Exercises  
In this exercise you will create a CSV file based on a set of transactions stored in a current account.  
1. To do this first define a new Account class to represent a type of bank account.

2. When the class is instantiated you should provide the account number, the name of the account holder, an opening balance and the type of account (which can be a string representing 'current', 'deposit' or 'investment' etc.). This means that there must be an \_\_init\_\_ method and you will need to store the data within the object.

3. Provide three instance methods for the Account; deposit(amount), withdraw(amount) and get\_balance(). The behaviour of these methods should be as expected, deposit will increase the balance, withdraw will decrease the balance and get\_balance() returns the current balance. Your Account class should also keep a history of the transactions it is involved in.

A Transaction is a record of a deposit or withdrawal along with an amount. Note that the initial amount in an account can be treated as an initial deposit

The history could be implemented as a list containing an ordered sequence to transactions. A Transaction itself could be defined by a class with an action (deposit or withdrawal) and an amount.

Each time a withdrawal or a deposit is made a new transaction record should be added to a transaction history list.

Next provide a function (which could be called something like write\_account\_transactions\_to\_csv()) that can take an account and then write each of the transactions it holds out to a CSV file, with each transaction type and the transaction amount separated by a comma.

The following sample application illustrates how this function might be used:  
print('Starting')  
acc = accounts.CurrentAccount('123', 'John', 10.05, 100.0)  
acc.deposit(23.45)  
acc.withdraw(12.33)  
print('Writing Account Transactions')  
write\_account\_transaction\_to\_csv('accounts.csv', acc)  
print('Done')  
The contents of the CSV file would then be:

