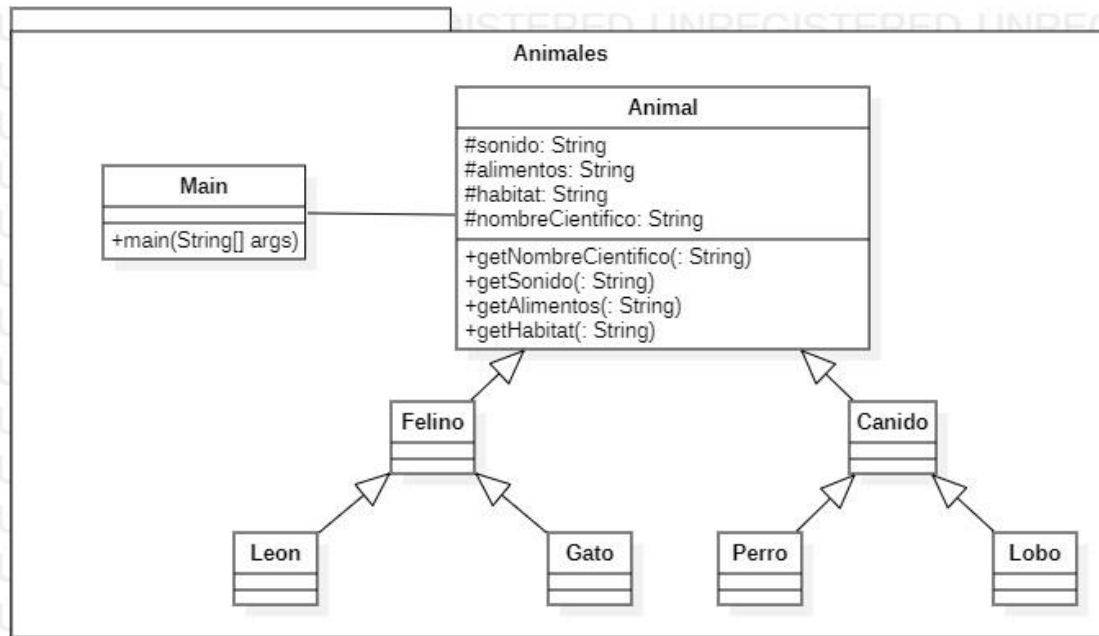


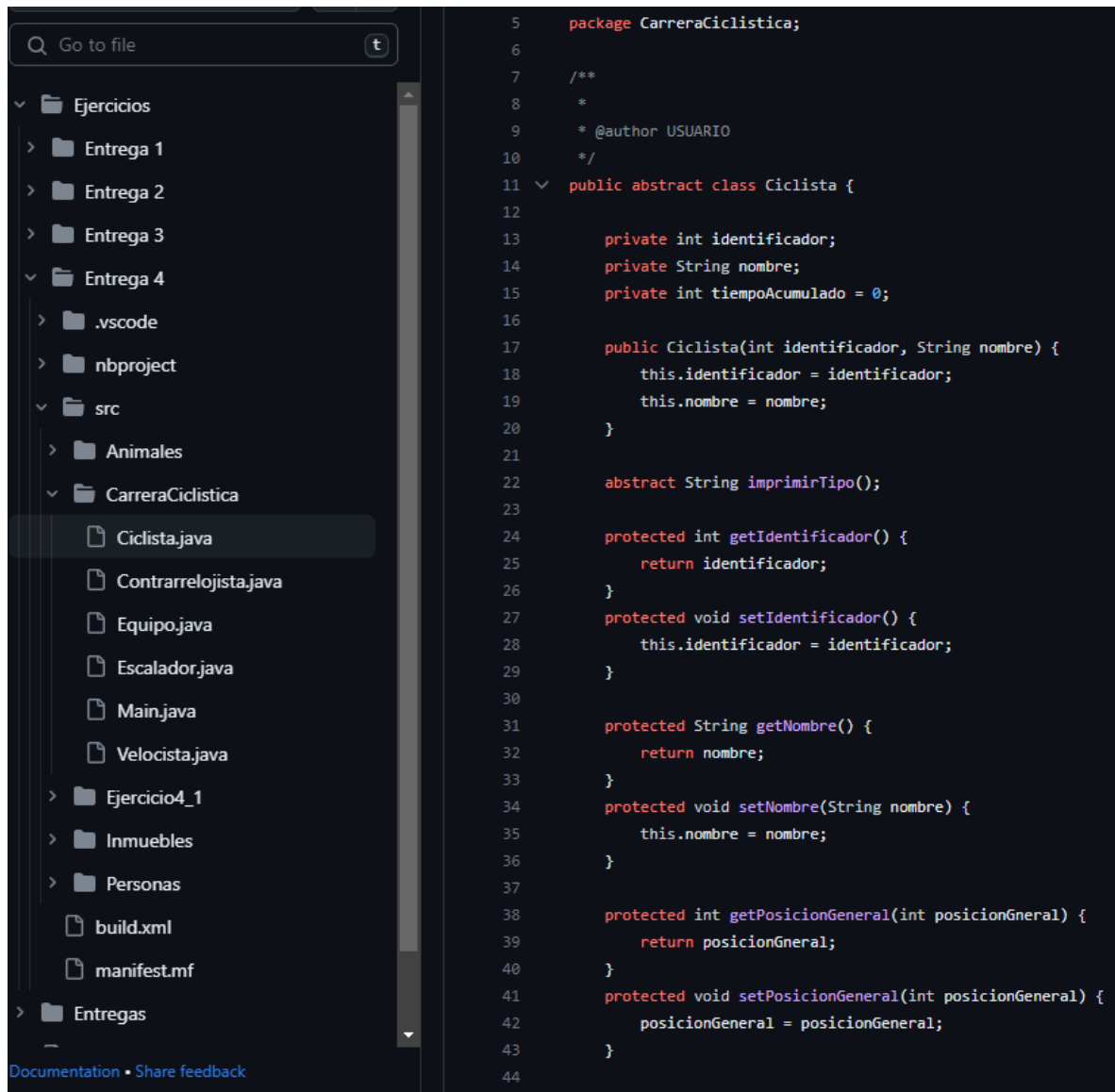


```
5 package Animales;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public class Perro extends Canido{
12
13     public String getSonido() {
14         return "Ladrado";
15     }
16
17     public String getAlimentos() {
18         return "Carnívoro";
19     }
20
21     public String getHabitat() {
22         return "Doméstico";
23     }
24
25     public String getNombreCientifico() {
26         return "Canis lupus familiaris";
27     }
28
29 }
```



```
5 package Animales;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public class Main {
12
13     public static void main(String[] args) {
14         Animal[] animales = new Animal[4];
15         animales[0] = new Gato();
16         animales[1] = new Perro();
17         animales[2] = new Lobo();
18         animales[3] = new Leon();
19
20         for (int i = 0; i < animales.length; i++) {
21             System.out.println(animales[i].getNombreCientifico());
22             System.out.println("Sonido: " + animales[i].getSonido());
23             System.out.println("Alimentos: " + animales[i].getAlimentos());
24             System.out.println("Hábitat: " + animales[i].getHabitat());
25             System.out.println();
26         }
27     }
28
29 }
```

