

TRABAJO PRÁCTICO N°1

Laboratorio I

Prof. Mercado Luis

Prof. Saez Juan José

Grupo 5

Ciruelos Matías

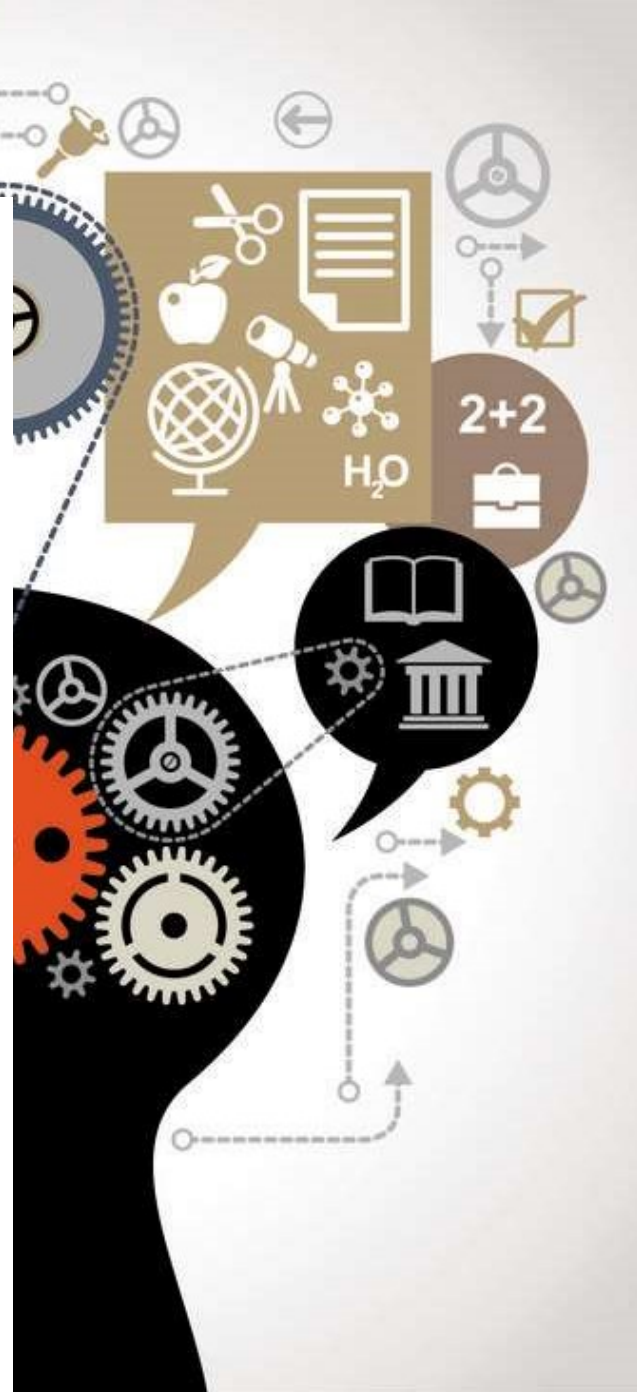
Ferrieres Mauricio

Ortega Simón

Videla Juan

PRIMER AÑO – C2

TEC.UNIVERSITARIA EN DESARROLLO DE SOFTWARE

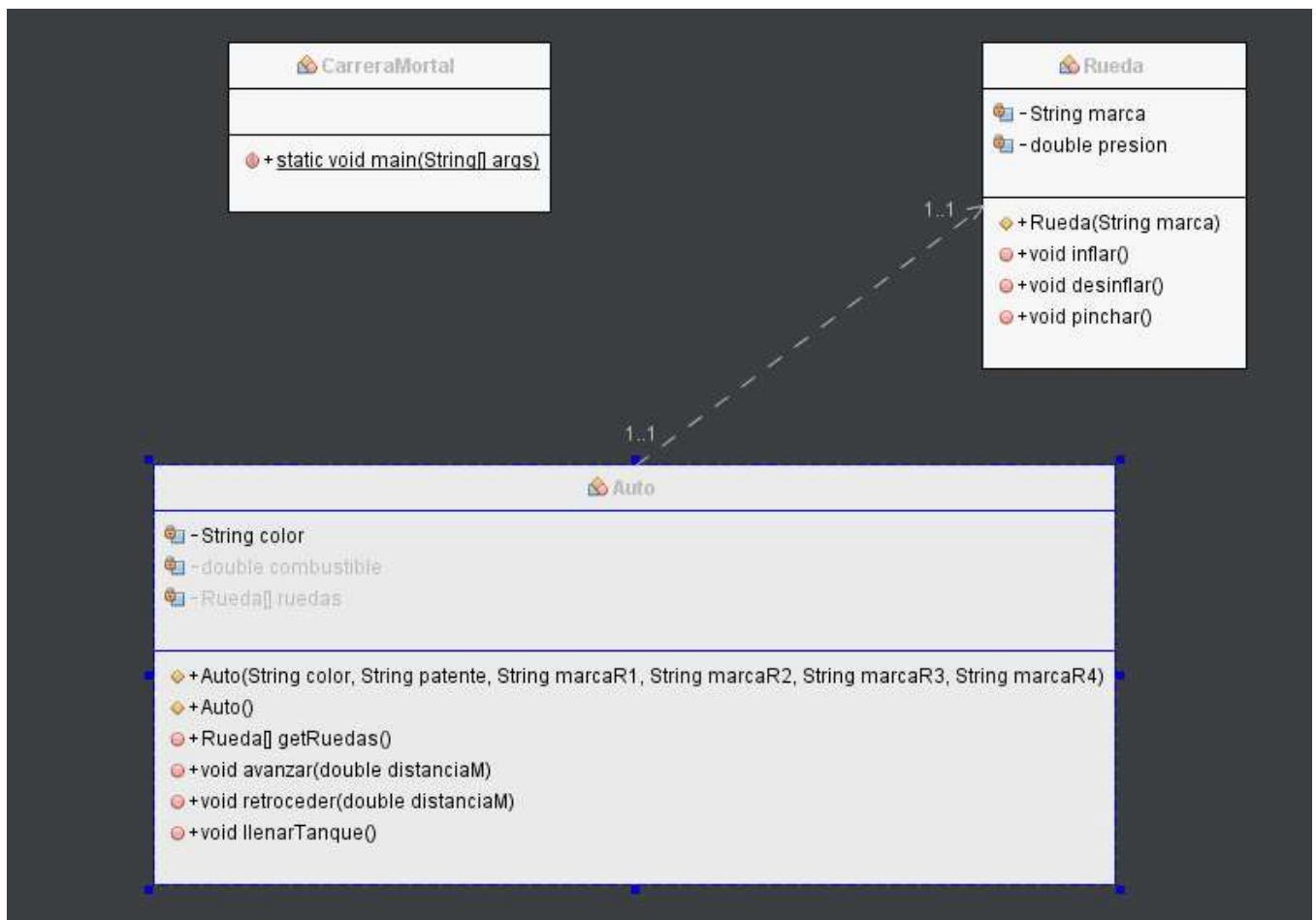


Trabajo Práctico

Ejercicio 1

1. Carrera Mortal (Auto-Ruedas)

UML



Clase Auto:

```
public class Auto {

    private String color, patente;
    private double combustible = 50;
    private Rueda[] ruedas = new Rueda[4];

    public Auto(String color, String patente, String marcaR1, String marcaR2, String marcaR3, String
marcaR4) {
        this.color = color;
        this.patente = patente;
        this.ruedas[0] = new Rueda(marcaR1);
        this.ruedas[1] = new Rueda(marcaR2);
        this.ruedas[2] = new Rueda(marcaR3);
        this.ruedas[3] = new Rueda(marcaR4);
    }

    public Auto() {
    }

    public Rueda[] getRuedas() {
        return ruedas;
    }

    public void avanzar(double distanciaM) {
        if (combustible >= (distanciaM / 10)) {
            combustible -= distanciaM / 10;
            System.out.println("Se avanzaron " + distanciaM + " metros");
            System.out.println("quedan " + this.combustible + "litros");
        } else {
            System.out.println("No hay combustible");
        }
    }

    public void retroceder(double distanciaM) {
        if (combustible >= (distanciaM / 10)) {
            combustible -= distanciaM / 10;
            System.out.println("Se retrocedieron " + distanciaM + " metros");
            System.out.println("quedan " + this.combustible + "litros");
        } else {
            System.out.println("No hay combustible");
        }
    }

    public void llenarTanque() {
        this.combustible = 50;
        System.out.println("Tanque lleno");
    }
}
```

```
}  
}
```

Clase Rueda:

```
public class Rueda {  
    private String marca;  
    private double presion = 28;  
  
    public Rueda(String marca) {  
        this.marca = marca;  
    }  
  
    public void inflar(){  
        this.presion=28;  
        System.out.println("Rueda inflada, la presion actual es " + this.presion);  
    }  
  
    public void desinflar(){  
        this.presion-= 0.5;  
        System.out.println("Rueda desinflada, La presion actual es " + this.presion);  
    }  
    public void pinchar(){  
        this.presion = 0;  
        System.out.println("Rueda pichada, la presion actual es"+ this.presion);  
    }  
}
```

