RED & WHITE®
Multimedia Education
Shaping "skills" for "scaling" higher...!!!

C++

Project - 3

Vehicle Registry System

RED & WHITE MULTIMEDIA EDUCATION
Shaping "skills" for "scaling" higher…!!!

From the Headquarter of RNW
Surat, Gujarat, India
https://www.rnwmultimedia.edu.in

Project Definition: Vehicle Registry System

Overview:

Develop a Vehicle Registry System using C++ that demonstrates all types of inheritance and encapsulation. The system will allow users to manage a registry of different types of vehicles.

Time Allocation:

- Total Duration: 4 Hours
- Total Marks: 10

Instructions:

1. Attempt all assigned tasks.
2. Make suitable assumptions wherever necessary.
3. Upload your exam task by uploading the project to GitHub and submitting the GitHub repository link which must have screenshots of your output in a README.md file.
4. This project is individual-based; copying code from classmates is prohibited.

Remember to follow the instructions provided professionally, make suitable assumptions wherever necessary, and avoid copying code or content from any unauthorized sources. Good luck with your project work!

Project Criteria:

Requirements:

- Class & Object
- Static member
- Constructors
- Destructor
- Array of objects
- Getters & Setters
- Encapsulation
- All Types of Inheritance (Single, Multiple, Multilevel, Hierarchical)

Implementation Details:

1. **Base Class:** Vehicle
   Attributes:
   - vehicleID
   - manufacturer
   - model
   - year

Static Member:

- totalVehicles

Methods:

- Constructors
- Destructor
- Getters and Setters

2. **Derived Classes:**

Car (Single Inheritance from Vehicle)

- Additional attribute: fuelType

ElectricCar (Multilevel Inheritance from Car)

- Additional attribute: batteryCapacity

Aircraft (Base Class for Multiple Inheritance)

- Additional attribute: flightRange

FlyingCar (Multiple Inheritance from Car and Aircraft)

SportsCar (Multilevel Inheritance from ElectricCar)

- Additional attribute: topSpeed

Sedan (Hierarchical Inheritance from Car)

SUV (Hierarchical Inheritance from Car)

3. **VehicleRegistry Class:**
   - Manages an array of Vehicle objects.
   - Methods for adding, displaying, and searching vehicles.

4. **Main Function:**
   - Menu-driven approach to interacting with the user and performing various vehicle registry operations:
     - Add a vehicle (of a different type)
     - View all vehicles
     - Search by Id
     - Exit

Marking Criteria: (Total 10 Marks)
Logic: 8 Mark
Output: 2 Mark

Vehicle Registry System
C++