

Universidad Nacional de Colombia

Tarea4 - Parte1

Autores:

Juan José Jiménez Maya

Programa: Programación Orientada a Objetos

Grupo: 3

Código: PruebaLibros.java

```
package Tareas.Tarea4.Partel.El_Libros;

public class PruebaLibros {

    public static void main(String[] args) {
        EjerciciosPOO ejerciciosPOO = new EjerciciosPOO();
        System.out.println(ejerciciosPOO);

        CienAnosDeSoledad cienAnosDeSoledad = new CienAnosDeSoledad();
        System.out.println(cienAnosDeSoledad);
    }

    /**
     * Clase que representa un libro de ejercicios de programación orientada a objetos.
     */
    public static class EjerciciosPOO extends LibroUNAL {
        public EjerciciosPOO() {
            super("Ejercicios de Programación Orientada a Objetos con Java y UML",
                "Leonardo Bermón Angarita",
                "Programación Orientada a Objetos",
                "Minas");
        }

        @Override
        float getPrecio() {
            return 50000;
        }
    }

    /**
     * Clase que representa la novela "Cien años de soledad" de Gabriel García Márquez.
     */
    public static class CienAnosDeSoledad extends Novela {
        public CienAnosDeSoledad() {
            super("Cien años de soledad",
                "Gabriel García Márquez",
                Genero.REALISTA);
        }

        @Override
        float getPrecio() {
            return 100000;
        }
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E1_Libros/PruebaLibros.java

Código: Libro.java

```
package Tareas.Tarea4.Partel.El_Libros;

public abstract class Libro {

    protected String titulo;
    protected String autor;

    public Libro(String titulo, String autor) {
        this.titulo = titulo;
        this.autor = autor;
    }

    abstract float getPrecio();

    public String getTitulo() {
        return titulo;
    }

    public void setTitulo(String titulo) {
        this.titulo = titulo;
    }

    public String getAutor() {
        return autor;
    }

    public void setAutor(String autor) {
        this.autor = autor;
    }

    @Override
    public String toString() {
        return "Título: " + getTitulo() +
            "\nAutor: " + getAutor() +
            "\nPrecio: " + getPrecio();
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E1_Libros/Libro.java

Código: LibroUNAL.java

```
package Tareas.Tarea4.Partel.El_Libros;

public abstract class LibroUNAL extends LibroDeCurso {
    private String facultad;

    public LibroUNAL(String titulo, String autor, String curso, String facultad) {
        super(titulo, autor, curso);
        this.facultad = facultad;
    }

    public String getFacultad() {
        return facultad;
    }

    public void setFacultad(String facultad) {
        this.facultad = facultad;
    }

    @Override
    public String toString() {
        return super.toString() +
            "\nFacultad: " + getFacultad();
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E1_Libros/LibroUNAL.java

Código: LibroDeCurso.java

```
package Tareas.Tarea4.Partel.El_Libros;

public abstract class LibroDeCurso extends Libro {
    private String curso;

    public LibroDeCurso(String titulo, String autor, String curso) {
        super(titulo, autor);
        this.curso = curso;
    }

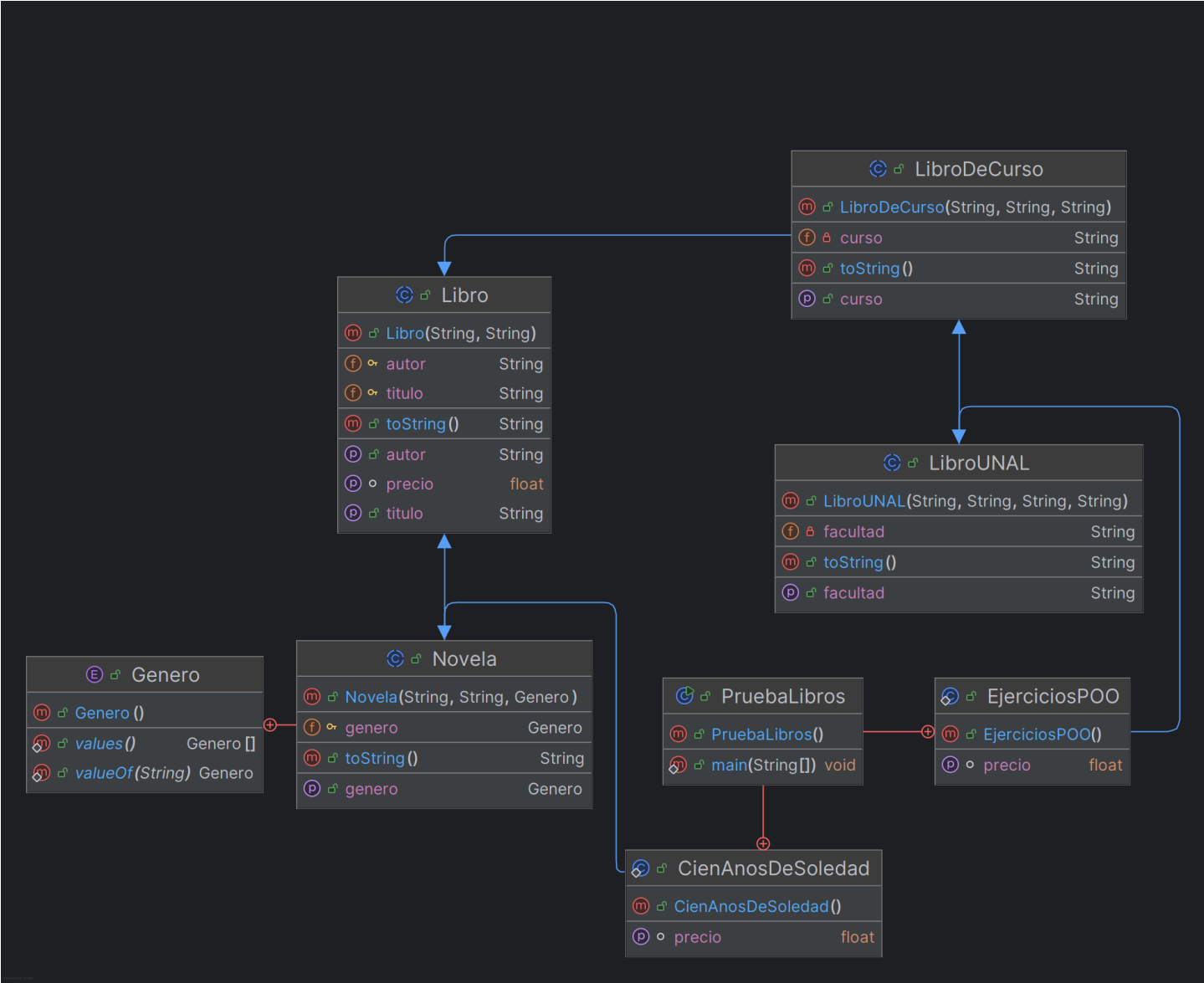
    public String getCurso() {
        return curso;
    }

    public void setCurso(String curso) {
        this.curso = curso;
    }

    @Override
    public String toString() {
        return super.toString() +
            "\nCurso: " + getCurso();
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E1_Libros/LibroDeCurso.java

