



Reto 5 - Concesionario

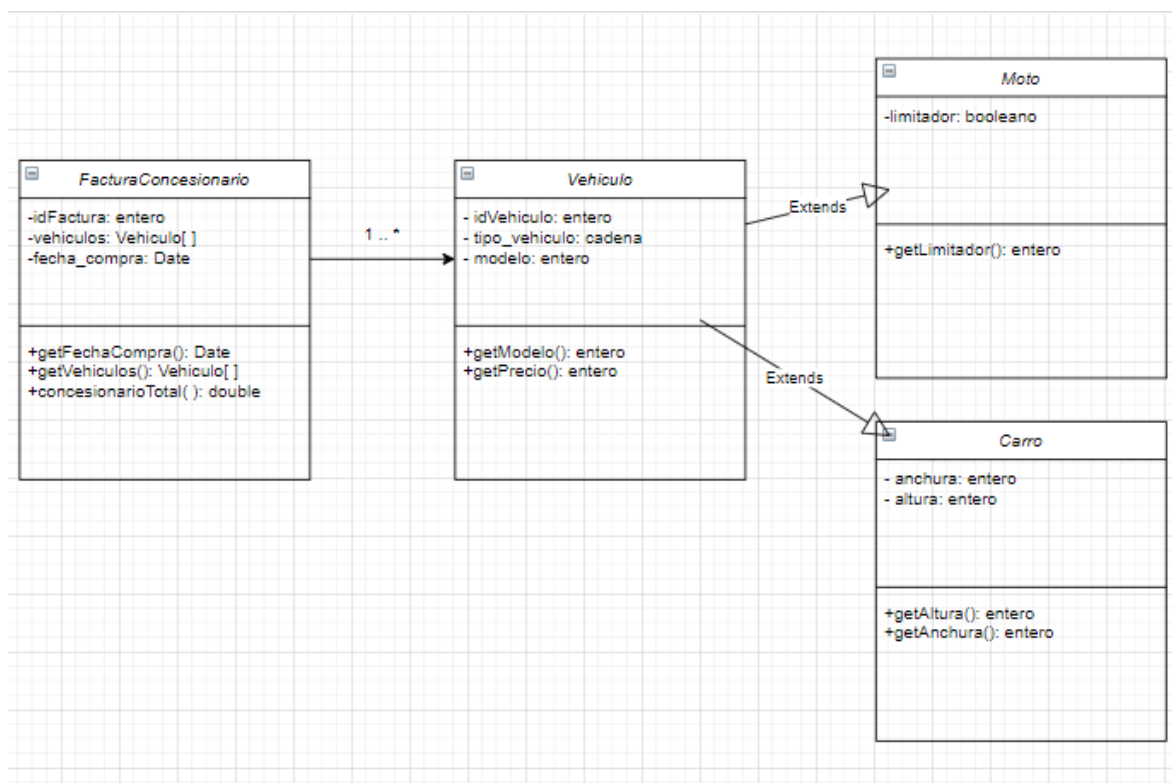
Objetivo

El objetivo de este reto es que el estudiante reconozca y aplique los elementos básicos del paradigma de la programación orientada a objetos en un escenario abstraído de la cotidianidad.

Contexto

El concesionario de carros ha venido en crecimiento últimamente y cada día su sistema necesita más eficiencia en todos los frentes, especialmente necesita apoyo en el sistema de devueltas. Por tanto, se le pide a usted como desarrollador, diseñar un programa que permita mejorar este aspecto del sistema y permitirle al cajero saber la forma en que debe devolverle al cliente después de una compra.

Considere el siguiente diagrama de clases para la implementación de las clases necesarias **FacturaConcesionario.java**, **Vehiculo.java**, **Carro.java** y **Moto.java**



NOTA: Las clases deben llamarse **FacturaConcesionario.java**, **Vehiculo.java**, **Carro.java** y **Moto.java**



Reto

Implemente una función nueva llamada `concesionarioDevuelta()`, la cual recibirá como parámetro el total a pagar por los vehículos adquiridos por parte del cliente (reutilizar la función `calcularTotal()` de reto 3) y además, también recibirá como parámetro el valor cancelado por el usuario, el cual es un atributo de la clase `FacturaConcesionario`.

Al final, la salida esperada para este reto es la cantidad de billetes y monedas que debe emplear el cajero para devolverle el dinero al usuario. Se deben seguir los siguientes parámetros para realizar este cálculo:

- a) En caso de que el vehículo sea una moto, se le debe devolver al usuario con billetes 50.000, 10.000 y monedas de 500.
- b) Por otra parte, si el vehículo es un carro, se le debe devolver al usuario con billetes de 100.000, 20.000 y monedas de 200.

SUGERENCIA: Usar función `Math.floor` para operaciones que impliquen divisiones.

