Task 1Z: Implement a Basic Class (Car Class in C++)

Objective:
The objective of this task is to design and implement a basic C++ class named Car. This task demonstrates fundamental Object-Oriented Programming (OOP) concepts such as encapsulation, access specifiers, and the use of getter and setter member functions.

Problem Statement

Create a Car class with the following specifications:
- Private data members: make, model, and year
- Public member functions to set and retrieve the values of these attributes

Concepts Used

- Class and Object
- Encapsulation
- Private and Public Access Specifiers
- Getter and Setter Methods

Source Code (C++)

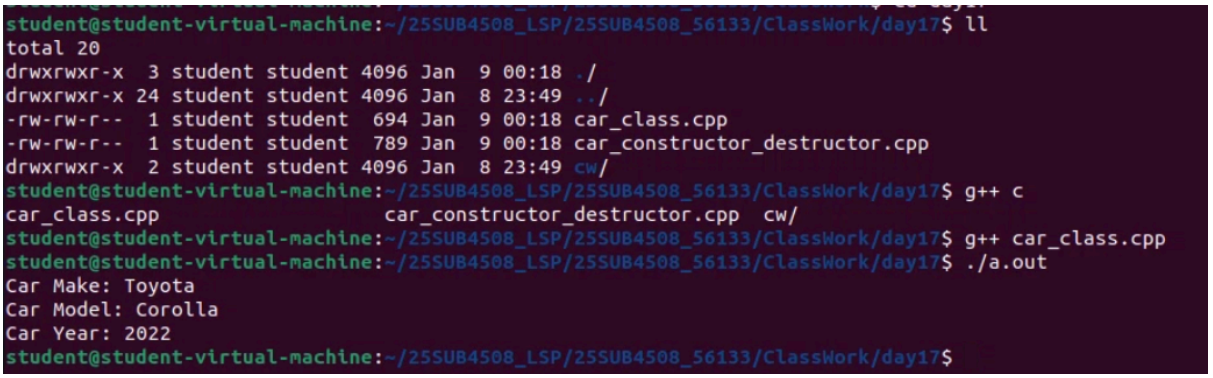
```
#include <iostream>
#include <string>
using namespace std;

class Car {
private:
        string make;
        string model;
        int year;

public:
        void setMake(string m) {
        make = m;
        }
```

```
        void setModel(string mo) {
        model = mo;
        }

        void setYear(int y) {
        year = y;
        }

        string getMake() {
        return make;
        }

        string getModel() {
        return model;
        }

        int getYear() {
        return year;
        }
};

int main() {
        Car car1;

        car1.setMake("Toyota");
        car1.setModel("Corolla");
        car1.setYear(2022);

        cout << "Car Make: " << car1.getMake() << endl;
        cout << "Car Model: " << car1.getModel() << endl;
        cout << "Car Year: " << car1.getYear() << endl;

        return 0;
}
```

Explanation of the Code

1. A class named Car is created using the class keyword.
2. The data members make, model, and year are declared as private to ensure data hiding.
3. Public setter methods (setMake, setModel, setYear) are used to assign values to private members.
4. Public getter methods (getMake, getModel, getYear) are used to access private data safely.
5. In the main() function, an object car1 is created.
6. Setter methods are called to initialize the car details.
7. Getter methods are called to display the stored values.

Program Output



Conclusion

This task successfully demonstrates the implementation of a basic C++ class with proper encapsulation. By using private data members and public getter and setter methods, data security and controlled access are achieved, which are core principles of Object-Oriented Programming.