

NOMBRE : Jose Manuel Vicente Checo.

MATRICULA : 2023-1073.

Crear una clase AutoMovil (Sin acento)

- Debe agregar 4 [atributos](#).
- Debe agregar 4 métodos.
- Debe aplicar [encapsulación](#) ([Atributos](#) privados y agregar setters y getters)
- Debe imprimir el nombre de la clase en el constructor. ("Constructor de AutoMóvil")

Crear la clase HondaCivic

- Debe heredar de la clase AutoMovil (extends)
- Debe imprimir el nombre de la clase en el constructor. ("Constructor de HondaCivic")
- Sobre escribir dos métodos.

CLASE AUTOMOVIL

```
public class AutoMovil {  
  
    //Atributos privados de la clase.  
  
    private String chasis;  
  
    private String marca;  
  
    private String modelo;  
  
    private String color;
```

//Constructor con parametros de la clase.

```
public AutoMovil(String chasis, String marca,  
    String modelo, String color, boolean presentar_mensaje) {
```

```
    this.chasis = chasis;
```

```
    this.marca = marca;
```

```
    this.modelo = modelo;
```

```
    this.color = color;
```

```
    /*Esto es un control para que este mensaje solo aparezca  
    en la clase AutoMovil y no en la heredada.*/
```

```
    if (presentar_mensaje) {
```

```
        System.out.println("Constructor de AutoMovil");
```

```
    }
```

```
}
```

//Metodos Getters y Setters.

```
public String getChasis() {
```

```
    return chasis;
```

```
}
```

```
public void setChasis(String chasis) {  
    this.chasis = chasis;  
}
```

```
public String getMarca() {  
    return marca;  
}
```

```
public void setMarca(String marca) {  
    this.marca = marca;  
}
```

```
public String getModelo() {  
    return modelo;  
}
```

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

```
public String getColor() {  
    return color;  
}
```

```
}
```

```
public void setColor(String color) {
```

```
    this.color = color;
```

```
}
```

```
//Metodos.
```

```
public void prenderAuto(int num) {
```

```
    if (num == 1) {
```

```
        System.out.println("El carro esta prendido");
```

```
    } else {
```

```
        System.out.println("Insertar la llave del AutoMovil");
```

```
    }
```

```
}
```

```
public void apagarAuto(int num2) {
```

```
    if (num2 == 0) {
```

```
        System.out.println("El carro esta apagado");
```

```
    } else {
```

```
        System.out.println("Favor de girar la llave en contra de las manecillas del reloj");
```

```
    }
```

```
}
```

```
public void lavarAuto() {
```

```
    System.out.println("Hoy toca lavar la maquina XD ");
```

```
}
```

```
public void conducirAuto() {
```

```
    System.out.println("Pasa un cambio y dale pa' lante...");
```

```
}
```

```
}
```

CLASE HONDACIVIC

```
//Clase HondaCivic hereda de la clase AutoMovil.
```

```
public class HondaCivic extends AutoMovil {
```

```
    //Declarando atributos propios de la clase Honda Civic.
```

```
private int year;
```

```
private int puertas;
```

```
//Constructor con parametros y atributos heredados de la clase AutoMovil.
```

```
public HondaCivic(String chasis, String marca, String modelo, String color, int year, int  
puertas) {
```

```
    super(chasis, marca, modelo, color, false);
```

```
    this.year = year;
```

```
    this.puertas = puertas;
```

```
    System.out.println("Constructor de Honda Civic");
```

```
}
```

```
//Metodos Getters y Setters de atributos propios de la clase Honda Civic.
```

```
public int getYear() {
```

```
    return year;
```

```
}
```

```
public void setYear(int year) {
```

```
    this.year = year;
```

```
}
```

```
public int getPuertas() {  
    return puertas;  
}
```

```
public void setPuertas(int puertas) {  
    this.puertas = puertas;  
}
```

```
//Metodos sobreescritos.
```

```
@Override
```

```
public void conducirAuto() {
```

```
    System.out.println("Este carro Honda es automatico, pon la [D] y dale pa' ya.");
```

```
}
```

```
@Override
```

```
public void lavarAuto() {
```

```
    System.out.println("Este carro necesita ser encerado luego de ser lavado.");
```

```
}
```

```
}
```

