

### ✓ Enunciado General:

Se desea crear un sistema para gestionar diferentes tipos de **vehículos**. Todos los vehículos comparten ciertas características y comportamientos, pero algunos detalles varían según el tipo. Se necesita aplicar conceptos de **clase abstracta**, **herencia**, **polimorfismo** y **manejo de excepciones**.

### 🔧 Parte 1: Clase Abstracta (25 pts)

1. Crea una clase abstracta llamada `Vehiculo`, con los siguientes atributos:
    - `marca` (cadena)
    - `modelo` (cadena)
    - `anio` (entero)
  2. Agrega un método abstracto llamado `calcular_impuesto()` que será implementado por las clases hijas.
  3. Agrega un método común llamado `mostrar_info()` que muestre los datos básicos del vehículo.
- 

### 🚗 Parte 2: Clases Hijas (30 pts)

1. Crea dos clases hijas de `Vehiculo`:
    - `Auto`
    - `Motocicleta`
  2. Cada clase debe:
    - Implementar el método `calcular_impuesto()`:
      - Para **Auto**: el impuesto es  $0.05 * (2025 - anio)$
      - Para **Motocicleta**: el impuesto es  $0.03 * (2025 - anio)$
- 

### 🔄 Parte 3: Polimorfismo (15 pts)

1. Crea una lista de vehículos con objetos de las clases `Auto` y `Motocicleta`.
  2. Recorre la lista usando un bucle y muestra la información de cada vehículo junto con el impuesto correspondiente, usando **polimorfismo**.
- 

### ⚠ Parte 4: Manejo de Excepciones (30 pts)

1. Al crear un vehículo, verifica que el `anio` no sea mayor al año actual (2025) ni menor a 1980. Si lo es, lanza una excepción personalizada llamada `AnioInvalidoError`.
2. Maneja la excepción y muestra un mensaje adecuado sin detener el programa.

### 🔧 Resultado Esperado:

- Uso correcto de una clase abstracta.
- Implementación de clases hijas con métodos específicos.
- Uso de polimorfismo para invocar métodos adecuados.
- Validación de datos y control de errores con excepciones personalizadas.

### 💡 Pregunta Extra (Opcional, +10 pts):

Agrega una clase `Camioneta` que implemente un impuesto diferente:

- Si el modelo es posterior a 2015, paga 500.
- Si no, paga 300.