1. Single Inheritance

Problem Statement:
Develop an application for a bookstore where you need a base class `Book` with attributes like `title`, `author`, and `price`. Derive a subclass `EBook` that adds additional attributes such as `fileSize` and `format`. Implement methods to display book details and eBook-specific details.

Expected Classes:

- `Book` (Base class)
- `EBook` (Derived class)

---

2. Multiple Inheritance (Using Interfaces)

Problem Statement:
Create a multimedia player system where a `SmartDevice` should support both audio and video functionalities. Implement two interfaces: `AudioPlayer` (with methods like `playAudio()`, `stopAudio()`) and `VideoPlayer` (with methods like `playVideo()`, `stopVideo()`). Implement these interfaces in the `SmartDevice` class.

Expected Classes/Interfaces:

- `AudioPlayer` (Interface)
- `VideoPlayer` (Interface)
- `SmartDevice` (Implements both interfaces)

---

3. Hierarchical Inheritance

Problem Statement:
Design a vehicle classification system. Implement a base class `Vehicle` with attributes `brand` and `speed`. Derive two subclasses `Car` (with attributes like `fuelType`) and `Bike` (with attributes like `engineCapacity`). Implement methods to display details for each type.

Expected Classes:

- `Vehicle` (Base class)
- `Car` (Derived class)
- `Bike` (Derived class)

---

4. Hybrid Inheritance (Combination of Hierarchical and Multiple Inheritance Using Interfaces)

Problem Statement:
Build a smart home automation system that combines multiple functionalities. Implement a base class `SmartDevice` with attributes like `deviceName` and `powerConsumption`. Create two interfaces `RemoteControl` (methods like `turnOn()`, `turnOff()`) and `InternetConnectivity` (methods like `connectToWiFi()`). Derive two classes `SmartTV` (which extends `SmartDevice` and implements both interfaces) and `SmartLight` (which implements `RemoteControl`).

Expected Classes/Interfaces:

- `SmartDevice` (Base class)
- `RemoteControl` (Interface)
- `InternetConnectivity` (Interface)
- `SmartTV` (Extends SmartDevice, Implements RemoteControl and InternetConnectivity)
- `SmartLight` (Implements RemoteControl)

---

5. Interface Implementation

Problem Statement:
Develop a transportation system that supports different types of vehicles. Create an interface `Transport` with methods like `start()`, `stop()`. Implement this interface in two classes `Bus` and `Train`, each providing specific implementations.

Expected Classes/Interfaces:

- `Transport` (Interface)
- `Bus` (Implements Transport)
- `Train` (Implements Transport)
