1. **Car_Factory Abstract Class:**
    - Fields: **company**, **car_name**, and **budget**.
    - Abstract methods: **getprice(double price)**, **detail(String company_name, String car_name)**, and **accessories()**.
    - **input** method to get user input for company, car name, and budget.
    - **display** method to display car details and accessories based on the implemented abstract methods.
2. **Small_car, Sedan, Luxary Classes (extend Car_Factory):**
    - Fields: **Ans** to determine whether certain accessories are available.
    - Implement abstract methods from **Car_Factory** and provide specific details and accessories for each car type.
3. **factory Class (main class):**
    - Contains the main program logic and user interaction loop.
    - Creates instances of **Small_car**, **Sedan**, and **Luxary** based on user choice.
    - Calls the **input** and **display** methods to get user input and display car details and accessories.
    - The loop continues until the user chooses to exit.

Explanation of Operations:

- The user can choose between Small Car, Sedan, and Luxury Car.
- For each car type, the program prompts the user to enter details such as company, car name, and budget.
- It then displays details and accessories based on the implemented methods in the respective car class.
- The program runs in a loop until the user chooses to exit.

In summary, this program demonstrates the use of abstract classes and polymorphism to create a simple car factory system with different types of cars, each having specific details and accessories. The user can interact with the program to view details of various cars