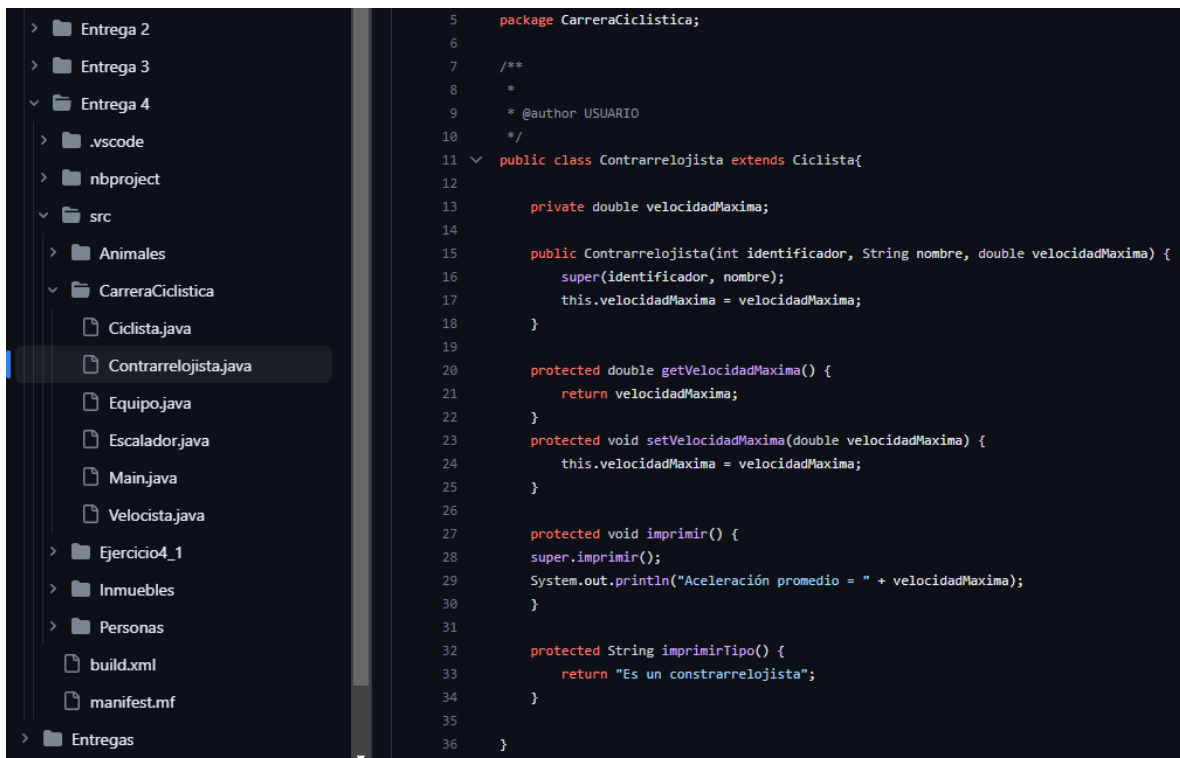




```

45     protected int getTiempoAcumulado() {
46         return tiempoAcumulado;
47     }
48     protected void setTiempoAcumulado(int tiempoAcumulado) {
49         this.tiempoAcumulado = tiempoAcumulado;
50     }
51
52     protected void imprimir() {
53         System.out.println("Identificador = " + identificador);
54         System.out.println("Nombre = " + nombre);
55         System.out.println("Tiempo Acumulado = " + tiempoAcumulado);
56     }
57
58 }

