**1️⃣ Library Management System**

📌 **Concepts Covered:**  
✔ OOP (**Classes, Objects, Inheritance**)  
✔ **Generics (Using Lists and Maps)**  
✔ Exception Handling

**Project Description:**

Create a simple **Library Management System** where users can **borrow and return books**. The system should track **available books and borrowed books** using collections.

**Requirements:**

* Create a **generic class** Library<T> that stores books in a List<T>.
* A Book class with **title, author, and book ID**.
* Allow users to **borrow and return books**.
* Handle cases where a book is not available.
* Display **all available books**.

**2️⃣ Student Gradebook System**

📌 **Concepts Covered:**  
✔ **Generics (Storing different student data types)**  
✔ **Encapsulation & Inheritance**  
✔ Exception Handling

**Project Description:**

Develop a **Student Gradebook System** where instructors can **add students and their grades**.

**Requirements:**

* Create a Student class with **name, ID, and a list of grades**.
* Use a **generic class** Gradebook<T> to store students.
* Allow **adding and removing students**.
* Compute and **display the average grade** for each student.

**3️⃣ Online Shopping Cart**

📌 **Concepts Covered:**  
✔ **Generics (Cart with different product types)**  
✔ **Encapsulation & OOP**  
✔ **Collections (List & Map)**

**Project Description:**

Develop a **Shopping Cart System** where users can **add products** and **calculate total price**.

**Requirements:**

* Create a Product class with **name, price, and category**.
* A ShoppingCart<T> generic class that stores **products**.
* Allow adding and removing products from the cart.
* Calculate the **total price**.

**4️⃣ Bank Account System**

📌 **Concepts Covered:**  
✔ **OOP (Inheritance: SavingsAccount & CurrentAccount)**  
✔ **Generics (Transaction Logs)**  
✔ Exception Handling

**Project Description:**

Create a **Bank Account System** where users can **deposit, withdraw, and check balances**.

**Requirements:**

* Create an **abstract class** Account with deposit() and withdraw().
* Subclasses: SavingsAccount (higher interest) and CurrentAccount (lower interest).
* A Transaction<T> class to log deposits and withdrawals.
* Display **account balance and transactions**.