

UNIVERSIDAD MARIANO GALVEZ DE GUATEMALA
INGENIERIA EN SISTEMAS DE INFORMACION Y CIENCIAS DE LA
COMPUTACION CAMPUS JUTIAPA



TECER SEMESTRE

SECCION: "A"

- **Robertson Alessandro Muñoz Paredes**

Carnet: 0905-24-16433

PROGRA 1

DOCUMENTACION DEL CODIGO

Para realizar esta tarea me guie de los ejercicios anteriores ya que eran similares y solo cambian las funciones, y como ya se tenían echas dos clases me fue un poco más fácil guiarme

- **Punto1:** para derivar la clase **AutoDeCombustión** de la clase **Vehículo**. Para derivar una clase se usa un carácter clave (:) y seguida el nombre de la clase, en este caso nos quedaría así class AutoDeCombustión: vehiculos

para esto hacemos una variable publica la cual contendrá propiedades comunes de los vehiculos

```
1 referencia
public AutoDeCombustión(int anio, string model, string elColor) : base(anio, model, elColor)
4 referencias
public int Velocidad
```

ya luego agregamos public void public int, el public void lo agregamos para que no nos devuelva ningún valor al que podamos acceder, y el int lo usaremos para que podamos ver el nivel de gasolina, y como con el int nos permite almacenar decimales, en la parte de void Frenos agg un if-else ya que hice una funcion de velocidad

```
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52

public void Frenar()
{
    if (Velocidad > 0)
    {
        Velocidad -= 10;
        Console.WriteLine($"El vehículo ha frenado. Velocidad actual: {Velocidad} km/h");
    }
    else
    {
        Console.WriteLine("El vehículo ya está detenido.");
    }
}

0 referencias
public void cargarGasolina()
{
    nivelGasolina++;
}

1 referencia
public int verNivelGasolina()
{
    return nivelGasolina;
}

0 referencias
public void MostrarInfo()
{
    Console.WriteLine($"Año: {anio}, Modelo: {Modelo}, Color: {Color}, Velocidad: {Velocidad} km/h");
}
```

- **Punto2:** encapsular propiedades en la clase de motocicletas encapsule 3 propiedades las cuales las puse privadas

```
private string encendido = "patada";  
private string combustible = "super ";  
private int velocidad;
```

Luego sobrescribimos el método frenar en la clase de motocicleta para darle una funcion como la motocicleta frena más rápido que otras

En la clase derivada Motocicleta, vamos a sobrescribir el método Frenar para que, por ejemplo, frene más rápido que otros vehículos. Luego hacemos una funcion de acelerar

```
3 referencias  
public int Velocidad  
{  
    get { return velocidad; }  
    private set { velocidad = value; }  
}  
1 referencia  
public string verencendido()  
{  
    return encendido;  
}  
0 referencias  
public void Acelerar(int cuanto)  
{  
    Velocidad += cuanto;  
    Console.WriteLine($"El vehículo ha acelerado, su velocidad actual: {Velocidad} km/h");  
}
```

Luego usamos la funcion de la clase derivada frenar usando un if-else para crear una cadena mas fácil de manipular y agregándole un void para que no nos devuelva resultados

