

SPOT Question

Date : 20-08-2025

Implement a class `BankAccount` with:

- A private `double balance` field.
- A constructor that initializes the balance.
- Methods:
 - `void deposit(double amount)` — adds the amount to balance but throws `NegativeAmountException` if amount is negative.
 - `void withdraw(double amount)` — subtracts the amount from balance but:
 - Throws `NegativeAmountException` if amount is negative.
 - Throws `InsufficientFundsException` if `amount > balance`.

• In the `main` method:

- Create a `BankAccount` instance with an initial balance.
- Simulate a series of deposits and withdrawals by calling the respective methods.
- Use nested `try-catch` blocks to:
 - Handle `InsufficientFundsException` specifically.
 - Handle `NegativeAmountException`.
 - Handle any other unexpected exceptions.
- Ensure that after every transaction, the program prints the current balance.
- Ensure that resources like scanner or any I/O are properly closed in a `finally` block.