```

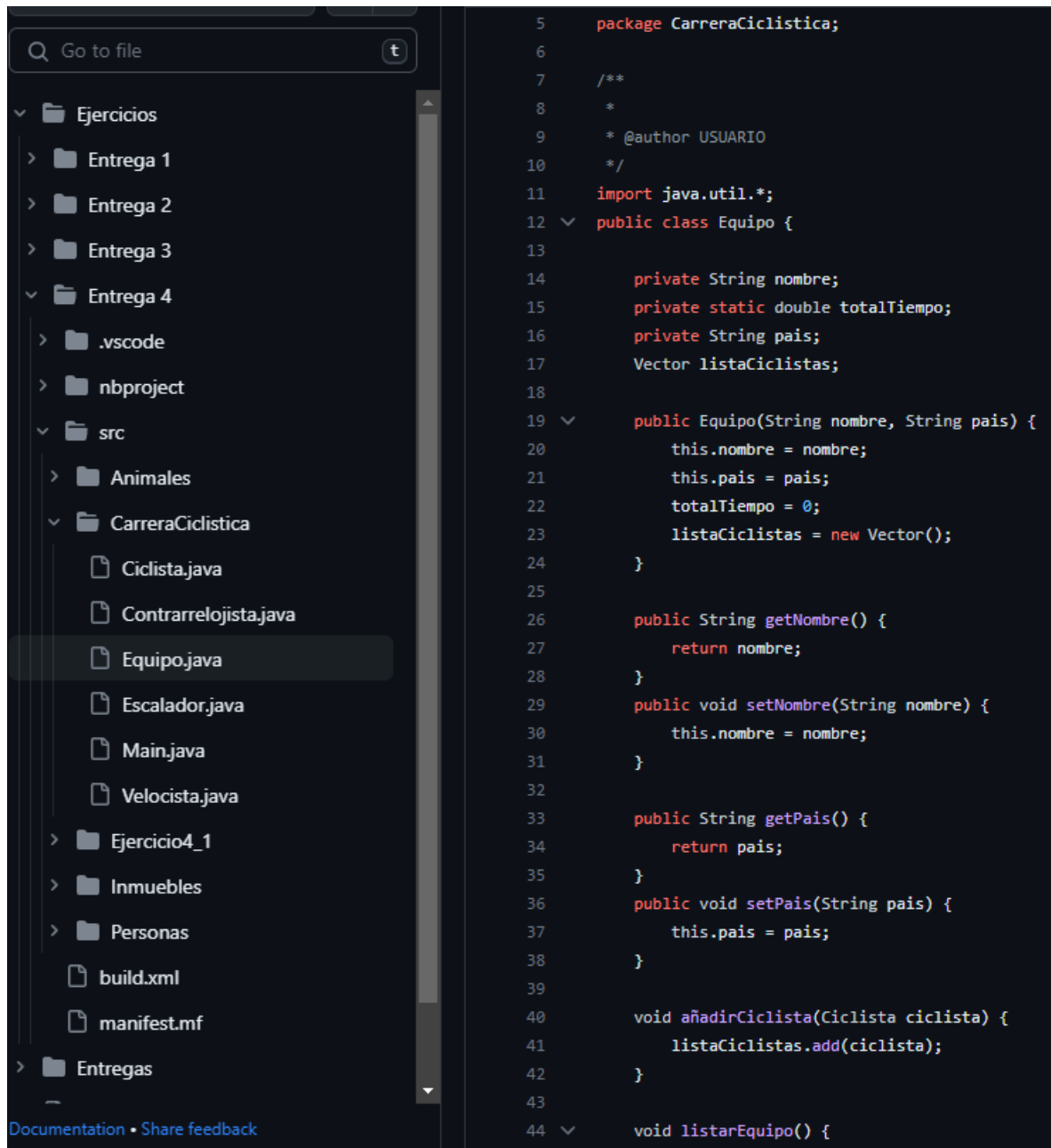


The screenshot shows a code editor with a project structure on the left and Java code on the right. The project structure includes folders for 'Entrega 2', 'Entrega 3', 'Entrega 4', and 'Entregas', along with various source files like 'Ciclista.java', 'Contrarrelojista.java', 'Equipo.java', 'Escalador.java', 'Main.java', and 'Velocista.java'. The 'Contrarrelojista.java' file is selected and its code is displayed in the editor.