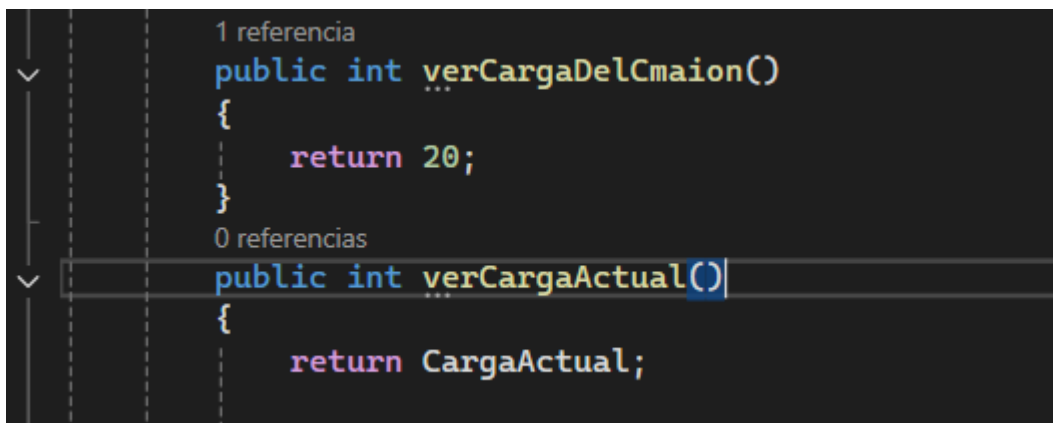
```
0 referencias  
public virtual void Frenar()  
{  
    if (Velocidad > 0)  
    {  
        Velocidad -= 10; // Frenado genérico  
        Console.WriteLine($"El vehículo ha frenado, su velocidad actual: {Velocidad} km/h");  
    }  
    else  
    {  
        Console.WriteLine("El vehículo ha sido detenido.");  
    }  
}
```

DOCUMENTACION GENERAL DE LOS PROBLEMAS Y SOLUCIONES QUE TUVE

Al crear las clases tuve varios problemas los cuales asta que vi todo el código logre solucionarlo, ya que cuando cree la clase, ejemplo camión la cree con la C mayúscula y cuando la escribía en el código me la daba como error también cuando quería sobre escribir las clases no me aceptaba los métodos adicionales que pedía en la tarea, el cual era el problema por una letra

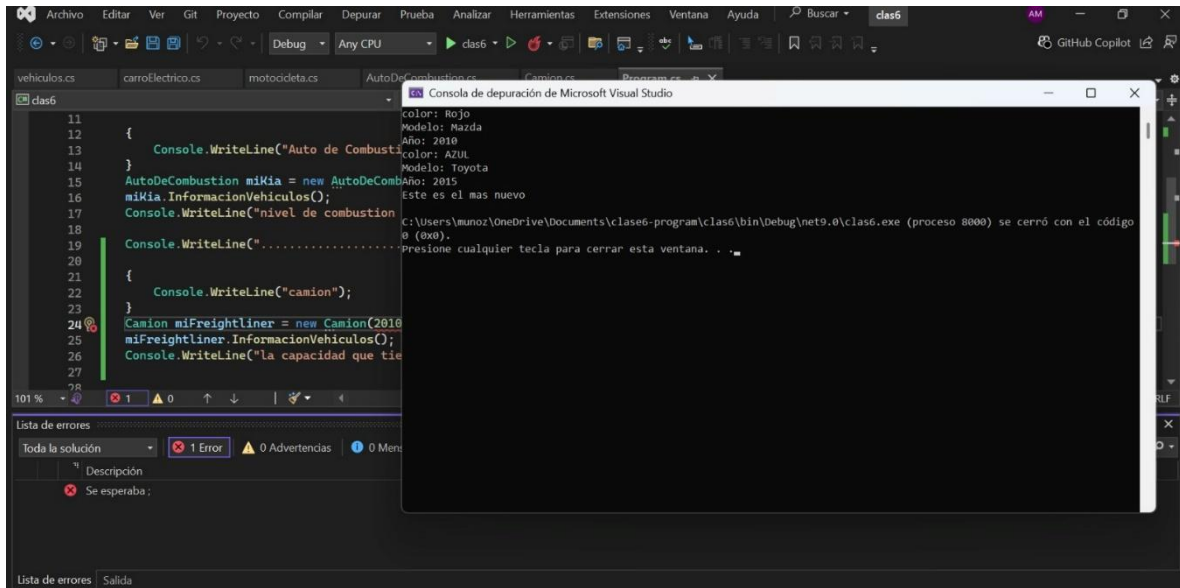
Cuando estaba colocando los datos en el program.cs y lo compilaba no me dejaba me daba errores y cuando me compilaba me ponía información la cual nada que ver con la que yo según estaba poniendo hasta que me percaté que en la clase de vehiculos no había colocado una funcion la cual puse en la clase de camiones

