

# Transactions

---

In bitcoin and Ethereum you can send funds from one account to another. This activity is captured in a transaction.

In this assignment we are going to add transactions to our blockchain.

You will need to implement code in `./cli/cli.go` around line 200. The function is `func (cc *CLI) SendFundsTransaction(`

Side Note: notice how line continuation works in go with the declaration of the function.

The function calls `cc.InstructorSendFundsTransaction` remove that. That is the instructors version of the code (The answer that I implemented).

Work through the pseudo code and implement the transaction.

Basically a transaction is finding all the outputs for an account that do not have any corresponding input. These are the unused outputs that represent the value of the account. Fortunately we have an index that tells us where to find these. Verify that there is sufficient funds in the account then create inputs for our new block/new transaction that collects all of the funds. Given that we have the sum of the funds, now create 1 or 2 output transactions. First a transaction output to the destination to account. If the amount of the funds is larger than the transferred amount then some "change" is owed back to the "from" account. If "change" is needed then create a transaction output with the "change".

The pseudo code is in a comment in the file `./cli/cli.go`.