**Talle 1 Programación Avanzada:**

Este programa modela vehículos, calcula su consumo de gasolina y proporciona una interfaz para administrar los vehículos registrados, dando la opción de agregar vehículos y mostrar los ya registrados.

**Clase Engine:**

Este código define la clase Engine, la cual representa el motor del vehículo y contiene los atributos type, potency y weight.

Propósito: Representa el motor del vehículo

Atributos:

* Type: Tipo del motor (cadena).
* potency: Potencia del motor (número de punto flotante).
* weight: Peso del motor (número de punto flotante).

Métodos:

* \_\_init\_\_(self, type: str, potency: float, weight: float): Constructor que inicializa los atributos del motor.
* \_\_str\_\_(self): Método que devuelve una representación en cadena de texto del motor, incluyendo su tipo, potencia y peso.

**Clase Vehicle:**

La clase padre Vehicle, representa el vehículo y contiene los atributos engine, chasis, model, gas consumption y year.

Propósito: Representa un vehículo completo

Atributos:

* Engine: Objeto de la clase Engine que representa el motor del vehículo.
* chasis: Tipo de chasis (cadena: ‘A’ o ‘B’).
* model: Modelo del vehículo (número entero).
* year: Año del vehículo (número entero).
* gas\_consumption: Consumo de gasolina calculado (número de punto flotante).

Métodos:

* \_\_init\_\_(self, engine: Engine, chasis: str, model: int, year: int): Constructor que inicializa los atributos del vehículo, incluyendo el cálculo del consumo de gasolina.
* calculate\_gas\_consumption(self): Método que calcula el consumo de gasolina según el tipo de chasis y los atributos del motor.
* \_\_str\_\_(self): Método que devuelve una representación en cadena de texto de todos los atributos del vehículo.

**Clases Car, Truck, Yacht y Motorcycle:**

Las siguientes clases heredan de la clase Vehicle:

* Car: Representa un automóvil específico.
* Truck: Representa un camión específico.
* Yacht: Representa un yate específico.
* Motorcycle: Representa una motocicleta específica.

Cada clase heredera puede tener atributos y métodos adicionales específicos según el tipo de vehículo.

**Menú de Interacción:**

El código incluye un menú interactivo donde los usuarios pueden:

* Agregar vehículos: Proporcionar detalles como motor, chasis, modelo y año para crear una instancia de la clase correspondiente (por ejemplo, Car, Truck, etc.).
* Ver vehículos registrados: Mostrar una lista de los vehículos creados junto con sus atributos.
* Salir: Salir del programa.