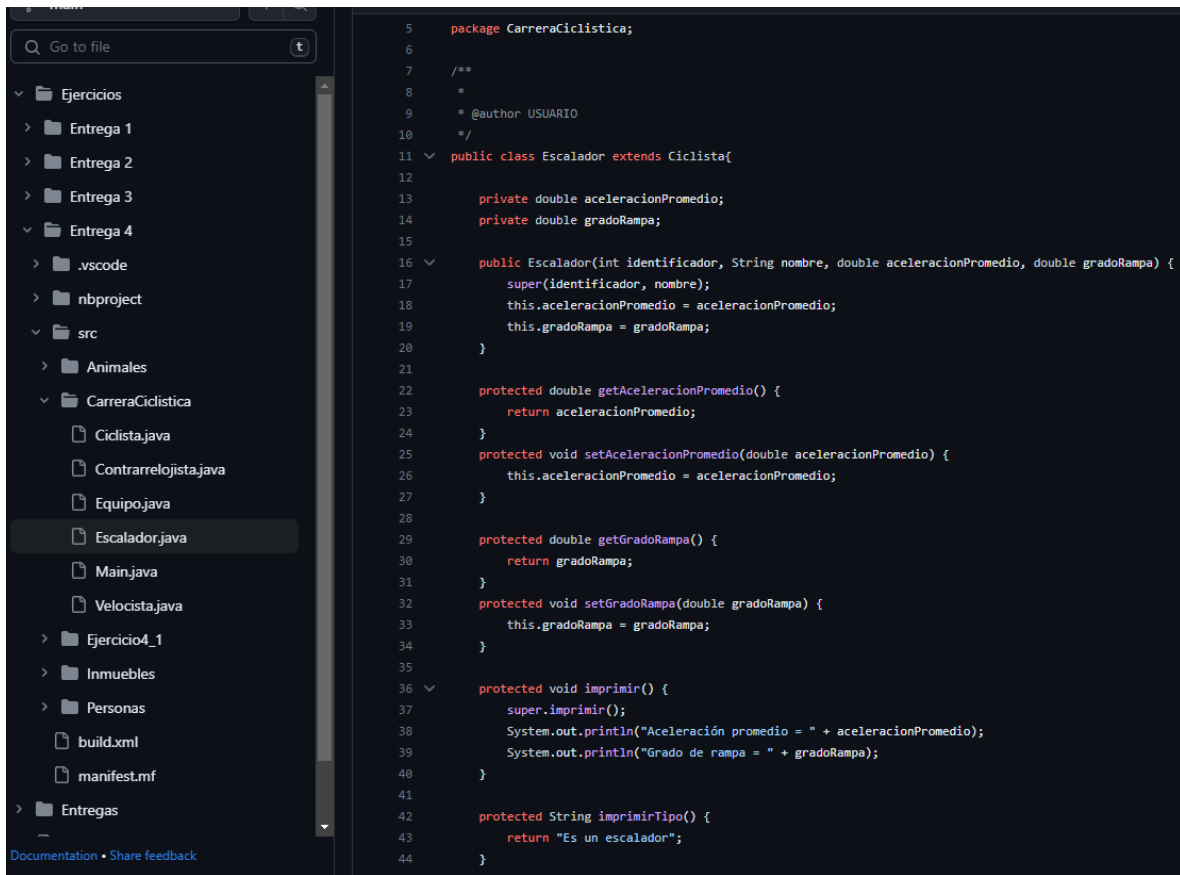
```

5     package CarreraCiclistica;
6
7     /**
8      *
9      * @author USUARIO
10    */
11    public class Contrarrelojista extends Ciclista{
12
13        private double velocidadMaxima;
14
15        public Contrarrelojista(int identificador, String nombre, double velocidadMaxima) {
16            super(identificador, nombre);
17            this.velocidadMaxima = velocidadMaxima;
18        }
19
20        protected double getVelocidadMaxima() {
21            return velocidadMaxima;
22        }
23        protected void setVelocidadMaxima(double velocidadMaxima) {
24            this.velocidadMaxima = velocidadMaxima;
25        }
26
27        protected void imprimir() {
28            super.imprimir();
29            System.out.println("Aceleración promedio = " + velocidadMaxima);
30        }
31
32        protected String imprimirTipo() {
33            return "Es un constrarrelojista";
34        }
35
36    }

```



```
45         for (int i = 0; i < listaCiclistas.size(); i++) {
46             Ciclista c = (Ciclista) listaCiclistas.elementAt(i);
47             System.out.println(c.getNombre());
48         }
49     }
50
51     void buscarCiclista() {
52         Scanner sc = new Scanner(System.in);
53         String nombreCiclista = sc.next();
54         for (int i = 0; i < listaCiclistas.size(); i++) {
55             Ciclista c = (Ciclista) listaCiclistas.elementAt(i);
56             if (c.getNombre().equals(nombreCiclista)) {
57                 System.out.println(c.getNombre());
58             }
59         }
60     }
61
62     void calcularTotalTiempo() {
63         for (int i = 0; i < listaCiclistas.size(); i++) {
64             Ciclista c = (Ciclista) listaCiclistas.elementAt(i);
65             totalTiempo += c.getTiempoAcumulado();
66         }
67     }
68
69     void imprimir() {
70         System.out.println("Nombre del equipo = " + nombre);
71         System.out.println("País = " + pais);
72         System.out.println("Total tiempo del equipo = " + totalTiempo);
73     }
74
75 }
```

