Java Coding Question: Exception Handling

Question

Write a Java program that simulates a simple banking system. The program should include the following features:

1. **Account Class**:
   - A class named `Account` with private instance variables `balance` (of type `double`) and `accountNumber` (of type `String`).
   - A constructor that takes `accountNumber` and an initial `balance` as parameters.
   - A method named `deposit(double amount)` that adds the specified amount to the balance.
   - A method named `withdraw(double amount)` that subtracts the specified amount from the balance.
2. **Custom Exceptions**:
   - Define a custom exception named `InsufficientFundsException` that extends `Exception`. This exception should be thrown when an attempt to withdraw an amount greater than the balance is made.
   - Define a custom exception named `InvalidAmountException` that extends `Exception`. This exception should be thrown when an attempt to deposit or withdraw a negative amount is made.
3. **Main Class**:
   - A `BankingSystem` class with a `main` method.
   - Create an instance of the `Account` class.
   - Implement exception handling in the `main` method to catch and handle `InsufficientFundsException` and `InvalidAmountException`.

Detailed Instructions

1. **Account Class**:
   - Create a class named `Account` with the required instance variables.
   - Implement the constructor to initialize `accountNumber` and `balance`.
   - Implement the `deposit` method to add the specified amount to the balance. If the amount is negative, throw an `InvalidAmountException`.
   - Implement the `withdraw` method to subtract the specified amount from the balance. If the amount is negative, throw an `InvalidAmountException`. If the amount is greater than the balance, throw an `InsufficientFundsException`.
2. **Custom Exceptions**:
   - Create a class named `InsufficientFundsException` that extends `Exception`. Implement a constructor that accepts a message and passes it to the superclass constructor.
   - Create a class named `InvalidAmountException` that extends `Exception`. Implement a constructor that accepts a message and passes it to the superclass constructor.
3. **Main Class**:
   - Create a class named `BankingSystem` with a `main` method.
   - Inside the `main` method, create an `Account` object with an initial balance.
   - Use a `try-catch` block to handle `InsufficientFundsException` and `InvalidAmountException`.
   - Print appropriate messages when exceptions are caught.