Implementación:

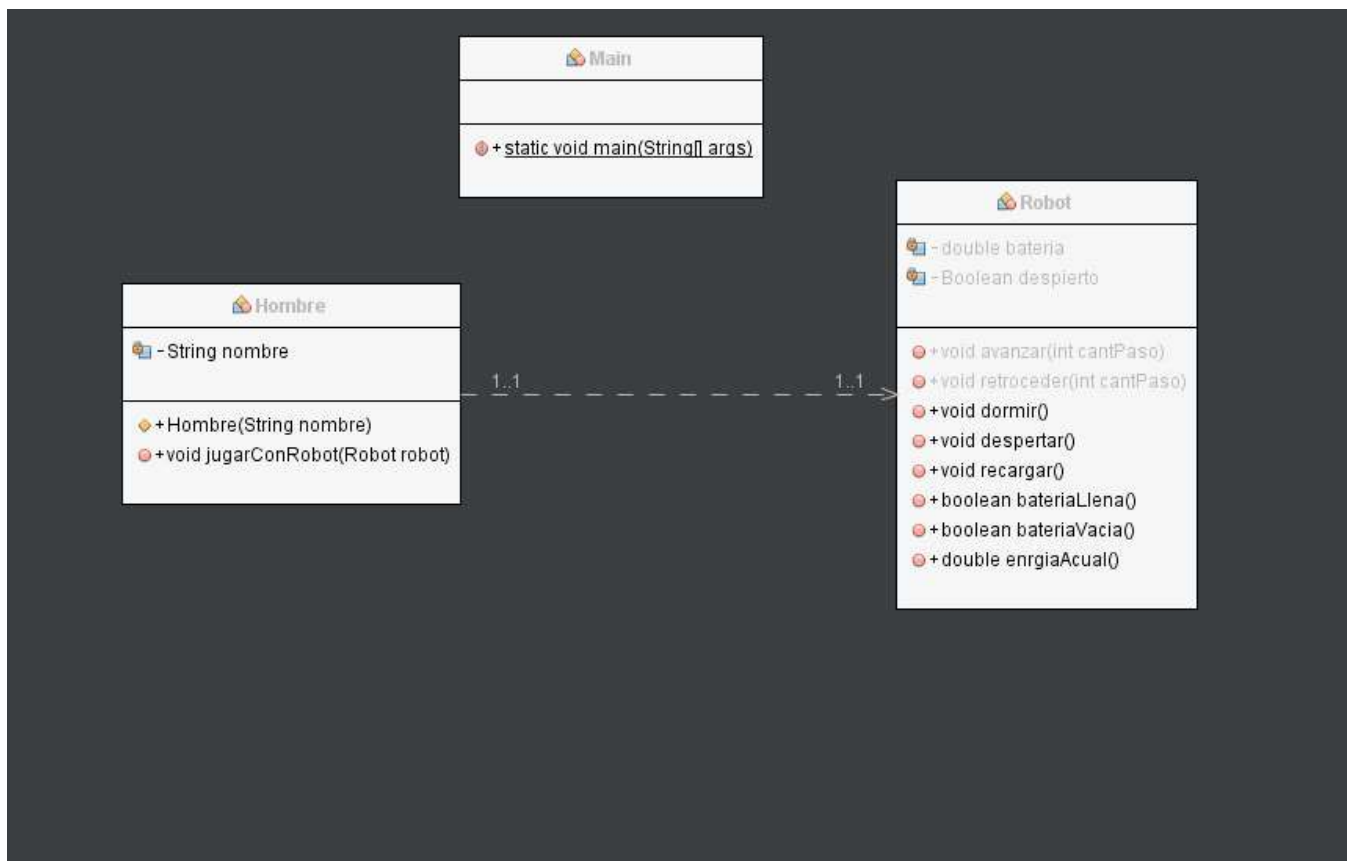
```
public class CarreraMortal {  
  
    /**  
     * @param args the command line arguments  
     */  
}
```

```
public static void main(String[] args) {  
    // TODO code application logic here  
    Auto auto = new Auto("rojo", "fiat", "Bridgestone","Bridgestone","Bridgestone","Bridgestone" );  
  
    auto.avanzar(15);  
    auto.retroceder(10);  
    auto.avanzar(20);  
    auto.avanzar(30);  
    auto.avanzar(20);  
    auto.llenarTanque();  
    auto.avanzar(30);  
    auto.getRuedas()[0].pinchar();  
    auto.getRuedas()[1].desinflar();  
    auto.getRuedas()[0].inflar();  
}  
  
}  
}
```

Ejercicio 2

Clase Hombre y Robot :