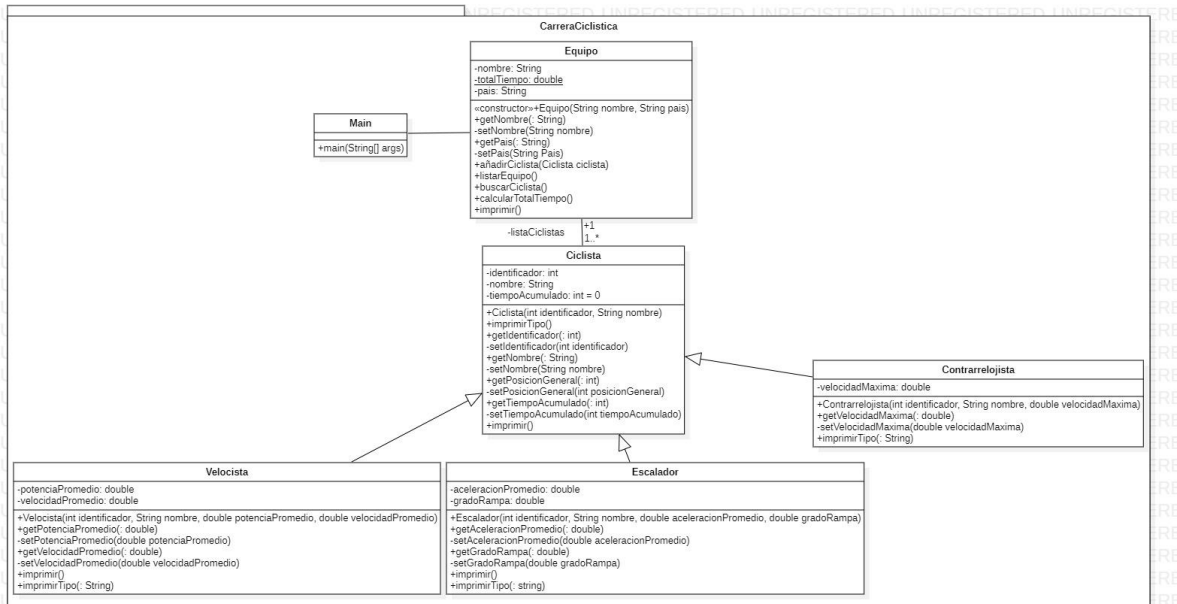


```
45
46 }
```

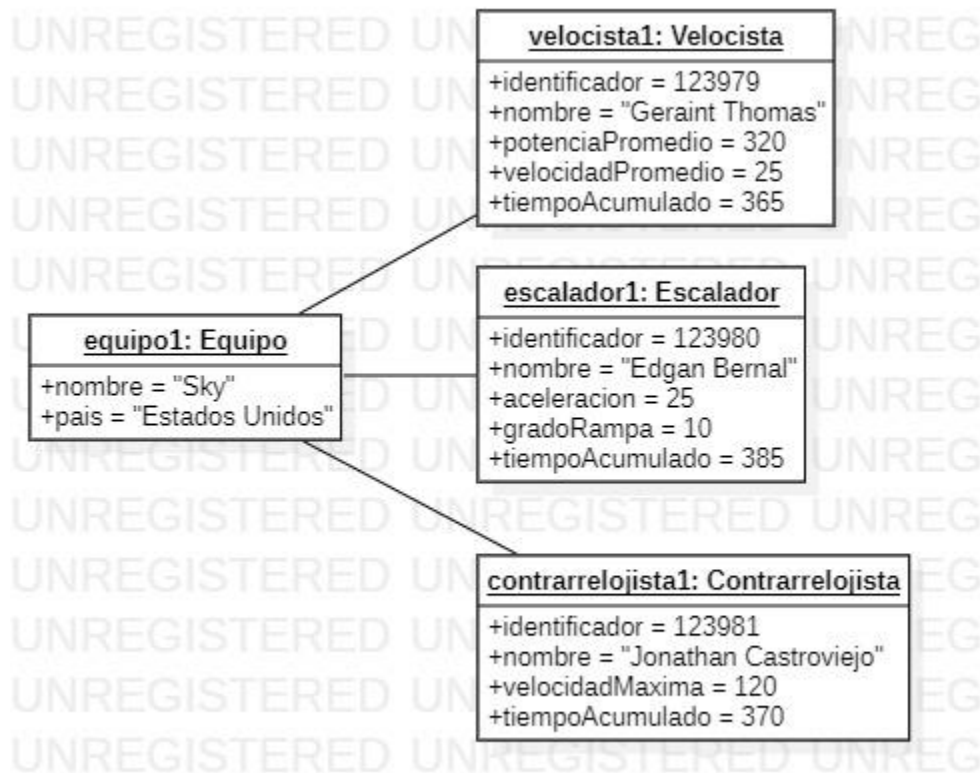
```
5 package CarreraCiclistica;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public class Velocista extends Ciclista{
12
13     private double potenciaPromedio;
14     private double velocidadPromedio;
15
16     public Velocista(int identificador, String nombre, double potenciaPromedio, double velocidadPromedio) {
17         super(identificador, nombre);
18         potenciaPromedio = potenciaPromedio;
19         this.velocidadPromedio = velocidadPromedio;
20     }
21
22     protected double getPotenciaPromedio() {
23         return potenciaPromedio;
24     }
25     protected void setPotenciaPromedio(double potenciaPromedio) {
26         this.potenciaPromedio = potenciaPromedio;
27     }
28
29     protected double getVelocidadPromedio() {
30         return velocidadPromedio;
31     }
32     protected void setVelocidadPromedio(double velocidadPromedio) {
33         this.velocidadPromedio = velocidadPromedio;
34     }
35
36     protected void imprimir() {
37         super.imprimir();
38         System.out.println("Potencia promedio = " + potenciaPromedio);
39         System.out.println("Velocidad promedio = " + velocidadPromedio);
40     }
41
42     protected String imprimirTipo() {
43         return "Es un velocista";
44     }
45 }
```

