

Coding Marathon Problem Statement 1

Create a class "Car" with the following attributes

- carChassisNumber of type string
- carBrand of type string
- carModel of type string
- carCategory of type enum class CAR_CATEGORY which can include the following literals
 - SEDAN , SUV OR SPORTS_HATCHBACK
- carPrice of type float
- carSeatCount of type integer
- carFuelCapacity of type integer
- A parameterized constructor that takes all parameters mentioned above.
- A parameterized constructor that takes 4 parameters carChassisNumber, carModel, carBrand & carPrice
- The default constructor must be disabled.
- This car should allow shallow copy of object's data using a defaulted copy constructor
- An overload of << operator must be added to display data of an object.
- An overload of + operator must be added. When 2 objects of this Car type are added, the combined carPrice of 2 car objects used with + operator "must be returned from the function".
- Getters & Setters for all attributes.

Create a functionalities.cpp file with the following top-level functions

- **FindAverageFuelCapacity** that accepts a primitive array of Car type pointers and returns the average value of carFuelCapacity attribute of the Car objects whose pointers are stored in the list.
- **FindCountOfGivenBrand** that accepts
 - A primitive array of Car type pointers.
 - A enum class CAR_CATEGORY value

The function must return the total count of Cars of given value type present in the list passed as the first parameter.

- **FindBestPassengerCar** that accepts
 - A primitive array of Car type pointers.
 - A enum class CAR_CATEGORY value.

The function must display the carBrand and carModel of the car that has "the highest number of seats (carSeatCount) among all cars whose type is the same as passed CAR_CATEGORY value.

Create a Main.cpp file with code to demonstrate each function from the functionalities.cpp file.