***CUSTOM EXCEPTION HANDLING IN C#***

**Part 1: Simple Custom Exception – Age Validation**

**Scenario:**

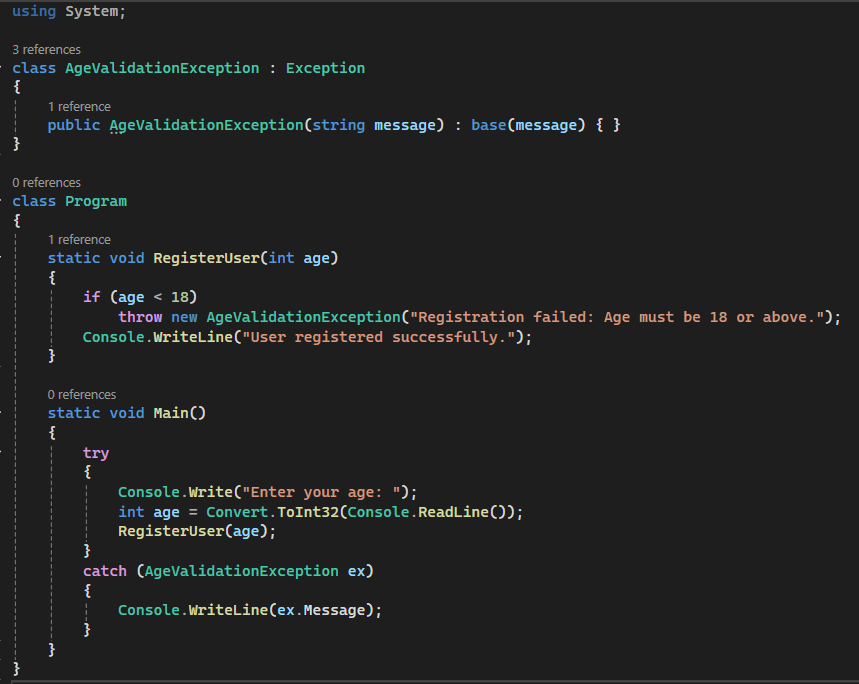
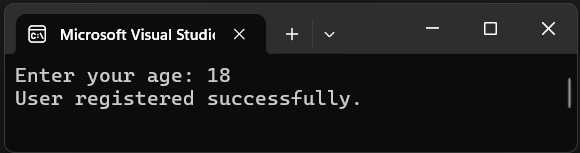
A government portal allows user registration only if the user is above 18 years.

**Tasks:**

Create a custom exception class AgeValidationException.

Create a method RegisterUser(int age) that throws AgeValidationException if age is less than 18.

In Main(), take age input and call RegisterUser(). Catch and display a meaningful message.

**** **** 

**Part 2: Banking System – Insufficient Balance Exception**

**Scenario:** A bank application must handle insufficient balance while withdrawing money.

**Tasks:**

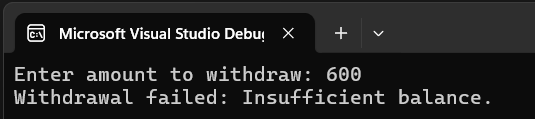
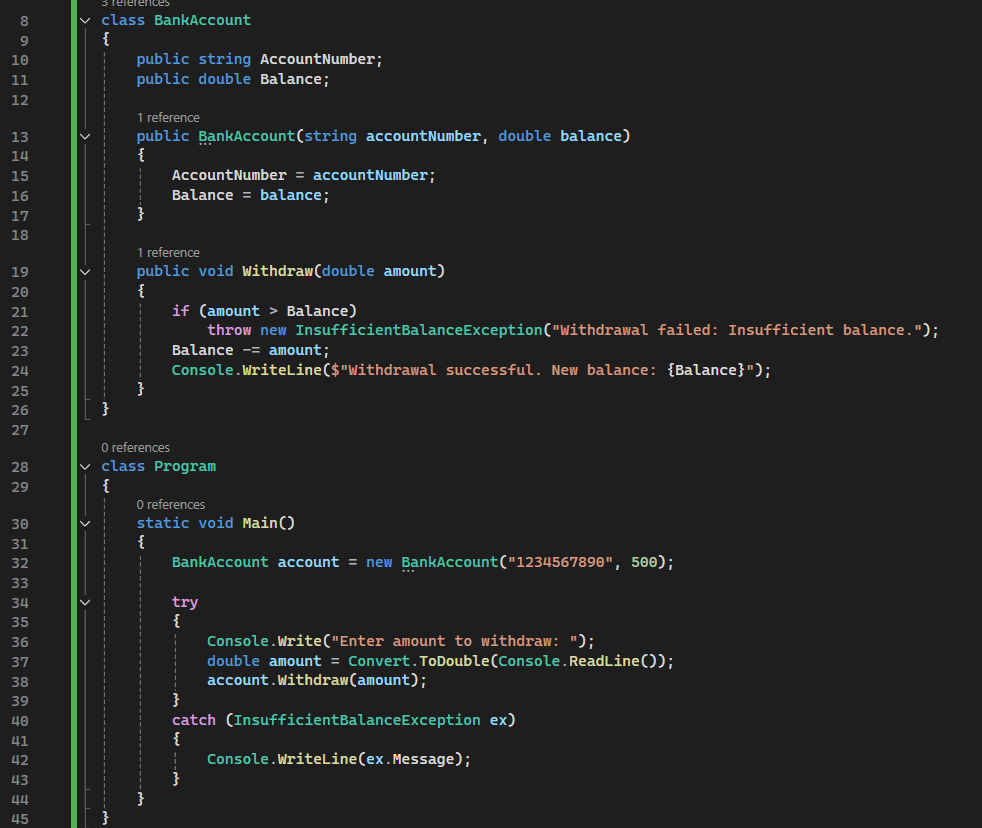
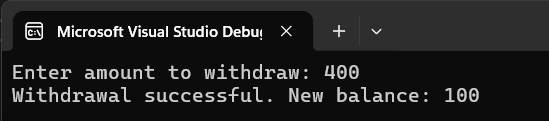
Create a class InsufficientBalanceException inheriting from Exception.