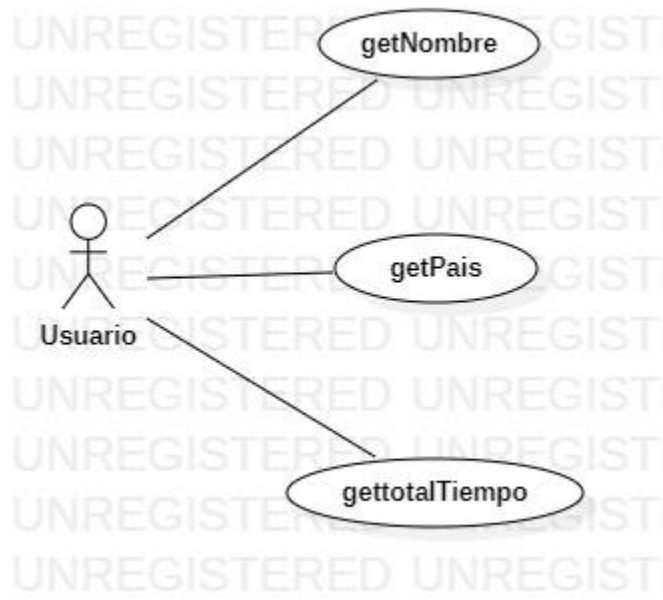
45
46 }

```
5 package CarreraCiclistica;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public class Main {
12     public static void main(String[] args) {
13
14         Equipo equipo1 = new Equipo("Sky", "Estados Unidos");
15         Velocista velocista1 = new Velocista(123970, "Geraint Thomas", 320, 25);
16         Escalador escalador1 = new Escalador(123980, "Edgan Bernal", 25, 10);
17         Contrarrelojista contrarrelojista1 = new Contrarrelojista(123981, "Jonathan Catroviejo", 120);
18
19         equipo1.añadirCiclista(velocista1);
20         equipo1.añadirCiclista(escalador1);
21         equipo1.añadirCiclista(contrarrelojista1);
22         velocista1.setTiempoAcumulado(365);
23         escalador1.setTiempoAcumulado(385);
24         contrarrelojista1.setTiempoAcumulado(370);
25         equipo1.calcularTotalTiempo();
26         equipo1.imprimir();
27         equipo1.listarEquipo();
28     }
29 }
30
31 }
```

UNREGISTERED UNREGISTERED UNREGISTERED UNREGISTERED UNREGISTERED UNREGISTERED UNREGISTERED UNREGISTERED UNREGISTERED





Entrega 3

Entrega 4

.vscode

nbproject

src

Animales

CarreraCiclistica

Ejercicio4_1

Inmuebles

Personas

ListaPersonas.java

Main.java

Persona.java

VentanaPrincipal.java

build.xml

manifest.mf

Entregas

.gitignore

README.md

```
5 package Personas;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 import java.util.*;
12
13 public class ListaPersonas {
14
15     Vector listaPersonas;
16
17     public ListaPersonas() {
18         listaPersonas = new Vector();
19     }
20
21     public void añadirPersona(Persona p) {
22         listaPersonas.add(p);
23     }
24
25     public void eliminarPersona(int i) {
26         listaPersonas.removeElementAt(i);
27     }
28
29     public void borrarLista() {
30         listaPersonas.removeAllElements();
31     }
32
33 }
```