Adicionalmente, use la siguiente imagen como referencia para la construcción de las clases con sus atributos y métodos necesarios.



```
import java.sql.Date;

public class FacturaConcesionario {
    private int id_factura;
    private Vehiculo[] vehiculos;
    private Date fecha_compra;
    private int valor_cancelado;

    public FacturaConcesionario(int id_factura, Vehiculo[] vehiculos, Date fecha_compra, int valor_cancelado) {
        this.id_factura = id_factura;
        this.vehiculos = vehiculos;
        this.fecha_compra = fecha_compra;
    }

    public int concesionarioTotal() {
        int suma = 0;

        return suma;
    }

    public String concesionarioDevuelta(double valorTotal) {
        String mensaje = "";

        return mensaje;
    }

    public FacturaConcesionario() {
        // TODO Auto-generated constructor stub
    }

    public static void main(String[] args) {

    }

    int getId_factura() {
        return id_factura;
    }

    void setId_factura(int id_factura) {
        this.id_factura = id_factura;
    }

    Vehiculo[] getVehiculos() {
        return vehiculos;
    }

    void setVehiculos(Vehiculo[] vehiculos) {
        this.vehiculos = vehiculos;
    }

    Date getFecha_compra() {
        return fecha_compra;
    }

    void setFecha_compra(Date fecha_compra) {
        this.fecha_compra = fecha_compra;
    }
}

public class Vehiculo {

    private int id_vehiculo;
    private String tipo_vehiculo;
    private int modelo;

    public Vehiculo(int id_vehiculo, String tipo_vehiculo, int modelo) {

    }

    public Vehiculo() {
        // TODO Auto-generated constructor stub
    }

    int getId_vehiculo() {
        return id_vehiculo;
    }

    void setId_vehiculo(int id_vehiculo) {
        this.id_vehiculo = id_vehiculo;
    }

    String getTipo_vehiculo() {
        return tipo_vehiculo;
    }

    void setTipo_vehiculo(String tipo_vehiculo) {
        this.tipo_vehiculo = tipo_vehiculo;
    }

    int getModelo() {
        return modelo;
    }

    void setModelo(int modelo) {
        this.modelo = modelo;
    }
}
```



```
public class Carro extends Vehiculo {
    private int altura;
    private int anchura;

    public Carro(int altura, int anchura) {
        this.altura = altura;
        this.anchura = anchura;
    }

    public Carro(int id_vehiculo,String tipo_vehiculo,int modelo,int altura, int anchura) {
    }

    public Carro() {
        // TODO Auto-generated constructor stub
    }

    int getAltura() {
        return altura;
    }

    void setAltura(int altura) {
        this.altura = altura;
    }

    int getAnchura() {
        return anchura;
    }

    void setAnchura(int anchura) {
        this.anchura = anchura;
    }
}

public class Moto extends Vehiculo {

    private boolean delimitador;

    public Moto(boolean delimitador) {
        this.delimitador = delimitador;
    }

    public Moto(int id_vehiculo,String tipo_vehiculo,int modelo,boolean delimitador) {
        this.setId_vehiculo(id_vehiculo);
        this.setTipo_vehiculo(tipo_vehiculo);
        this.setModelo(modelo);
        this.delimitador = delimitador;
    }

    public Moto() {
        // TODO Auto-generated constructor stub
    }

    boolean getDelimitador() {
        return delimitador;
    }

    void setDelimitador(boolean delimitador) {
        this.delimitador = delimitador;
    }
}
```



Casos de prueba

Finalmente, para verificar el funcionamiento del programa se sugiere considerar los siguientes casos de prueba:

# CASO DE PRUEBA	DATO DE ENTRADA	SALIDA ESPERADA				
1	<table><tr><th>FacturaConcesionario</th><th>Moto</th></tr><tr><td>Id_factura: 0 Fecha_compra: 7/22/2021" valor_cancelado: 10'020.000</td><td>Id_vehiculo: 1 Tipo_vehiculo: "Moto" modelo: 2007 limitador: false</td></tr></table> <p>En este caso, el valor total a pagar por la función implementada en reto 3 debe ser 8500000</p>	FacturaConcesionario	Moto	Id_factura: 0 Fecha_compra: 7/22/2021" valor_cancelado: 10'020.000	Id_vehiculo: 1 Tipo_vehiculo: "Moto" modelo: 2007 limitador: false	El asistente del concesionario debe devolverle al comprador con 30 billetes de 50.000, 2 billetes de 10.000 y 0 monedas de 500
FacturaConcesionario	Moto					
Id_factura: 0 Fecha_compra: 7/22/2021" valor_cancelado: 10'020.000	Id_vehiculo: 1 Tipo_vehiculo: "Moto" modelo: 2007 limitador: false					
3	<table><tr><th>FacturaConcesionario</th><th>Carro</th></tr><tr><td>Id_factura: 1 Fecha_compra: 12/12/2021" valor_cancelado: 40'022.000</td><td>Id_vehiculo: 1 Tipo_vehiculo: "Carro" modelo: 2011 limitador: false</td></tr></table> <p>En este caso, el valor total a pagar por la función implementada en reto 3 debe ser 25'500.000</p>	FacturaConcesionario	Carro	Id_factura: 1 Fecha_compra: 12/12/2021" valor_cancelado: 40'022.000	Id_vehiculo: 1 Tipo_vehiculo: "Carro" modelo: 2011 limitador: false	El asistente del concesionario debe devolverle al comprador con 145 billetes de 100.000, 1 billete de 20.000 y 10 monedas de 200
FacturaConcesionario	Carro					
Id_factura: 1 Fecha_compra: 12/12/2021" valor_cancelado: 40'022.000	Id_vehiculo: 1 Tipo_vehiculo: "Carro" modelo: 2011 limitador: false					



Entrega:

1. Suba a la plataforma un archivo con el nombre de **FacturaConcesionario.java, Vehiculo.java, Moto.java y Carro.java** este nombre debe de respetarse, dado que, si no se nombre de dicha manera no se tendrá en cuenta para la calificación del reto.
2. **Importante:** Los métodos deben de llamarse **exactamente igual** a como se muestra en el ejemplo de la estructura del código.
3. **Importante:** Las salidas deben ser tal cual se muestran en los casos de pruebas. De lo contrario, el sistema no lo reconocerá.