CLASE MAIN

```
public class Main {
```

```
    public static void main(String[] args) {
```

```
        //Objeto de la clase AutoMovil.
```

```
        //Aqui se llenan los parametros del constructor.
```

```
        AutoMovil auto1 = new AutoMovil("JUDM152487AZX", "NISSAN", "FRONTIER", "NEGRA",  
true);
```

```
        //Invocando los metodos.
```

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

```
        auto1.apagarAuto(0);
```

```
        auto1.conducirAuto();
```

```
        auto1.prenderAuto(1);
```

```
        auto1.lavarAuto();
```

```
        //Probando el metodo Get.
```



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

```
System.out.println("Chasis: " + auto1.getChasis());
```

```
System.out.println("Marca : " + auto1.getMarca());
```

```
System.out.println("Modelo : " + auto1.getModelo());
```

```
System.out.println("Color :" + auto1.getColor());
```

```
//Probando el metodo Set.
```

```
auto1.setColor("VERDE");
```

```
auto1.setMarca("MERCEDES BENZ");
```

```
auto1.setModelo("GLE");
```

```
//Segunda impresion.
```

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

```
System.out.println("ACTUALIZACION DE DATOS.");
```

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

```
System.out.println("Chasis: " + auto1.getChasis());
```

```
System.out.println("Marca : " + auto1.getMarca());
```

```
System.out.println("Modelo : " + auto1.getModelo());
```

```
System.out.println("Color :" + auto1.getColor());
```

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

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

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

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

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

//Clase HondaCivic heredada de la clase AutoMovil.

//Aqui se llenan los parametros del constructor.

HondaCivic auto2 = new HondaCivic("WER4787984SDF125F", "HONDA", "CIVIC",
"BLANCA", 2022, 4);

//Invocado los metodos de la clase HondaCivic.

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

auto2.apagarAuto(0);

auto2.conducirAuto();

auto2.prenderAuto(1);

auto2.lavarAuto();

//Probando el metodo Get.

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

System.out.println("Chasis: " + auto2.getChasis());

System.out.println("Marca : " + auto2.getMarca());

System.out.println("Modelo : " + auto2.getModelo());

System.out.println("Color : " + auto2.getColor());

System.out.println("Year : " + auto2.getYear());

System.out.println("Puerta : " + auto2.getPuertas());
```

```
//Probando el metodo Set.
```

```
auto2.setColor("NARANJA");
```

```
auto2.setMarca("AUDI");
```

```
auto2.setModelo("R8");
```

```
//Segunda impresion.
```

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

```
System.out.println("ACTUALIZACION DE DATOS.");
```

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

```
System.out.println("Chasis: " + auto2.getChasis());
```

```
System.out.println("Marca : " + auto2.getMarca());
```

```
System.out.println("Modelo : " + auto2.getModelo());
```

```
System.out.println("Color : " + auto2.getColor());
```

```
System.out.println("Year : " + auto2.getYear());
```

```
System.out.println("Puerta : " + auto2.getPuertas());
```

```
}
```

```
}
```

run:

Constructor de AutoMovil

El carro esta apagado

Pasa un cambio y dale pa' lante...

El carro esta prendido

Hoy toca lavar la maquina XD

Chasis: JUDM152487AZX

Marca : NISSAN

Modelo : FRONTIER

Color :NEGRA

ACTUALIZACION DE DATOS.

Chasis: JUDM152487AZX

Marca : MERCEDES BENZ

Modelo : GLE

Color :VERDE

```
-----  
Constructor de Honda Civic  
-----  
El carro esta apagado  
Este carro Honda es automatico, pon la [D] y dale pa' ya.  
El carro esta prendido  
Este carro necesita ser encerado luego de ser lavado.  
-----  
Chasis: WER4787984SDF125F  
Marca : HONDA  
Modelo : CIVIC  
Color :BLANCA  
Year :2022  
Puerta :4  
  
ACTUALIZACION DE DATOS.  
-----  
Chasis: WER4787984SDF125F  
Marca : AUDI  
Modelo : R8  
Color :NARANJA  
Year :2022  
Puerta :4  
BUILD SUCCESSFUL (total time: 0 seconds)
```