Imagen: DiagramaUML.png



Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E1_Libros/DiagramaUML.png

Código: Novela.java

```
package Tareas.Tarea4.Partel.El_Libros;

public abstract class Novela extends Libro {
    protected Genero genero;

    public Novela(String titulo, String autor, Genero genero) {
        super(titulo, autor);
        this.genero = genero;
    }

    public Genero getGenero() {
        return genero;
    }

    public void setGenero(Genero genero) {
        this.genero = genero;
    }

    public enum Genero {
        HISTORIA,
        ROMANTICA,
        POLICIACA,
        REALISTA,
        CIENCIA_FICCION,
        AVENTURAS
    }

    @Override
    public String toString() {
        return super.toString() +
            "\nGénero: " + getGenero();
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E1_Libros/Novela.java

Código: Inmueble.java

```
package Tareas.Tarea4.Partel.E2_Inmuebles;

public abstract class Inmueble {

    protected int id;
    protected int area;
    protected String direccion;

    public Inmueble(int id, int area, String direccion) {
        this.id = id;
        this.area = area;
        this.direccion = direccion;
    }

    public int getId() {
        return id;
    }

    public int getArea() {
        return area;
    }

    public String getDireccion() {
        return direccion;
    }

    public abstract double valorArea();

    public double precioVenta() {
        return area * valorArea();
    }

    @Override
    public String toString() {
        return "ID: " + getId() +
            "\nÁrea: " + getArea() +
            "\nDirección: " + getDireccion() +
            "\nPrecio de venta: " + precioVenta();
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E2_Inmuebles/Inmueble.java

Código: Apartaestudio.java

```
package Tareas.Tarea4.Partel.E2_Inmuebles.Vivienda;

public class Apartaestudio extends AbstractApartamento {
    public Apartaestudio(int id, int area, String direccion, int banos) {
        super(id, area, direccion, 1, banos);
    }

    @Override
    public double valorArea() {
        return 1500000;
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E2_Inmuebles/Vivienda/Apartaestudio.java

Código: AbstractCasaUrbana.java

```
package Tareas.Tarea4.Partel.E2_Inmuebles.Vivienda;
```

```
public abstract class AbstractCasaUrbana extends AbstractCasa {  
    public AbstractCasaUrbana(int id, int area, String direccion, int habitaciones, int banos, int pisos) {  
        super(id, area, direccion, habitaciones, banos, pisos);  
    }  
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E2_Inmuebles/Vivienda/AbstractCasaUrbana.java

Código: ConjuntoCerrado.java

```
package Tareas.Tarea4.Partel.E2_Inmuebles.Vivienda;

public class ConjuntoCerrado extends AbstractCasaUrbana {
    protected int admin;
    protected boolean comunes;

    public ConjuntoCerrado(int id, int area, String direccion, int habitaciones, int banos,
        int pisos, int admin, boolean comunes) {
        super(id, area, direccion, habitaciones, banos, pisos);
        this.admin = admin;
        this.comunes = comunes;
    }

    public int getAdmin() {
        return admin;
    }

    public boolean isComunes() {
        return comunes;
    }

    @Override
    public double valorArea() {
        return 2500000;
    }

    @Override
    public String toString() {
        return super.toString() +
            "\nAdministración: " + getAdmin() +
            "\nZonas comunes: " + (isComunes() ? "Sí" : "No");
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E2_Inmuebles/Vivienda/ConjuntoCerrado.

Código: ApartamentoFamiliar.java

```
package Tareas.Tarea4.Partel.E2_Inmuebles.Vivienda;

public class ApartamentoFamiliar extends AbstractApartamento{

    protected int admin;
    public ApartamentoFamiliar(int id, int area, String direccion, int habitaciones, int banos, int admin) {
        super(id, area, direccion, habitaciones, banos);
        this.admin = admin;
    }

    public int getAdmin() {
        return admin;
    }

    @Override
    public double valorArea() {
        return 2000000;
    }

    @Override
    public String toString() {
        return super.toString() +
            "\nTipo: Apartamento Familiar" +
            "\nAdministración: " + getAdmin();
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E2_Inmuebles/Vivienda/ApartamentoFamiliar.java

Código: CasaRural.java

```
package Tareas.Tarea4.Partel.E2_Inmuebles.Vivienda;

public class CasaRural extends AbstractCasa {

    protected float distancia;
    protected float altitud;

    public CasaRural(int id, int area, String direccion, int habitaciones, int banos,
        int pisos, float distancia, float altitud) {
        super(id, area, direccion, habitaciones, banos, pisos);

        this.distancia = distancia;
        this.altitud = altitud;
    }

    public float getDistancia() {
        return distancia;
    }

    public float getAltitud() {
        return altitud;
    }

    @Override
    public double valorArea() {
        return 1500000;
    }

    @Override
    public String toString() {
        return super.toString() +
            "\nDistancia: " + getDistancia() +
            "\nAltitud: " + getAltitud();
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E2_Inmuebles/Vivienda/CasaRural.java

Código: AbstractApartamento.java

```
package Tareas.Tarea4.Partel.E2_Inmuebles.Vivienda;

public abstract class AbstractApartamento extends AbstractVivienda {
    public AbstractApartamento(int id, int area, String direccion, int habitaciones, int banos) {
        super(id, area, direccion, habitaciones, banos);
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E2_Inmuebles/Vivienda/AbstractApartamento.java

Código: AbstractCasa.java

```
package Tareas.Tarea4.Partel.E2_Inmuebles.Vivienda;

public abstract class AbstractCasa extends AbstractVivienda {

    protected int pisos;

    public AbstractCasa(int id, int area, String direccion, int habitaciones, int banos, int pisos) {
        super(id, area, direccion, habitaciones, banos);
        this.pisos = pisos;
    }

    public int getPisos() {
        return pisos;
    }

    @Override
    public String toString() {
        return super.toString() +
            "\nPisos: " + getPisos();
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E2_Inmuebles/Vivienda/AbstractCasa.java

Código: AbstractVivienda.java

