

Code Explanation Report

This Python code defines a system for creating and managing vehicles of various types. Below is a breakdown of each component and its functionality:

Motor Class:

- Purpose: Represents a motor with specific attributes such as name, type, power, and weight.
- Attributes:
 - `nombre_`: Name of the motor.
 - `tipo_motor_`: Type of the motor (string).
 - `potencia_motor_`: Power of the motor (integer).
 - `peso_`: Weight of the motor (float).
- Constructor:
 - Initializes the attributes of the motor with the provided values.

Vehiculo Class:

- Purpose: Represents a generic vehicle with characteristics like chassis, model, year, and associated motor.
- Attributes:
 - `chasis_`: Chassis of the vehicle (string).
 - `modelo_`: Model of the vehicle (string).
 - `year_`: Year of manufacture of the vehicle (integer).
 - `consumo_de_gas_`: Gas consumption of the vehicle (float).
 - `motor_`: Instance of the Motor class associated with the vehicle.
- Constructor:
 - Initializes the attributes of the vehicle with provided values.
 - Calculates gas consumption based on motor power, weight, and chassis type.
- Methods:
 - `consumo_de_gas_()`: Calculates and prints the gas consumption of the vehicle.

Vehicle Subclasses:

- Subclasses like `Carrito`, `Tractor`, `Camion`, `Yate`, and `Moto` inherit from the `Vehiculo` class.
- They provide specific vehicle types and call the superclass constructor (`Vehiculo`) to initialize attributes.
- `__str__()` method is overridden in each subclass to provide a string representation of the object.

Global Variables:

- `valor_chasis_`: Stores a value based on the type of chassis selected.

Functions:

- `crear_vehiculo_(vehiculo)`: Prompts the user to input motor and vehicle details, creates a new instance of the respective vehicle subclass, and adds it to the list of vehicles.
- `menu()`: Provides a menu interface for creating vehicles, listing all vehicles, or exiting the program.

Main Execution:

- The `menu()` function is called to start the program.
- User input determines the actions to be performed, including creating vehicles, listing vehicles, or exiting the program.