

# Vehicle Management System

## Overview

This document outlines the structure and functionality of a vehicle management system, which includes different types of vehicles and a workshop (`Taller`) that manages these vehicles. The system is designed to handle various operations such as adding, displaying, updating, and removing vehicles, as well as calculating average age and applying maintenance promotions.

## Class Structure

### Vehicle

The `Vehicle` class serves as a base class for different types of vehicles. It contains the following attributes and methods:

- **Attributes:**
  - `-matricula: string` The vehicle's registration number.
  - `-marca: string` The brand of the vehicle.
  - `-model: string` The model of the vehicle.
  - `-any_fabricacio: int` The year of manufacture.
  - `-quilometratge: int` The mileage of the vehicle.
- **Methods:**
  - `+mostrar_info(): void` Displays the vehicle's information.
  - `+matricula: string` Getter for the vehicle's registration number.
  - `+any_fabricacio: int` Getter for the year of manufacture.
  - `+quilometratge: int` Getter for the mileage.
  - `+quilometratge(valor: int): void` Setter for updating the mileage.

### Cotxe (Car)

Inherits from `Vehicle` and includes:

- **Attributes:**
  - `-num_portes: int` Number of doors.
- **Methods:**
  - `+mostrar_info(): void` Displays the car's information.

### Moto (Motorcycle)

Inherits from `Vehicle` and includes:

- **Attributes:**
  - `-tipus: string` Type of motorcycle.
- **Methods:**
  - `+mostrar_info(): void` Displays the motorcycle's information.

## Camio (Truck)

Inherits from `Vehicle` and includes:

- **Attributes:**
  - `-capacitat_carga: int` Load capacity.
- **Methods:**
  - `+mostrar_info(): void` Displays the truck's information.

## Taller (Workshop)

The `Taller` class manages a list of vehicles and provides various functionalities:

- **Attributes:**
  - `-vehicles: List[Vehicle]` A list containing all the vehicles in the workshop.
- **Methods:**
  - `+afegir_vehicle(vehicle: Vehicle): void` Adds a new vehicle to the list.
  - `+mostrar_vehicle(matricula: string): void` Displays information of a specific vehicle by registration number.
  - `+mostrar_tots_vehicles(): void` Displays information of all vehicles.
  - `+actualitzar_quilometratge(matricula: string, quilometres: int): void` Updates the mileage of a specific vehicle.
  - `+eliminar_vehicle(matricula: string): void` Removes a vehicle from the list by registration number.
  - `+llistar_vehicles_per_any(any: int): void` Lists all vehicles manufactured in a specific year.
  - `+calcular_edat_mitjana(): float` Calculates the average age of all vehicles.
  - `+aplicar_promocio_manteniment(tipus_vehicle: string): void` Applies a maintenance promotion to a specific type of vehicle.
  - `-trobar_vehicle(matricula: string): Vehicle` Finds and returns a vehicle by registration number.

## Conclusion

This vehicle management system provides a comprehensive framework for handling various operations related to vehicles in a workshop setting. It supports different vehicle types and offers functionalities to manage and maintain vehicle data efficiently.