

1. Creación de Clases Derivadas

Para este proyecto, se crearon tres clases derivadas de la clase base Vehiculo:

- **AutoDeCombustion** 🚗
- **Motocicleta** 🏍️
- **Camion** 🚚

Cada una de estas clases hereda de Vehiculo y tiene sus propias propiedades y comportamientos.

2. Encapsulación

Para garantizar la seguridad de los datos, se implementó encapsulación mediante modificadores private en cada clase derivada:

- **AutoDeCombustion:** private int nivelCombustible, private string tipoCombustible, private int capacidadTanque
- **Motocicleta:** private bool tieneCasco, private string tipoMoto, private int cilindrada
- **Camion:** private int capacidadCarga, private int cargaActual, private bool tieneRemolque

Esto protege los valores internos de cada clase, asegurando que solo puedan ser modificados a través de métodos definidos.

3. Sobrescritura de Métodos

Se sobrescribieron los siguientes métodos de la clase Vehiculo para personalizar su comportamiento en cada clase derivada:

- **Acelerar(int cuanto)**
 - En AutoDeCombustion, se reduce el nivel de combustible al acelerar.
 - En Motocicleta, se duplica la aceleración.
 - En Camion, la aceleración es la mitad para reflejar su mayor peso.
- **Frenar(int cuanto)**
 - En AutoDeCombustion, el auto frena normalmente.
 - En Motocicleta, se imprime un mensaje indicando que la frenada es más brusca.
 - En Camion, el frenado es más cuidadoso.
- **Otros métodos adicionales:**
 - AutoDeCombustion incluye CargarCombustible() para llenar el tanque.

- Motocicleta tiene RealizarCaballito() como una acción especial.
- Camion permite agregar carga con Cargar(int peso).

4. Resumen

Este proyecto demuestra el uso de **herencia, encapsulación y polimorfismo** en C# mediante la creación de una jerarquía de clases para distintos tipos de vehículos. Se sobrescribieron métodos de la clase base Vehiculo para ajustar su funcionalidad según las características específicas de cada tipo de vehículo.

El programa principal permite probar cada clase creando instancias y llamando a sus métodos para verificar su comportamiento.