```
package Tareas.Tarea4.Part1.E2_Inmuebles.Vivienda;

import Tareas.Tarea4.Part1.E2_Inmuebles.Inmuelle;

public abstract class AbstractVivienda extends Inmuelle {

    protected int habitaciones;
    protected int banos;

    public AbstractVivienda(int id, int area, String direccion, int habitaciones, int banos) {
        super(id, area, direccion);
        this.habitaciones = habitaciones;
        this.banos = banos;
    }

    public int getHabitaciones() {
        return habitaciones;
    }

    public int getBanos() {
        return banos;
    }

    @Override
    public String toString() {
        return super.toString() +
            "\nHabitaciones: " + getHabitaciones() +
            "\nBaños: " + getBanos();
    }
}
```

Enlace: [https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E2_Inmuebles/Vivienda/AbstractVivienda.](https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E2_Inmuebles/Vivienda/AbstractVivienda)

Código: Independiente.java

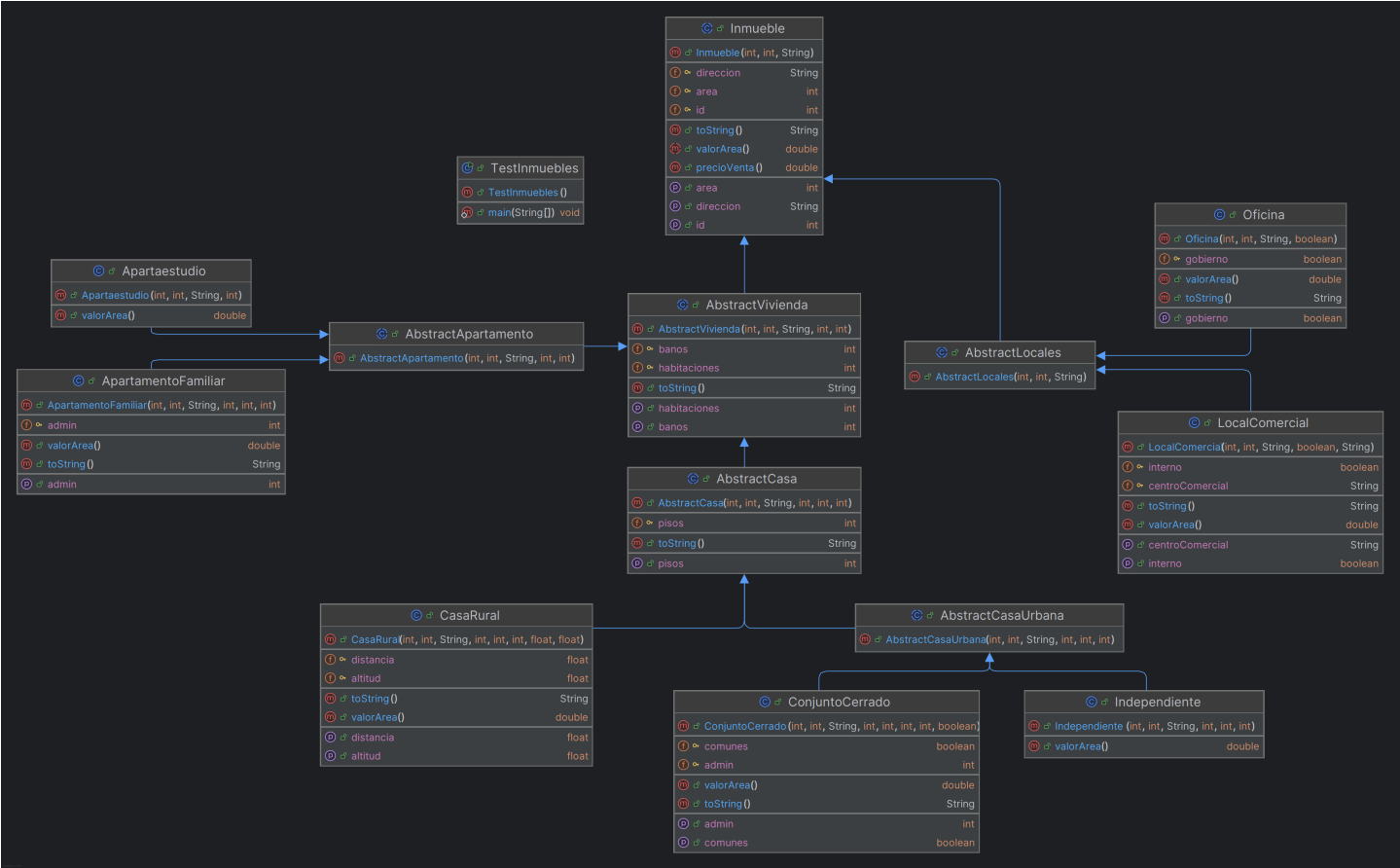
```
package Tareas.Tarea4.Partel.E2_Inmuebles.Vivienda;

public class Independiente extends AbstractCasaUrbana {
    public Independiente(int id, int area, String direccion, int habitaciones, int banos, int pisos) {
        super(id, area, direccion, habitaciones, banos, pisos);
    }

    @Override
    public double valorArea() {
        return 3000000;
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E2_Inmuebles/Vivienda/Independiente.java

Imagen: DiagramaUML.png



Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E2_Inmuebles/DiagramaUML.png

Código: Oficina.java

```
package Tareas.Tarea4.Partel.E2_Inmuebles.Locales;

public class Oficina extends AbstractLocales{
    protected boolean gobierno;

    public Oficina(int id, int area, String direccion, boolean gobierno) {
        super(id, area, direccion);
        this.gobierno = gobierno;
    }

    public boolean isGobierno() {
        return gobierno;
    }

    @Override
    public double valorArea() {
        return 3500000;
    }

    @Override
    public String toString() {
        return super.toString() +
            "\nGobierno: " + (isGobierno() ? "Sí" : "No");
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E2_Inmuebles/Locales/Oficina.java

Código: AbstractLocales.java

```
package Tareas.Tarea4.Partel.E2_Inmuebles.Locales;

import Tareas.Tarea4.Partel.E2_Inmuebles.Inmueble;

public abstract class AbstractLocales extends Inmueble {
    public AbstractLocales(int id, int area, String direccion) {
        super(id, area, direccion);
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E2_Inmuebles/Locales/AbstractLocales.java

Código: LocalComercial.java

```
package Tareas.Tarea4.Partel.E2_Inmuebles.Locales;

public class LocalComercial extends AbstractLocales{
    protected boolean interno;
    protected String centroComercial;

    public LocalComercial(int id, int area, String direccion, boolean interno, String centroComercial) {
        super(id, area, direccion);
        this.interno = interno;
        this.centroComercial = centroComercial;
    }

    public boolean isInterno() {
        return interno;
    }

    public String getCentroComercial() {
        return centroComercial;
    }

    @Override
    public double valorArea() {
        return 3000000;
    }

    @Override
    public String toString() {
        return super.toString() +
            "\nInterno: " + (isInterno() ? "Sí" : "No") +
            "\nCentro comercial: " + getCentroComercial();
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E2_Inmuebles/Locales/LocalComercial.java

Código: TestInmuebles.java