UML



Clase Robot

```
package tp_Robot;

import javax.swing.JOptionPane;

public class Robot {

    private double bateria = 1000;
    private Boolean despierto = true;

    public void avanzar(int cantPaso){
        if ( despierto){
            if (bateria >= cantPaso /10){
                bateria -= cantPaso/10;
                // System.out.println("Avanzo " + cantPaso + "Pasos");
                JOptionPane.showMessageDialog(null, "Avanzó " + cantPaso + "Pasos");
                //System.out.println("Energia Restante "+ bateria);
                JOptionPane.showMessageDialog(null, "Energia Restante "+ bateria);
            } else {
                // System.out.println( "Energia insuficiente");
                JOptionPane.showMessageDialog(null, "Energia insuficiente");
            }
        } else {
            //System.out.println("No se puede Avanzar, esta dormido");
            JOptionPane.showMessageDialog(null, "No se puede Avanzar, esta dormido");
        }
    }

    public void retroceder (int cantPaso){
        JOptionPane.showMessageDialog(null, "Retrocediendo");//al usar Avanzar
        (retroceder no puede mostrar sus carteles)
        avanzar(cantPaso);
    }
}
```

```
public void dormir(){
    if(despierto){
        despierto = false;
    }else {
        //System.out.println("Esta Dormido");
        JOptionPane.showMessageDialog(null, "Esta Dormido");
    }
}

public void despertar(){
    if( !despierto){
        despierto = true;
    } else{
        //System.out.println("Esta Despierto");
        JOptionPane.showMessageDialog(null, "Esta Despierto");
    }
}

public void recargar(){
    bateria=1000;//faltaba el llenado de la bateria
    // System.out.println("Bateria cargada");
    JOptionPane.showMessageDialog(null, "Bateria cargada");
}

public boolean bateriaLlena(){
    return bateria == 1000;
}

public boolean bateriaVacía(){
    return bateria == 0;
}

public double enrgiaAcual(){
    return bateria;
}

}
```

Clase Hombre:

```
package tp_Robot;

import java.util.Scanner;
import javax.swing.JOptionPane;

public class Hombre {
    private String nombre;

    public Hombre(String nombre) {
        this.nombre = nombre;
    }

    // public void jugarConRobot(Robot robot){
    //     robot.avanzar(500);
    //     robot.retroceder(20);
    //     robot.enrgiaAcual();
    //     robot.dormir();
    // }

    public void jugarConRobot(Robot robot){
        //Scanner sc = new Scanner(System.in);
        int entrada;

        do{
            try {
                entrada = Integer.parseInt(JOptionPane.showInputDialog( "Seleccione una
opción:\n1. Avanzar\n2.Retroceder\n"
                + "3. Dormir\n4. Despertar\n5. Recargar\n6. Bateria LLena\n7. Bateria
Vacía\n8. Energia Actual\n9. Salir del programa" ));

            } catch (NumberFormatException e) {
                entrada = 9;
            }
        }
```

```
switch (entrada){
    case 1:

        robot.avanzar(Integer.parseInt(JOptionPane.showInputDialog("Ingrese la
cantidad de paso")));
        break;
    case 2:
        robot.retroceder(Integer.parseInt(JOptionPane.showInputDialog("Ingrese la
cantidad de paso")));
        break;
    case 3:
        robot.dormir();
        break;
    case 4:
        robot.despertar();
        break;
    case 5:
        robot.recargar();
        break;
    case 6:

        if (robot.bateriaLlena()){
            JOptionPane.showMessageDialog(null,"Bateria Llena");
        }
        else{JOptionPane.showMessageDialog(null,"La Bateria no esta Llena");}
        break;
    case 7:
        if(robot.bateriaVacía()){

            JOptionPane.showMessageDialog(null,"Bateria Vacía");
        }
        else{ JOptionPane.showMessageDialog(null,"La bateria aun tiene carga");}
        break;
    case 8:
        int e = (int) robot.enrgiaAcual();
```

```

        JOptionPane.showMessageDialog(null,"Energia Actual "+e);
        break;
    case 9:
        //System.out.println("Ejecución finalizada");
        JOptionPane.showMessageDialog(null, "Ejecución finalizada");
        System.exit(0);
        break;
    }
}while(entrada <= 9 || entrada > 1 );
}

}

```

Implementación:

```

package tp_Robot;

public class Main {
    public static void main(String[] args) {

        Hombre h1 =new Hombre("Pepe");
        Robot r1 = new Robot();
        h1.jugarConRobot(r1);
        // Hombre h2 = new Hombre("raul");

        //h2.jugarConRobot(r1);

    }

}

```