Define a class BankAccount with:

Fields: AccountNumber, Balance

Method: Withdraw(double amount) which throws InsufficientBalanceException if balance is not enough.

Use try-catch block to handle the exception.

** **

**Part 3: Student Management – InvalidGradeException**

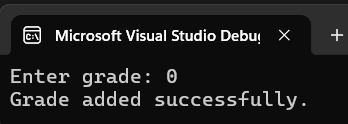
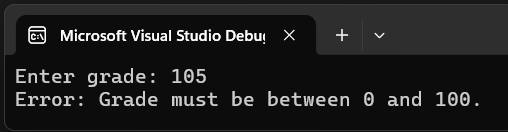
**Scenario:** While entering student grades, the system should only accept grades between 0 and 100.

**Tasks:**

Create a class InvalidGradeException.

Define a class Student with method AddGrade(int grade) that throws InvalidGradeException if grade is invalid.

Demonstrate exception handling in Main() with different test cases.

**Part 4: Custom Exception with Inner Exception**

**Scenario:**

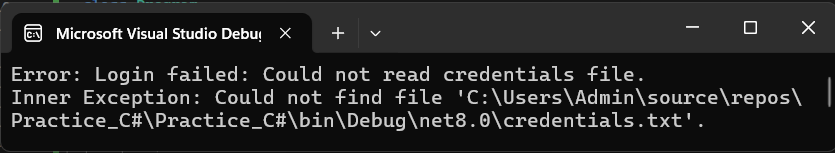
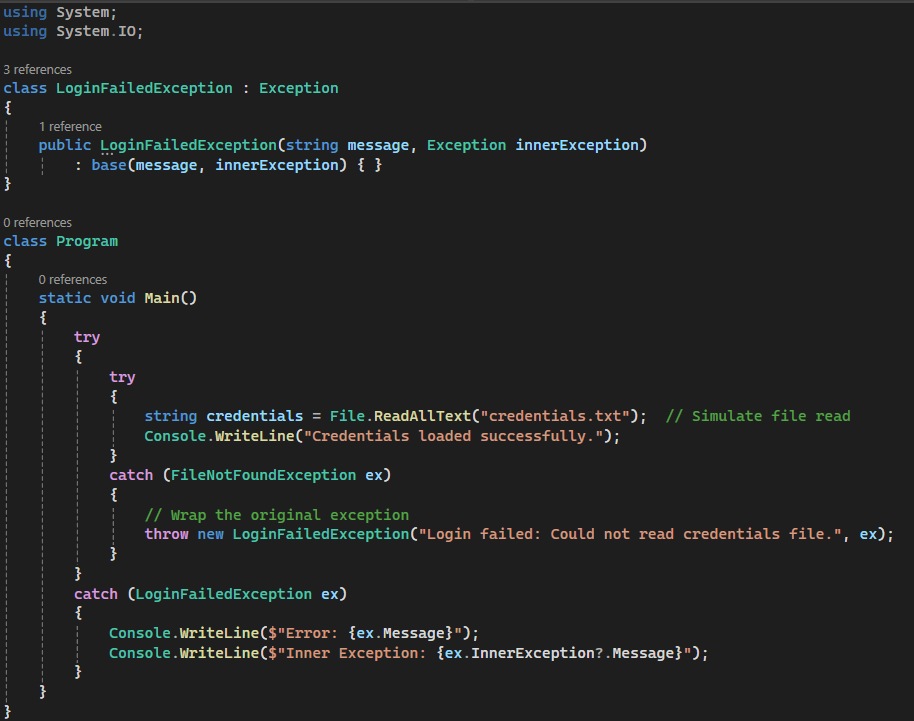
In a login system, if reading credentials from a file fails, you want to throw a custom LoginFailedException with the original file read exception as an inner exception.

**Tasks:**

Create a LoginFailedException that accepts inner exception.

Simulate file reading with File.ReadAllText() and catch FileNotFoundException.

Throw LoginFailedException with the caught exception as inner.



**Bonus:** Create a Custom Base Exception

Create a base class ApplicationCustomException and inherit all other custom exceptions from it. Demonstrate polymorphic catching using this base type.