```
package Tareas.Tarea4.Partel.E2_Inmuebles;

import Tareas.Tarea4.Partel.E2_Inmuebles.Vivienda.Apartaestudio;
import Tareas.Tarea4.Partel.E2_Inmuebles.Vivienda.ApartamentoFamiliar;

public class TestInmuebles {

    public static void main(String... args) {
        ApartamentoFamiliar apto1 = new ApartamentoFamiliar(103067,120,
            "Avenida Santander 45-45",3,2,200000);
        System.out.println(apto1);

        System.out.println("-----");

        Apartaestudio aptestudio1 = new Apartaestudio(12354,50,"Avenida Caracas 30-15",1);
        System.out.println(aptestudio1);
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E2_Inmuebles/TestInmuebles.java

Código: MainWindow.java

```
package Tareas.Tarea4.Partel.E5_ListaPersonasInterfaz;

import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.util.*;

public class MainWindow extends JFrame implements ActionListener {

    private final Map<UUID, Persona> personas = new HashMap<>();
    private final DefaultListModel<String> listModel = new DefaultListModel<>();
    private JList<String> listView;
    private JTextField txtNombre, txtApellido, txtDireccion, txtTelefono;

    public MainWindow() {
        super("Lista de personas");
        this.setSize(270, 350);
        this.fillContent(this.getContentPane());
        this.setLocationRelativeTo(null);
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        this.setResizable(false);
        this.setVisible(true);
    }

    protected void fillContent(Container container) {
        container.setLayout(null);

        JLabel lblNombre = new JLabel("Nombre:");
        lblNombre.setBounds(10, 10, 80, 25);
        container.add(lblNombre);

        txtNombre = new JTextField();
        txtNombre.setBounds(100, 10, 150, 25);
        container.add(txtNombre);

        JLabel lblApellido = new JLabel("Apellido:");
        lblApellido.setBounds(10, 40, 80, 25);
        container.add(lblApellido);

        txtApellido = new JTextField();
        txtApellido.setBounds(100, 40, 150, 25);
        container.add(txtApellido);

        JLabel lblDireccion = new JLabel("Dirección:");
        lblDireccion.setBounds(10, 70, 80, 25);
        container.add(lblDireccion);

        txtDireccion = new JTextField();
        txtDireccion.setBounds(100, 70, 150, 25);
        container.add(txtDireccion);

        JLabel lblTelefono = new JLabel("Teléfono:");
        lblTelefono.setBounds(10, 100, 80, 25);
        container.add(lblTelefono);

        txtTelefono = new JTextField();
        txtTelefono.setBounds(100, 100, 150, 25);
        container.add(txtTelefono);

        JButton btnAgregar = new JButton("Agregar");
        btnAgregar.setBounds(10, 130, 100, 30);
        btnAgregar.addActionListener(this);
    }
}
```

```

        container.add(btnAgregar);

        JButton btnEliminar = new JButton("Eliminar");
        btnEliminar.setBounds(120, 130, 100, 30);
        btnEliminar.addActionListener(this);
        container.add(btnEliminar);

        JButton btnModificar = new JButton("Modificar");
        btnModificar.setBounds(10, 170, 210, 30);
        btnModificar.addActionListener(this);
        container.add(btnModificar);

        listView = new JList<>(listModel);
        listView.setBounds(10, 210, 230, 100);
        container.add(listView);
    }

    @Override
    public void actionPerformed(ActionEvent e) {
        String command = e.getActionCommand();

        switch (command) {
            case "Agregar" -> {

                BuildPersona();
            }
            case "Eliminar" -> {
                int selectedIndex = listView.getSelectedIndex();
                if (selectedIndex != -1) {
                    String selectedPerson = listView.getSelectedValue();
                    personas.entrySet().removeIf(entry -> entry.getValue().toString().equals(selectedPerson));
                    listModel.remove(selectedIndex);
                }
            }
            case "Modificar" -> {
                int selectedIndex = listView.getSelectedIndex();
                if (selectedIndex != -1) {
                    String selectedPerson = listView.getSelectedValue();
                    UUID selectedId = null;

                    for (Map.Entry<UUID, Persona> entry : personas.entrySet()) {
                        if (entry.getValue().toString().equals(selectedPerson)) {
                            selectedId = entry.getKey();
                            break;
                        }
                    }

                    if (selectedId != null) {
                        Persona persona = personas.get(selectedId);

                        txtNombre.setText(persona.getNombre());
                        txtApellido.setText(persona.getApellido());
                        txtDireccion.setText(persona.getDireccion());
                        txtTelefono.setText(persona.getTelefono());

                        JButton btnActualizar = new JButton("Actualizar");
                        btnActualizar.setBounds(10, 130, 100, 30);
                        btnActualizar.addActionListener(e1 -> {
                            BuildPersona();
                        });
                        this.getContentPane().add(btnActualizar);
                    }
                }
            }
        }
    }
}

```



```

    }

    private void BuildPersona() {
        Persona.Builder builder = new Persona.Builder();
        builder.setNombre(txtNombre.getText())
            .setApellido(txtApellido.getText())
            .setDireccion(txtDireccion.getText())
            .setTelefono(txtTelefono.getText());
        Persona p = builder.build();

        if (p != null) {
            UUID id = UUID.randomUUID();
            personas.put(id, p);
            listModel.addElement(p.toString());
            clearTextFields();
        }
    }

    private void clearTextFields() {
        txtNombre.setText("");
        txtApellido.setText("");
        txtDireccion.setText("");
        txtTelefono.setText("");
    }
}

```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E5_ListaPersonasInterfaz/MainWindow.java

Código: Persona.java

```
package Tareas.Tarea4.Partel.E5_ListaPersonasInterfaz;

public class Persona {

    private String nombre;
    private String apellido;
    private String direccion;
    private String telefono;

    private Persona(String nombre, String apellido, String direccion, String telefono) {
        this.nombre = nombre;
        this.apellido = apellido;
        this.direccion = direccion;
        this.telefono = telefono;
    }

    public String getNombre() {
        return nombre;
    }

    public void setNombre(String nombre) {
        this.nombre = nombre;
    }

    public String getApellido() {
        return apellido;
    }

    public void setApellido(String apellido) {
        this.apellido = apellido;
    }

    public String getDireccion() {
        return direccion;
    }

    public void setDireccion(String direccion) {
        this.direccion = direccion;
    }

    public String getTelefono() {
        return telefono;
    }

    public void setTelefono(String telefono) {
        this.telefono = telefono;
    }

    @Override
    public String toString() {
        return nombre + " " + apellido;
    }

    public static class Builder {
        private String nombre;
        private String apellido;
        private String direccion;
        private String telefono;

        public Builder setNombre(String nombre) {
            this.nombre = nombre;
            return this;
        }
    }
}
```

```

    }

    public Builder setApellido(String apellido) {
        this.apellido = apellido;
        return this;
    }

    public Builder setDireccion(String direccion) {
        this.direccion = direccion;
        return this;
    }

    public Builder setTelefono(String telefono) {
        this.telefono = telefono;
        return this;
    }

    public Persona build() {
        if (nombre == null || apellido == null) {
            return null;
        }
        return new Persona(nombre, apellido, direccion, telefono);
    }
}

