Banking Problem

Consider an object-oriented design for the following problem. You need to store information for bank accounts. The goal for the problem is to avoid duplicating code between the three types of account.

An account needs to respond to the following messages:

1. `void deposit(double amt)`: adds the `amt` to the balance and increment the number of transactions

2. `void withdraw(double amt)`: subtract the `amt` to the balance and increment the number of transactions

3. `double getBalance()`: returns the current balance

4. `void endOfMonthCharges()`: the account will be sent this message once a month, so it should levy any monthly fees at that time and prints out the account monthly summary.

There are three types of account:

1. Normal: deposit and withdraw just affect the balance. There is a $5.00 monthly fee which counts as a normal withdrawal transaction.

2. NickleDime: Each withdrawal generates a $0.50 fee. Both the withdrawal and the fee count as a transaction.

3. Gambler: A withdrawal returns the requested amount of money- however the amount deducted from the balance is as follows: there is a 0.49 probability that no money will actually be subtracted from the balance. There is a 0.51 probability that twice the amount actually withdrawn will be subtracted.

Assignment

To come up with 4 classes: Account, Normal, NickleDime, Gambler, such that it conforms to above specification. Something’s to consider:

1. What data fields are needed for any account? Who can access them?

2. Which methods need overriding? Do we want to just use overriding of methods or can we have some abstract methods?