Entrega 3

Entrega 4

.vscode

nbproject

src

Animales

CarreraCiclistica

Ejercicio4_1

Inmuebles

Personas

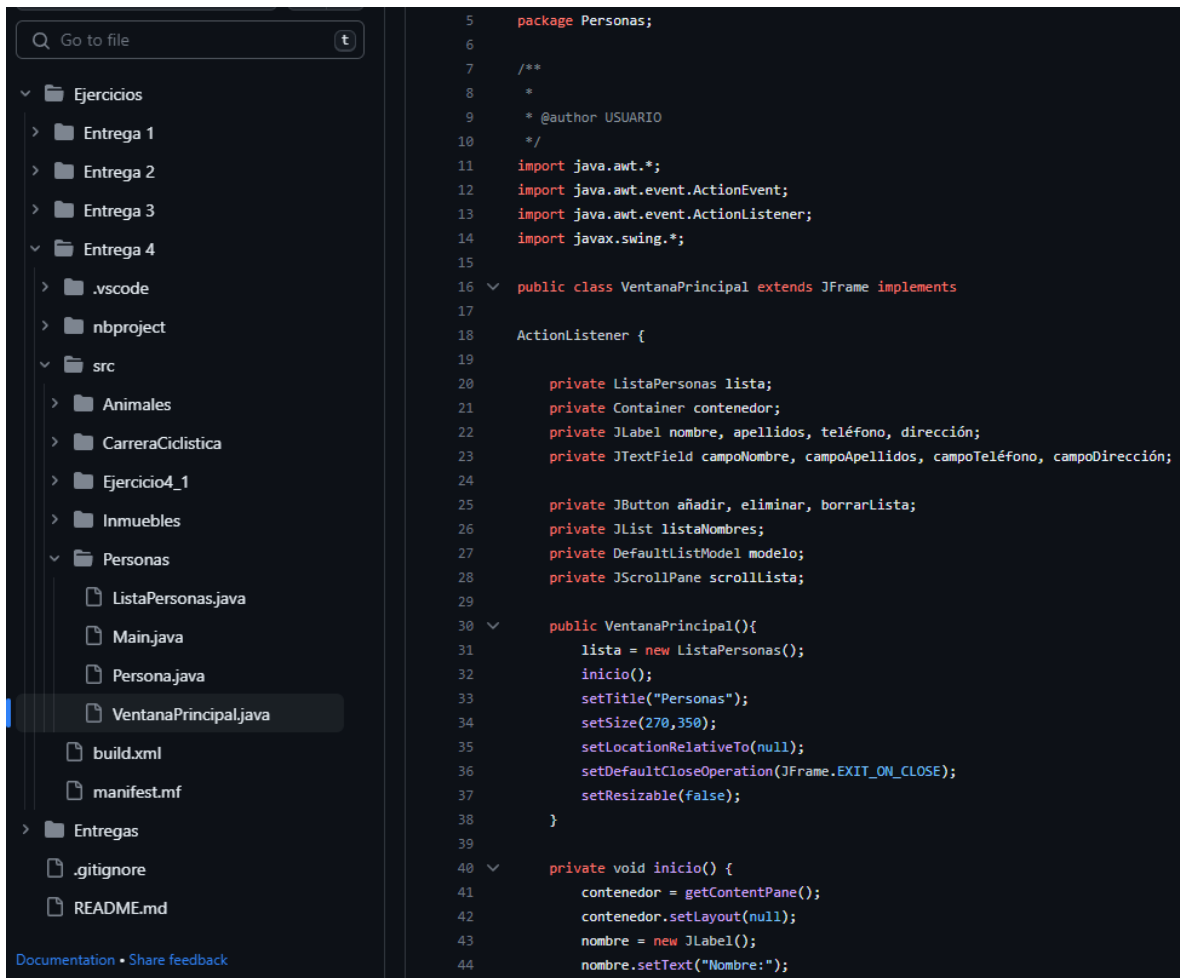
ListaPersonas.java

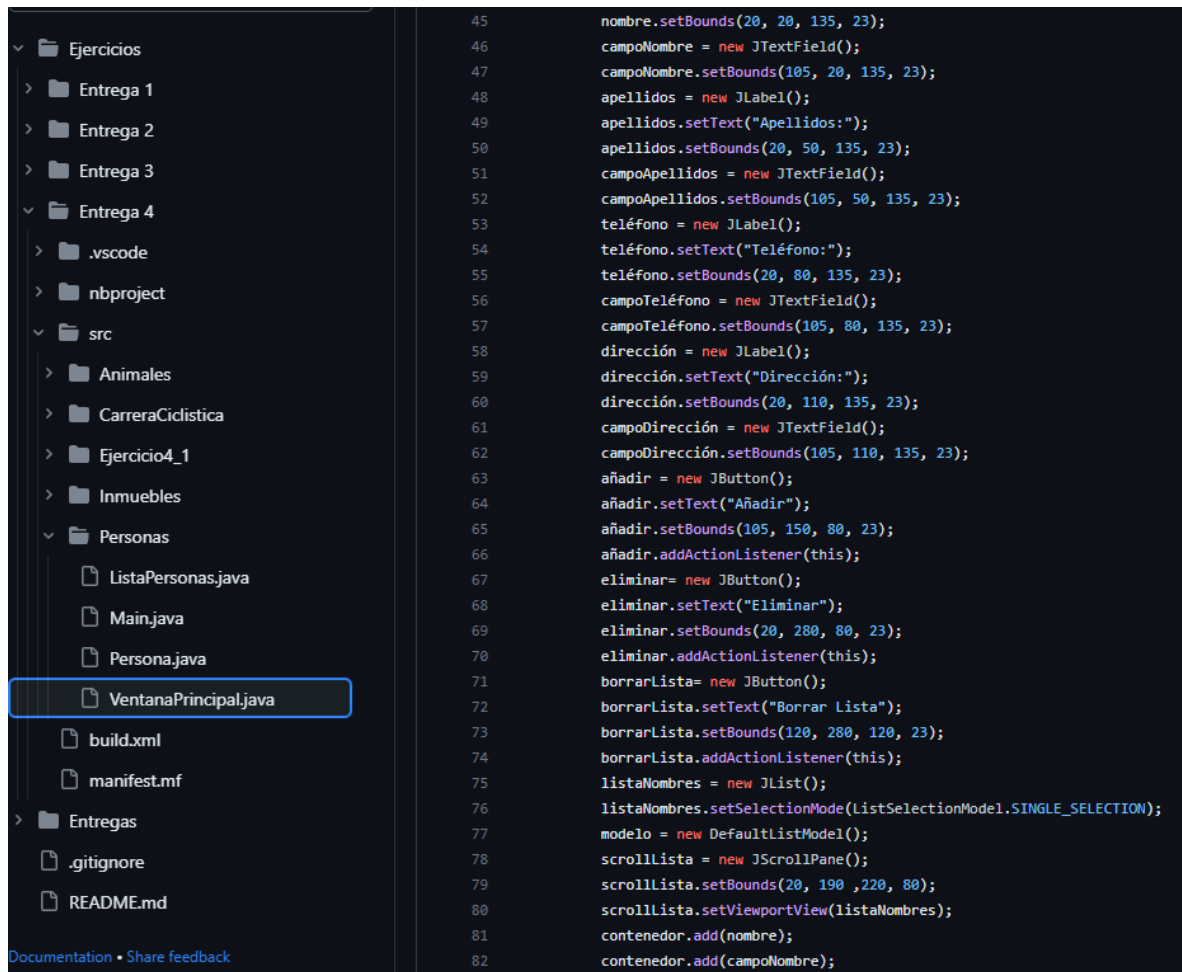
Main.java

Persona.java

VentanaPrincipal.java

```
5 package Personas;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public class Persona {
12
13     String nombre;
14     String apellidos;
15     String telefono;
16     String direccion;
17
18     public Persona(String nombre, String apellidos, String telefono, String direccion) {
19         this.nombre = nombre;
20         this.apellidos = apellidos;
21         this.telefono = telefono;
22         this.direccion = direccion;
23     }
24
25 }
```





```

83         contenedor.add(apellidos);
84         contenedor.add(campoApellidos);
85         contenedor.add(teléfono);
86         contenedor.add(campoTeléfono);
87         contenedor.add(dirección);
88         contenedor.add(campoDirección);
89         contenedor.add(añadir);
90         contenedor.add(eliminar);
91         contenedor.add(borrarLista);
92         contenedor.add(scrollLista);
93     }
94
95     public void actionPerformed(ActionEvent evento) {
96         if (evento.getSource() == añadir) {
97             añadirPersona();
98         }
99         if (evento.getSource() == eliminar) {
100             eliminarNombre(listaNombres.getSelectedIndex() );
101         }
102         if (evento.getSource() == borrarLista) {
103             borrarLista();
104         }
105     }
106
107     private void añadirPersona() {