```

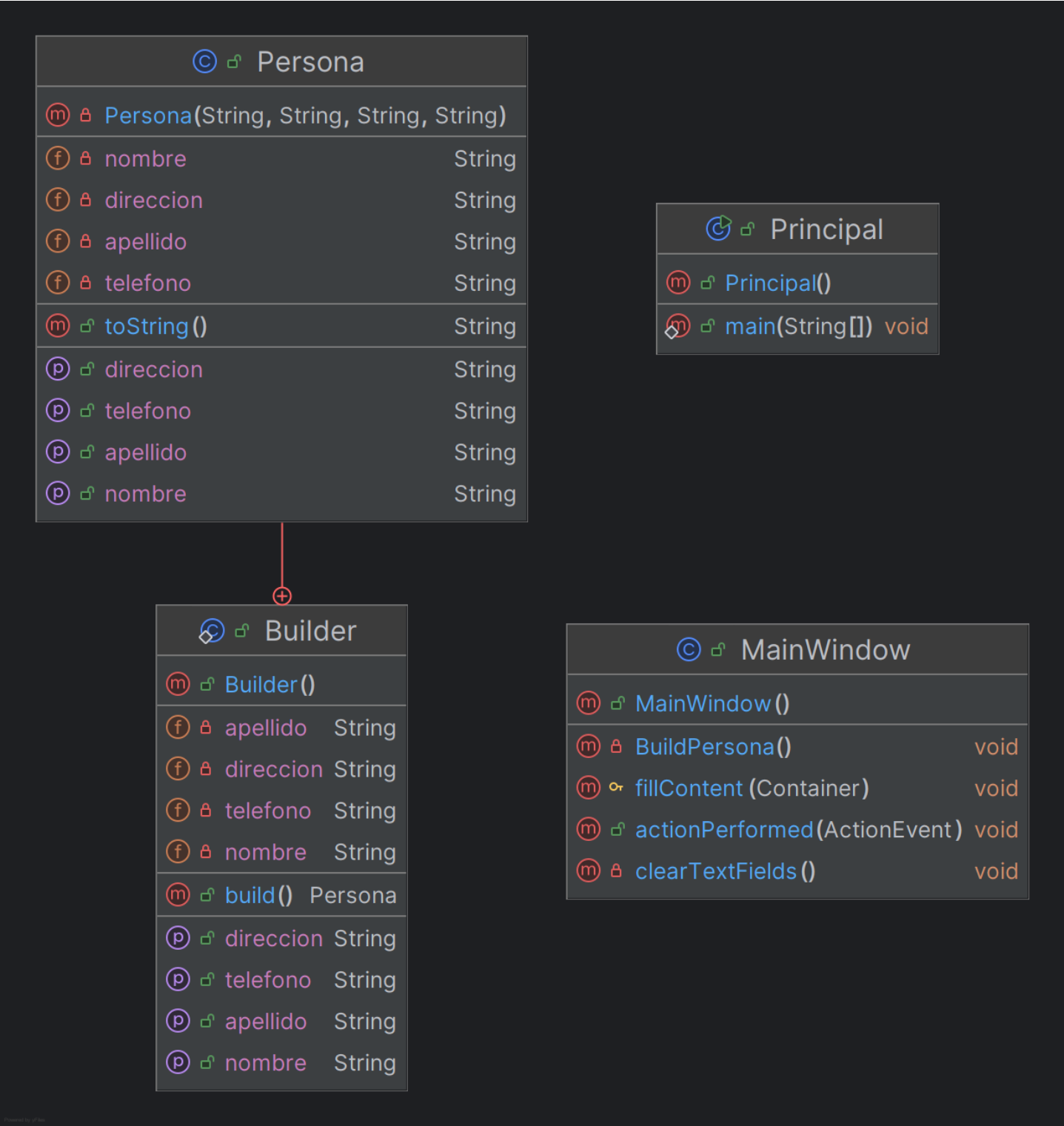
Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E5_ListaPersonasInterfaz/Persona.java

Código: Principal.java

```
package Tareas.Tarea4.Partel.E5_ListaPersonasInterfaz;  
  
public class Principal {  
  
    public static void main(String... args) {  
        MainWindow mainWindow = new MainWindow();  
    }  
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E5_ListaPersonasInterfaz/Principal.java

Imagen: DiagramaUML.png



Enlace: [https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E5_ListaPersonasInterfaz/DiagramaUML.](https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E5_ListaPersonasInterfaz/DiagramaUML)

Código: Felinos.java

```
package Tareas.Tarea4.Partel.E3_Animales;  
  
public abstract class Felinos extends Animal {  
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E3_Animales/Felinos.java

Código: Canidos.java

```
package Tareas.Tarea4.Partel.E3_Animales;  
  
public abstract class Canidos extends Animal{  
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E3_Animales/Canidos.java

Código: PruebaAnimales.java

```
package Tareas.Tarea4.Partel.E3_Animales;

public class PruebaAnimales {

    public static void main(String... args) {
        Animal[] animales = {
            new Leon(),
            new Lobo(),
            new Gato(),
            new Perro()
        };

        for (Animal animal : animales) {
            System.out.println(animal);
        }
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E3_Animales/PruebaAnimales.java

Código: Leon.java

```
package Tareas.Tarea4.Partel.E3_Animales;

public class Leon extends Felinos {
    @Override
    public String getNombreCientifico() {
        return "Panthera leo";
    }

    @Override
    public String getSonido() {
        return "rugido";
    }

    @Override
    public String getAlimentos() {
        return "carnivora";
    }

    @Override
    public String getHabitat() {
        return "pradera";
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E3_Animales/Leon.java

Código: Animal.java

```
package Tareas.Tarea4.Partel.E3_Animales;

public abstract class Animal {

    public abstract String getNombreCientifico();
    public abstract String getSonido();
    public abstract String getAlimentos();
    public abstract String getHabitat();

    @Override
    public String toString() {
        return "Animal{" +
            "nombreCientifico='" + getNombreCientifico() + '\'' +
            ", sonido='" + getSonido() + '\'' +
            ", alimentos='" + getAlimentos() + '\'' +
            ", habitat='" + getHabitat() + '\'' +
            '}'
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E3_Animales/Animal.java