Estas eran las funciones las cuales no me percate que las tenía mal escritas, esta cap. ya las había arreglado, pero antes todas las tenía en minúsculas



```
1 referencia
public int verCargaDelCmaion()
{
    return 20;
}
0 referencias
public int verCargaActual()
{
    return CargaActual;
}
```

Cuando no había arreglado esos errores en consola me aparecían datos los cuales no había puesto



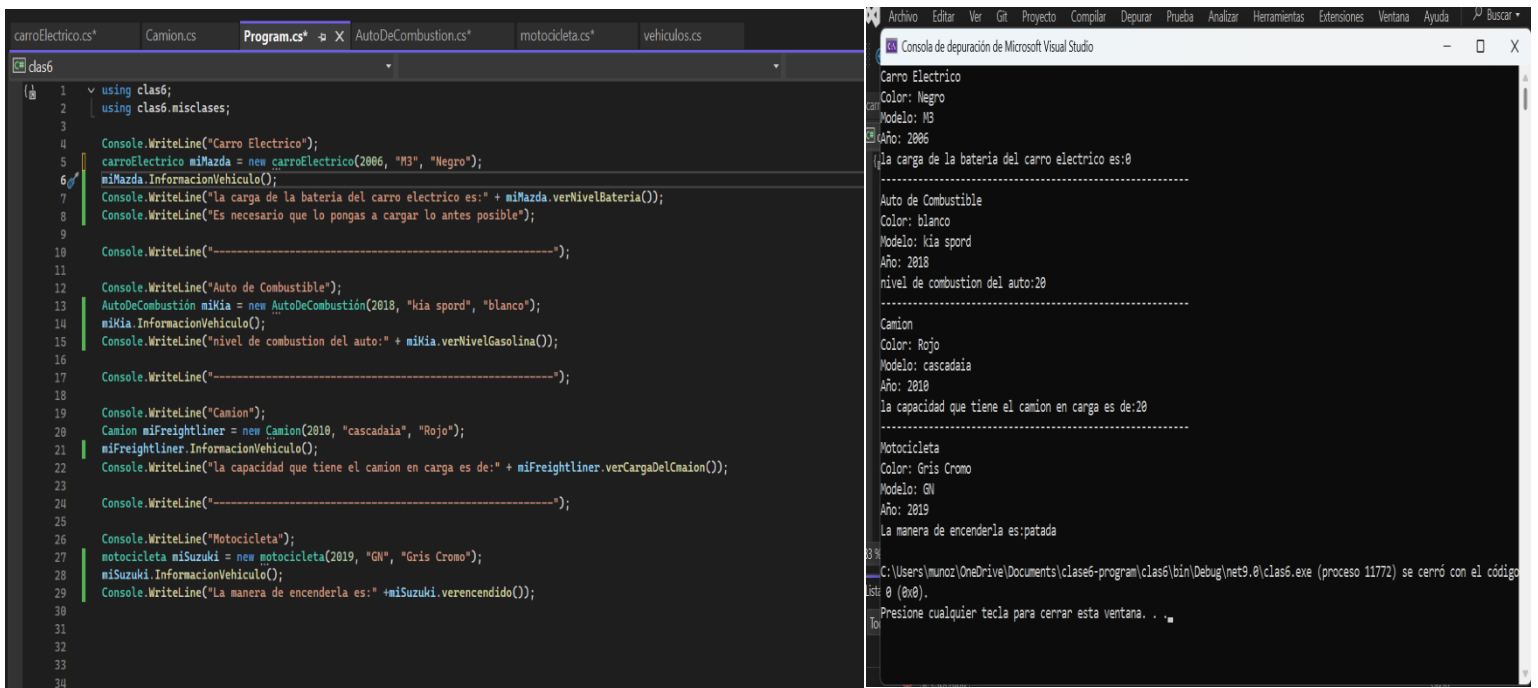
The screenshot shows the Visual Studio IDE with a console window open. The console displays the following output:

```
color: Rojo
Modelo: Mazda
Año: 2010
Color: AZUL
Modelo: Toyota
Año: 2015
Este es el mas nuevo
C:\Users\munoz\OneDrive\Documents\clase6-program\clase6\bin\Debug\net9.0\clase6.exe (proceso 8000) se cerró con el código 0 (0x0).
Presione cualquier tecla para cerrar esta ventana. . .
```