108
109         Persona p = new Persona(campoNombre.getText(),
110             campoApellidos.getText(),
111             campoTeléfono.getText(), campoDirección.getText());
112         lista.añadirPersona(p);
113         String elemento = campoNombre.getText() + "-" + campoApellidos.getText() + "-" + campoTeléfono.getText() + "-" + campoDirección.getText();
114         modelo.addElement(elemento);
115         listaNombres.setModel(modelo);
116         campoNombre.setText("");
117         campoApellidos.setText("");
118         campoTeléfono.setText("");
119         campoDirección.setText("");
120
121     }
122
123     private void eliminarNombre(int indice) {
124         if (indice >= 0) {
125             modelo.removeElementAt(indice);
126             lista.eliminarPersona(indice);
127         } else {
128             JOptionPane.showMessageDialog(null, "Debe seleccionar un elemento","Error", JOptionPane.ERROR_MESSAGE);
129         }
130     }
131
132     private void borrarLista() {
133         lista.borrarLista();
134         modelo.clear();
135     }
136 }

```

Entrega 4

Entrega 4

.vscode

nbproject

src

Animales

CarreraCiclistica

Ejercicio4_1

Inmuebles

Personas

ListaPersonas.java

Main.java

Persona.java

```
5 package Personas;
6
7 /**
8  *
9  * @author USUARIO
10 */
11 public class Main {
12
13     public static void main(String[] args) {
14         VentanaPrincipal miVentanaPrincipal;
15         miVentanaPrincipal = new VentanaPrincipal();
16         miVentanaPrincipal.setVisible(true);
17     }
18
19 }
```