Código: Lobo.java

```
package Tareas.Tarea4.Partel.E3_Animales;

public class Lobo extends Canidos {
    @Override
    public String getNombreCientifico() {
        return "Canis lupus";
    }

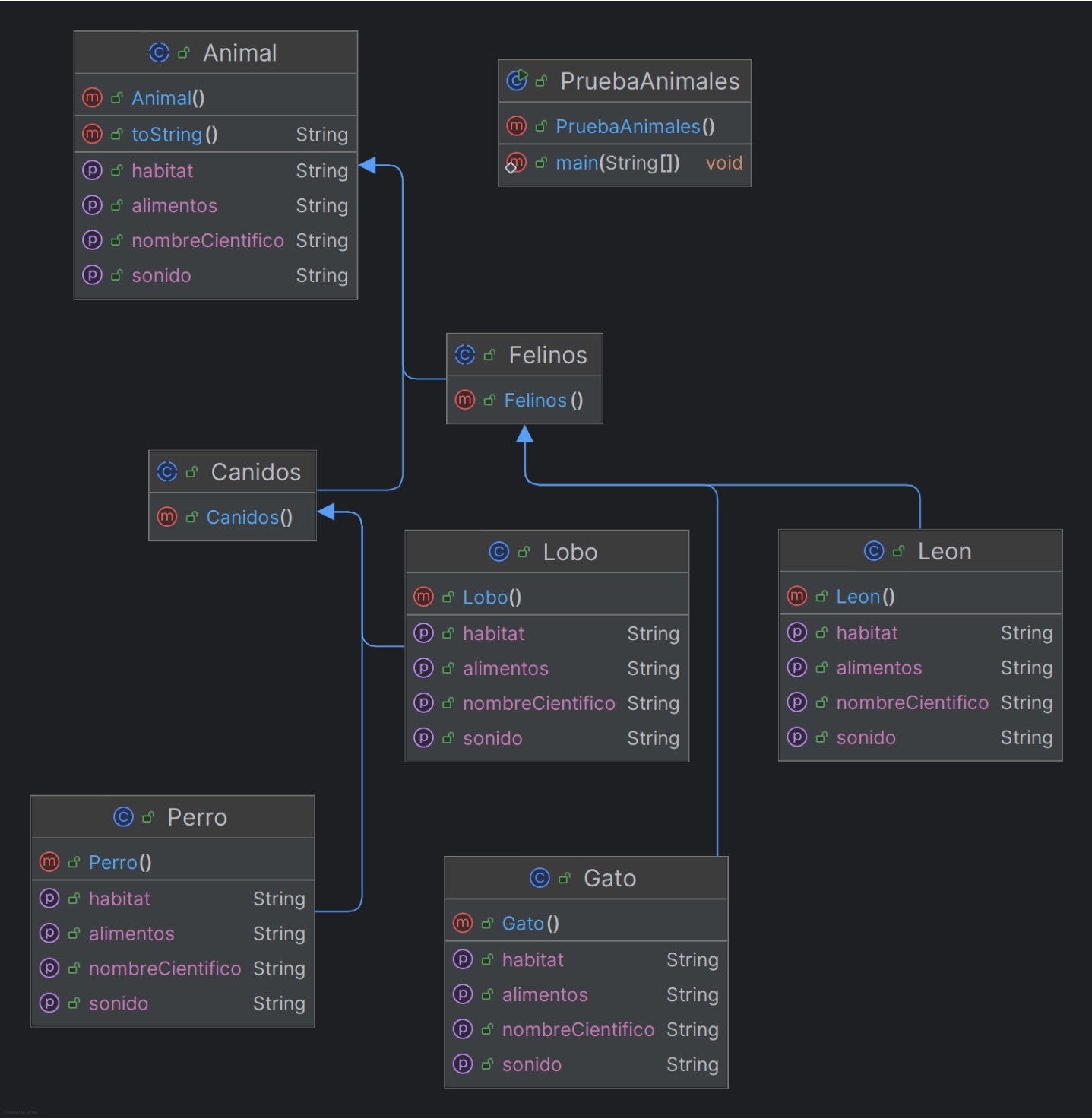
    @Override
    public String getSonido() {
        return "aullido";
    }

    @Override
    public String getAlimentos() {
        return "carnivora";
    }

    @Override
    public String getHabitat() {
        return "bosque";
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E3_Animales/Lobo.java

Imagen: DiagramaUML.png



Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E3_Animales/DiagramaUML.png

Código: Gato.java

```
package Tareas.Tarea4.Partel.E3_Animales;

public class Gato extends Felinos {
    @Override
    public String getNombreCientifico() {
        return "Felis silvestris catus";
    }

    @Override
    public String getSonido() {
        return "maullido";
    }

    @Override
    public String getAlimentos() {
        return "ratones";
    }

    @Override
    public String getHabitat() {
        return "domestico";
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E3_Animales/Gato.java

Código: Perro.java

```
package Tareas.Tarea4.Partel.E3_Animales;

public class Perro extends Canidos {
    @Override
    public String getNombreCientifico() {
        return "Canis lupus familiaris";
    }

    @Override
    public String getSonido() {
        return "ladrido";
    }

    @Override
    public String getAlimentos() {
        return "carnivora";
    }

    @Override
    public String getHabitat() {
        return "domestico";
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E3_Animales/Perro.java

Código: CuentaBancaria.java

```
package Tareas.Tarea4.Partel.El_CuentasBancarias;

public abstract class CuentaBancaria {

    protected float saldo;
    protected int consignaciones;
    protected int retiros;
    protected float tasaAnual;
    protected float comision;

    public CuentaBancaria(float saldo, float tasaAnual) {
        this.saldo = saldo;
        this.consignaciones = 0;
        this.retiros = 0;
        this.tasaAnual = tasaAnual;
        this.comision = 0;
    }

    public void consignar(float cantidad) {
        if (cantidad <= 0) {
            System.out.println("No se puede consignar esa cantidad, el valor debe ser mayor a 0");
            return;
        }
        saldo += cantidad;
        consignaciones++;
    }

    public void retirar(float cantidad) {
        if (cantidad > saldo) {
            System.out.println("No se puede retirar esa cantidad, saldo insuficiente");
            return;
        }
        saldo -= cantidad;
        retiros++;
    }

    public void calcularInteres() {
        saldo += saldo * (tasaAnual / 12);
    }

    public void extractoMensual() {
        saldo -= comision;
        calcularInteres();
    }

    public void mostrarSaldo() {
        System.out.println(this);
    }

    @Override
    public String toString() {
        return "Saldo: " + saldo +
            "\nConsignaciones: " + consignaciones +
            "\nRetiros: " + retiros +
            "\nTasa Anual: " + tasaAnual +
            "\nComision: " + comision;
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E1_CuentasBancarias/CuentaBancaria.java

