**Garage Management System**

**Submitters:**

**- Ilai Shohat – 322445198**

**- Itai Kurzweil – 209025568**

**Ex03.GarageLogic:**

1. GarageManager

- Manages all operations within the garage, including adding, removing, and updating vehicles. It acts as the central hub for maintaining the state of the garage and its vehicles.

2. VehicleGenerator

- Generates vehicle instances with the correct attributes based on predefined properties and user input. Ensures that each vehicle is instantiated consistently.

3. VehicleProperties

- Provides static properties and constants for vehicles, ensuring standardization and consistency in vehicle creation. These properties are used by the VehicleGenerator.

4. ValueOutOfRangeException

- Custom exception for handling cases where input values exceed defined limits. This helps maintain data integrity by preventing invalid data from being processed.

5. Wheel

- Represents a wheel, including attributes such as the manufacturer, current air pressure, and maximum air pressure. It includes methods for inflating the wheel.

6. Engine

- Abstract class for common engine properties, including attributes like maximum capacity and current capacity. Serves as the base class for specific engine types like electric and fuel engines.

7. ElectricEngine

- Inherits from Engine, represents an electric engine with specific properties such as battery life and charge capacity. Includes methods for charging the battery.

8. FuelEngine

- Inherits from Engine, represents a fuel engine with specific properties such as fuel type and current fuel level. Includes methods for refueling.

9. Vehicle

- Abstract class for common vehicle properties, including model name, license number, energy percentage, and a list of wheels. Serves as the base class for specific vehicle types like cars, trucks, and motorcycles.

10. VehicleEntry

- Represents a vehicle entry in the garage, including details such as entry time, owner information, and the vehicle's current status.

11. VehicleRecord

- Maintains a detailed record of each vehicle, including the owner's details and the vehicle's status history. Helps track the lifecycle and maintenance history of vehicles.

12. Car

- Inherits from Vehicle, represents a car with additional attributes like color and number of doors. Includes methods specific to car functionalities.

13. Truck

- Inherits from Vehicle, represents a truck with additional attributes like cargo volume and hazardous materials indicator. Includes methods specific to truck functionalities.

14. Motorcycle

- Inherits from Vehicle, represents a motorcycle with additional attributes like engine volume and license type. Includes methods specific to motorcycle functionalities.

**Ex03.ConsoleUI:**

1. Program

- Entry point for the application. It contains the Main method, which initializes the system and runs the garage management operations using the Runner class.

2. Runner

- Manages the overall flow of the program. It interacts with the user through the ConsoleRenderer, processes user inputs, and invokes the appropriate methods in GarageManager to perform operations like adding, removing, and querying vehicles.

3. ConsoleRenderer

- Handles all aspects of rendering information to the console. This includes displaying menus, prompts, and responses to the user. It ensures that the user interface is clear and user-friendly.

4. UiValidator

- Validates user inputs to ensure they meet the required format and constraints. This helps prevent errors and ensures that the data entered by the user is valid and can be processed by the system.

5. MenuOptions

- Manages the options displayed in the menu. It allows users to navigate through different functionalities of the garage management system, such as adding vehicles, viewing vehicle details, and updating vehicle statuses.

**Enums:**

1. VehicleStatus

- Possible statuses for a vehicle in the garage

2. CarDoors

- Possible door counts for cars.

3. CarColor

- Possible colors for cars.

4. MotorcycleLicense

- Possible license types for motorcycles.

5. FuelType

- Possible fuel types for engines.

6. VehiclesType

- Possible current vehicle types known to the system.

