Q1 Answers:
-----------

So, the variables in object of `DodgyBankAccount` class can be directly accessed externally and changed, ultimately changing the amount of cash held in the account,
whereas the `SecureBankAccount` class/object disallows external access this.

This is because within an object of `DodgyBankAccount` class, the internal variables are declared to be publicly accessible so any 'outside' class can access and change the values.
With an object of `SecureBankAccount`class, not only are the variables all declared private, _ accountNumber is a readonly string, and REWARD_AMOUNT is a const int.
Once instantiated, they cannot be changed dynamically - these two values would otherwise present a security risk. Furthe5, they will rarely, if ever, change, and should not do so via a publicly accessible method.

It also can also be seen that the `SecureBankAccount` class contains a call to object method AddReward() which is declared private, thus any call to it from any object outside the class in which it is itself declared is not allowed.
Compiler will refuse to build this code. In contrast, the object of`DodgyBankAccount` class is able to access the same method as it is declared public, with dodgy results.

| | |
|---|---|
| As far as method names go, one that catches the eye is : | private void AddReward() |
| Perhaps as a private method it might be declared: | private void addReward() |
| Or even: | private void addRewardToAccount() |
| Similarly, the other methods might have fuller names, ie: | DepositAmount(int amount) |
| might become: | DepositAmountInAccount(int deposit_amount) |
| Similarly: | DisplayAccountBalanceDetails() |
| might become: | DisplayAmountInAccount() |
| and: | DebitAmount(int amount) |
| becoming: | DebitAmountFromAccount(int deposit_amount) |