Código: CuentaAhorros.java

```
package Tareas.Tarea4.Partel.El_CuentasBancarias;

public class CuentaAhorros extends CuentaBancaria {

    protected boolean isActive;

    public CuentaAhorros(float saldo, float tasaAnual) {
        super(saldo, tasaAnual);
        checkActive();
    }

    public void checkActive() {
        isActive = saldo >= 10000;
    }

    @Override
    public void retirar(float cantidad) {
        if (!isActive) return;
        super.retirar(cantidad);
    }

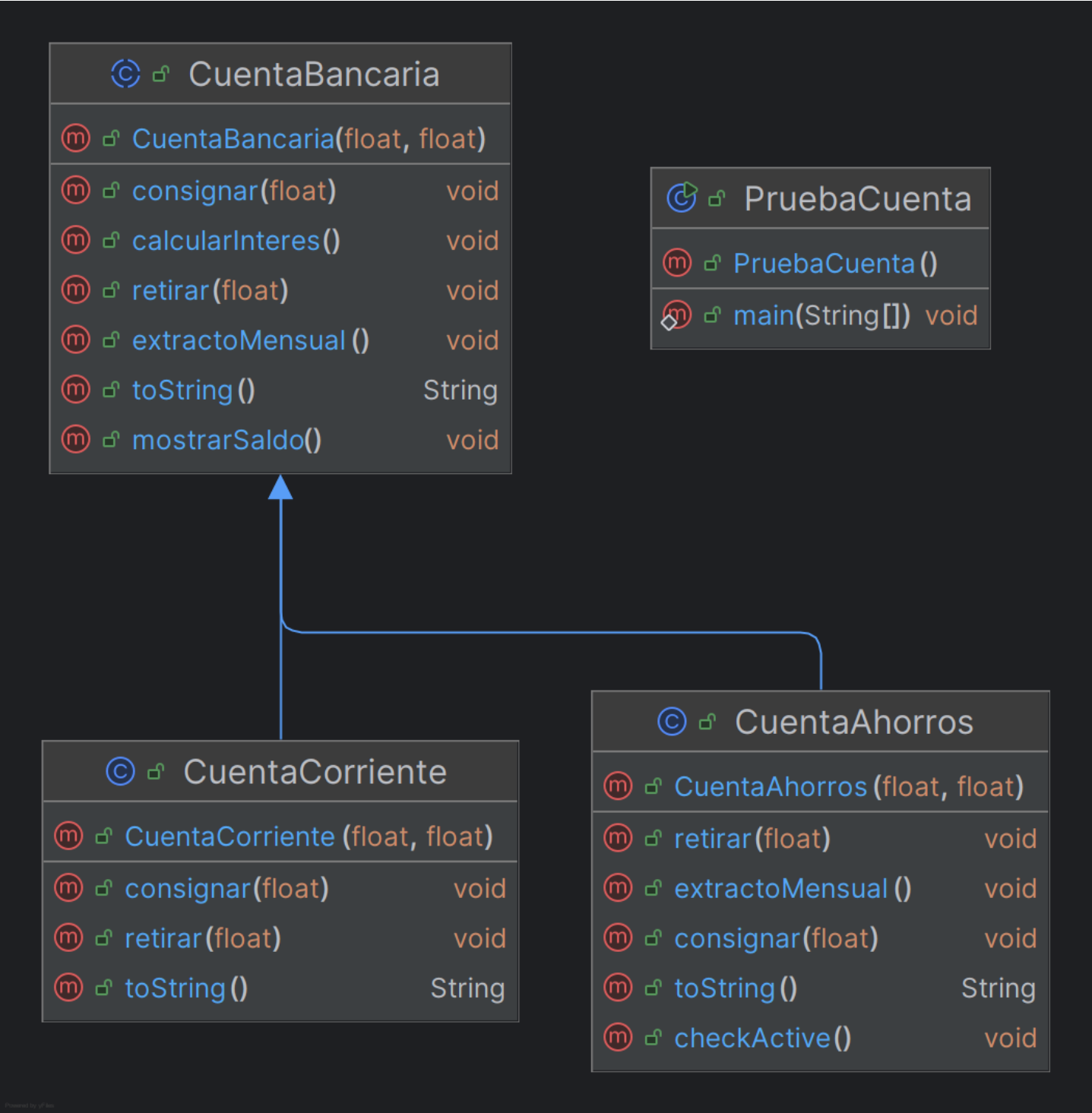
    @Override
    public void consignar(float cantidad) {
        if (!isActive) return;
        super.consignar(cantidad);
    }

    @Override
    public void extractoMensual() {
        if (retiros > 4) comision += (retiros - 4) * 1000;
        super.extractoMensual();
        checkActive();
    }

    @Override
    public String toString() {
        return super.toString() +
            "\nEstado: " + (isActive ? "Activa" : "Inactiva");
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E1_CuentasBancarias/CuentaAhorros.java

Imagen: DiagramaUML.png



Código: CuentaCorriente.java

```
package Tareas.Tarea4.Partel.El_CuentasBancarias;

public class CuentaCorriente extends CuentaBancaria {

    protected float sobregiro = 0;

    public CuentaCorriente(float saldo, float tasaAnual) {
        super(saldo, tasaAnual);
    }

    @Override
    public void retirar(float cantidad) {
        float nuevoSaldo = saldo - cantidad;
        if (nuevoSaldo < 0) {
            sobregiro = sobregiro - nuevoSaldo;
            saldo = 0;
            return;
        }
        super.retirar(cantidad);
    }

    @Override
    public void consignar(float cantidad) {
        if (sobregiro <= 0) {
            super.consignar(cantidad);
            return;
        }
        float nuevoSobregiro = sobregiro - cantidad;
        if (nuevoSobregiro > 0) {
            sobregiro = 0;
            saldo = nuevoSobregiro;
        } else {
            sobregiro = -nuevoSobregiro;
            saldo = 0;
        }
    }

    @Override
    public String toString() {
        return super.toString() +
            "\nSobregiro: " + sobregiro;
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E1_CuentasBancarias/CuentaCorriente.java

Código: PruebaCuenta.java

```
package Tareas.Tarea4.Partel.El_CuentasBancarias;

import java.util.Scanner;

public class PruebaCuenta {

    public static void main(String... args) {
        Scanner input = new Scanner(System.in);
        System.out.println("Cuenta de ahorros");
        System.out.println("Ingrese saldo inicial= $");
        float saldoInicialAhorros = input.nextFloat();
        System.out.print("Ingrese tasa de interés= ");
        float tasaAhorros = input.nextFloat();
        CuentaAhorros cuental = new CuentaAhorros(saldoInicialAhorros, tasaAhorros);
        System.out.print("Ingresar cantidad a consignar: $");
        float cantidadDepositar = input.nextFloat();
        cuental.consignar(cantidadDepositar);
        System.out.print("Ingresar cantidad a retirar: $");
        float cantidadRetirar = input.nextFloat();
        cuental.retirar(cantidadRetirar);
        cuental.extractoMensual();
        cuental.mostrarSaldo();
    }
}
```