Below the console window, the 'Lista de errores' (List of errors) pane shows one error:

Descripción
Se esperaba ;

esto era lo que me daba en consola cuando no había verificado ese erro, ya luego que lo verifique ya me aparecía lo que yo había colocado en el program.cs



The screenshot shows the Visual Studio IDE with the 'Program.cs' file open. The code in the file is as follows:

```
1 using clas6;
2 using clas6.misclases;
3
4 Console.WriteLine("Carro Electrico");
5 carroElectrico miMazda = new carroElectrico(2006, "M3", "Negro");
6 miMazda.InformacionVehiculo();
7 Console.WriteLine("la carga de la bateria del carro electrico es:" + miMazda.verNivelBateria());
8 Console.WriteLine("Es necesario que lo pongas a cargar lo antes posible");
9
10 Console.WriteLine("-----");
11
12 Console.WriteLine("Auto de Combustible");
13 AutoDeCombustion miKia = new AutoDeCombustion(2018, "kia spord", "blanco");
14 miKia.InformacionVehiculo();
15 Console.WriteLine("nivel de combustion del auto:" + miKia.verNivelGasolina());
16
17 Console.WriteLine("-----");
18
19 Console.WriteLine("Camion");
20 Camion miFreightliner = new Camion(2010, "cascadaia", "Rojo");
21 miFreightliner.InformacionVehiculo();
22 Console.WriteLine("la capacidad que tiene el camion en carga es de:" + miFreightliner.verCargaDelCmaion());
23
24 Console.WriteLine("-----");
25
26 Console.WriteLine("Motocicleta");
27 motocicleta miSuzuki = new motocicleta(2019, "GN", "Gris Cromo");
28 miSuzuki.InformacionVehiculo();
29 Console.WriteLine("La manera de encenderla es:" + miSuzuki.verencendido());
30
31
32
33
34
```

The console window on the right shows the following output:

```
Carro Electrico
Color: Negro
Modelo: M3
Año: 2006
la carga de la bateria del carro electrico es:0
-----
Auto de Combustible
Color: blanco
Modelo: kia spord
Año: 2018
nivel de combustion del auto:20
-----
Camion
Color: Rojo
Modelo: cascadaia
Año: 2010
la capacidad que tiene el camion en carga es de:20
-----
Motocicleta
Color: Gris Cromo
Modelo: GN
Año: 2019
La manera de encenderla es:patada
C:\Users\munoz\OneDrive\Documents\clase6-program\clase6\bin\Debug\net9.0\clase6.exe (proceso 11772) se cerró con el código 0 (0x0).
Presione cualquier tecla para cerrar esta ventana. . .
```

esto era lo que le había agg al program.cs y ello aquí ya me lo daba en consola

como una línea de código o una letra que nosotros la declaremos de una forma y cuando estamos haciendo el código lo pongamos de otra forma nos de ciertos problemas, por eso es bueno verificar bien como las declaramos, ya que si no nos recordamos y lo hacemos solo por hacerlo y ya n sabremos desde donde empezó el problema

