

Clase Abstracta:

```
package claseAbstracta;

/**
 *
 * @author umg
 */
public abstract class vehiculos{

    protected double placa;
    protected boolean alquilado;

    public vehiculos(double placa, boolean alquilado) {
        this.placa = placa;
        this.alquilado = alquilado;
    }

    public double getPlaca() {
        return placa;
    }

    public void setPlaca(double placa) {
        this.placa = placa;
    }

    public boolean isAlquilado() {
        return alquilado;
    }
}
```

```
public void setAlquilado(boolean alquilado) {  
    this.alquilado = alquilado;  
}
```

```
public abstract double calcularImporte();
```

```
public abstract void mostrarDatos();  
}
```

Clase Autobús:

```
package clases;
```

```
import java.util.Scanner;
```

```
import claseAbstracta.vehiculos;
```

```
/**
```

```
*
```

```
* @author umg
```

```
*/
```

```
public class Autobus extends vehiculos{
```

```
    private double precioPorKilometro;
```

```
    private double kilometrosRenta;
```

```
    private double kilometrosDevolucion;
```

```
public Autobus(double placa, boolean alquilado) {
```

```
    super(placa, alquilado);
```

```
    this.precioPorKilometro = 0.0; // Inicializamos el precio por kilómetro a cero
```

```
}
```

```
// Setter para establecer el precio por kilómetro  
public void setPrecioPorKilometro(double precioPorKilometro) {  
    this.precioPorKilometro = precioPorKilometro;  
}
```

```
// Método para capturar los kilómetros de renta  
public void capturarKilometrosRenta() {  
    Scanner scanner = new Scanner(System.in);  
    System.out.print("Ingrese los kilómetros de renta: ");  
    this.kilometrosRenta = scanner.nextDouble();  
}
```

```
// Método para capturar los kilómetros de devolución  
public void capturarKilometrosDevolucion() {  
    Scanner scanner = new Scanner(System.in);  
    System.out.print("Ingrese los kilómetros de devolución: ");  
    this.kilometrosDevolucion = scanner.nextDouble();  
}
```

@Override

```
public double calcularImporte() {  
    return (kilometrosDevolucion - kilometrosRenta) * precioPorKilometro;  
}
```

@Override

```
public void mostrarDatos() {
```

```

        System.out.println("Placa: " + placa);
        System.out.println("Alquilado: " + alquilado);
        if (alquilado) {
            System.out.println("Kilómetros de renta: " + kilometrosRenta);
            System.out.println("Kilómetros de devolución: " + kilometrosDevolucion);
            System.out.println("Importe a pagar: $" + calcularImporte());
        }
    }
}

```

Clase Tractor:

```

import java.util.Scanner;
import claseAbstracta.vehiculos;

/**
 *
 * @author umg
 */
public class Tractor extends vehiculos {

    private double precioPorDia;
    private int diasRenta;

    public Tractor(double placa, boolean alquilado) {
        super(placa, alquilado);
        this.precioPorDia = 0.0; // Inicializamos el precio por día a cero
    }
}

```

```

// Setter para establecer el precio por día
public void setPrecioPorDia(double precioPorDia) {
    this.precioPorDia = precioPorDia;
}

// Método para capturar la cantidad de días de renta
public void capturarDiasRenta() {
    Scanner scanner = new Scanner(System.in);
    System.out.print("Ingrese la cantidad de días de renta: ");
    this.diasRenta = scanner.nextInt();
}

@Override
public double calcularImporte() {
    return diasRenta * precioPorDia;
}

@Override
public void mostrarDatos() {
    System.out.println("Placa: " + placa);
    System.out.println("Alquilado: " + alquilado);
    if (alquilado) {
        System.out.println("Días de renta: " + diasRenta);
        System.out.println("Importe a pagar: $" + calcularImporte());
    }
}
}

```

Clase principal:

```
package polimorfismo;

import claseAbstracta.vehiculos;
import clases.Autobus;
import clases.Tractor;

/**
 *
 * @author umg
 */
public class Polimorfismo {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {

        // Crear una instancia de Autobus
        Autobus autobus = new Autobus(123, false);

        // Establecer el precio por kilómetro
        autobus.setPrecioPorKilometro(0.5);

        // Capturar los kilómetros de renta y devolución
        autobus.capturarKilometrosRenta();
        autobus.capturarKilometrosDevolucion();
    }
}
```

```
// Calcular e imprimir el importe
double importeAutobus = autobus.calcularImporte();
System.out.println("Importe a pagar por el Autobús: $" + importeAutobus);

// Crear una instancia de Tractor
Tractor tractor = new Tractor(456, false);

// Establecer el precio por día
tractor.setPrecioPorDia(25.0);

// Capturar la cantidad de días de renta
tractor.capturarDiasRenta();

// Calcular e imprimir el importe
double importeTractor = tractor.calcularImporte();
System.out.println("Importe a pagar por el Tractor: $" + importeTractor);
}
}
```

Vista:

Tipo de Vehículo

☐ Autobús

Rentar

Devolver

Design Preview [vista]

Tipo de Vehículo

☐ Autobús

☒ Tractor

Rentar

Devolver

Salir

Datos del Vehículo

Placa

Precio Renta

Kms al rentar

Kms al devolver

Fecha de Renta

Fecha de Devolucion