Enlace: https://github.com/Simppplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E1_CuentasBancarias/PruebaCuenta.java

Código: Escalador.java

```
package Tareas.Tarea4.Partel.E4_Ciclista;

public class Escalador extends Ciclista {

    private float aceleracion;
    private float gradoRampa;

    public Escalador(int id, String nombre, float aceleracion, float gradoRampa) {
        super(id, nombre);
        this.aceleracion = aceleracion;
        this.gradoRampa = gradoRampa;
    }

    @Override
    public String imprimirTipo() {
        return "Es un escalador";
    }

    protected float getAceleracion() {
        return aceleracion;
    }

    protected void setAceleracion(float aceleracion) {
        this.aceleracion = aceleracion;
    }

    protected float getGradoRampa() {
        return gradoRampa;
    }

    protected void setGradoRampa(float gradoRampa) {
        this.gradoRampa = gradoRampa;
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E4_Ciclista/Escalador.java

Código: PruebaCiclista.java

```
package Tareas.Tarea4.Partel.E4_Ciclista;

public class PruebaCiclista {

    public static void main(String... args) {
        Equipo equipo1 = new Equipo("Sky","Estados Unidos");
        Velocista velocista1 = new Velocista(123979, "Geraint Thomas",
            320, 25);
        Escalador escalador1 = new Escalador(123980, "Egan Bernal",
            25, 10);
        Contrarrelojista contrarrelojista1 = new Contrarrelojista(123981,
            "Jonathan Castroviejo", 120);
        equipo1.addCiclista(velocista1);
        equipo1.addCiclista(escalador1);
        equipo1.addCiclista(contrarrelojista1);
        velocista1.setTiempo(365);
        escalador1.setTiempo(385);
        contrarrelojista1.setTiempo(370);
        equipo1.calcularTotalTiempo();
        equipo1.imprimir();
        equipo1.listarEquipo();
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E4_Ciclista/PruebaCiclista.java

Código: Velocista.java

```
package Tareas.Tarea4.Partel.E4_Ciclista;

public class Velocista extends Ciclista {
    private double potencia;
    private double velocidad;

    public Velocista(int id, String nombre, double potencia, double velocidad) {
        super(id, nombre);
        this.potencia = potencia;
        this.velocidad = velocidad;
    }

    @Override
    public String imprimirTipo() {
        return "Es un velocista";
    }

    protected double getPotencia() {
        return potencia;
    }

    protected void setPotencia(double potencia) {
        this.potencia = potencia;
    }

    protected double getVelocidad() {
        return velocidad;
    }

    protected void setVelocidad(double velocidad) {
        this.velocidad = velocidad;
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E4_Ciclista/Velocista.java

Código: Ciclista.java

```
package Tareas.Tarea4.Partel.E4_Ciclista;

public abstract class Ciclista {

    private int id;
    private String nombre;
    private double tiempo;

    public Ciclista(int id, String nombre) {
        this.id = id;
        this.nombre = nombre;
        this.tiempo = 0;
    }

    protected int getId() {
        return id;
    }

    protected void setId(int id) {
        this.id = id;
    }

    protected String getNombre() {
        return nombre;
    }

    protected void setNombre(String nombre) {
        this.nombre = nombre;
    }

    protected double getTiempo() {
        return tiempo;
    }

    protected void setTiempo(double tiempo) {
        this.tiempo = tiempo;
    }

    public abstract String imprimirTipo();
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E4_Ciclista/Ciclista.java

Imagen: DiagramaUML.png



Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E4_Ciclista/DiagramaUML.png

Código: Equipo.java

```
package Tareas.Tarea4.Partel.E4_Ciclista;

import java.util.HashMap;
import java.util.Map;

public class Equipo {

    private String nombre;
    private String pais;
    private double tiempoTotal;
    private final Map<Integer, Ciclista> ciclistas = new HashMap<>();

    public Equipo(String nombre, String pais) {
        this.nombre = nombre;
        this.pais = pais;
    }

    protected String getNombre() {
        return nombre;
    }

    protected void setNombre(String nombre) {
        this.nombre = nombre;
    }

    protected String getPais() {
        return pais;
    }

    protected void setPais(String pais) {
        this.pais = pais;
    }

    protected double getTiempoTotal() {
        return tiempoTotal;
    }

    protected void setTiempoTotal(double tiempoTotal) {
        this.tiempoTotal = tiempoTotal;
    }

    public void addCiclista(Ciclista ciclista) {
        this.ciclistas.put(ciclista.getId(), ciclista);
    }

    public void calcularTotalTiempo() {
        this.tiempoTotal =
            this.ciclistas.values().stream().mapToDouble(Ciclista::getTiempo).sum();
    }

    public void imprimir() {
        System.out.println("Nombre del equipo: " + this.nombre);
        System.out.println("Pais del equipo: " + this.pais);
        System.out.println("Tiempo total del equipo: " + this.tiempoTotal);
    }

    protected String getCiclistaString(Ciclista ciclista) {
        return "Nombre del ciclista: " + ciclista.getNombre() + "\n" +
            "ID del ciclista: " + ciclista.getId() + "\n" +
            "Tiempo del ciclista: " + ciclista.getTiempo() + "\n" +
            ciclista.imprimirTipo();
    }
}
```

```
public void listarEquipo() {  
    this.ciclistas.values().forEach(ciclista -> System.out.println(getCiclistaString(ciclista)));  
}  
  
public void imprimirCiclista(int id) {  
    Ciclista ciclista = this.ciclistas.getDefault(id, null);  
    if (ciclista != null) System.out.println(getCiclistaString(ciclista));  
    else System.out.println("No se encontró el ciclista");  
}  
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E4_Ciclista/Equipo.java

Código: Contrarrelojista.java

```
package Tareas.Tarea4.Partel.E4_Ciclista;

public class Contrarrelojista extends Ciclista{

    private double velocidadMaxima;

    public Contrarrelojista(int id, String nombre, double velocidadMaxima) {
        super(id, nombre);
        this.velocidadMaxima = velocidadMaxima;
    }

    @Override
    public String imprimirTipo() {
        return "Es un contrarrelojista";
    }

    protected double getVelocidadMaxima() {
        return velocidadMaxima;
    }

    protected void setVelocidadMaxima(double velocidadMaxima) {
        this.velocidadMaxima = velocidadMaxima;
    }
}
```

Enlace: https://github.com/Simpplay/POO-2024-2/tree/master//Tareas/Tarea4/Parte1/E4_Ciclista/Contrarrelojista.java