

Python Course  
Cryptocurrency Project  
February 15, 2019

By Jennifer Jenkins and Barbara Bellafiore

Github  
jjnkns/jenbar\_repo

Folder structure

**jenbar\_repo** — folder

app.py — this is the python file. It controls almost everything

**templates** - folder with each of the html files

index.html — default landing page

<http://localhost/5000>

buy.html — page that comes up when user clicks Buy link

<http://localhost/5000/buy>

sell.html — page that comes up when user clicks Sell link

<http://localhost/5000/sell>

view\_acct.html - page that comes up when user clicks Account link

[http://localhost/5000/view\\_acct](http://localhost/5000/view_acct)

form.html - page that comes up when user clicks Register link

[http://localhost/5000/view\\_acct](http://localhost/5000/view_acct)

**static** — folder

bitcoin.png - image of Bitcoin

ethereum.png — image of Ethereum

litecoin.png — image of Litecoin

dollar.png — image of US dollar

**css** - folder with bootstrap cascading style sheets

.css files

Mysql schema  
jenbar

## MySQL tables

- customer — name and personal info
- cust\_account — account number and balance — one account per currency
- cust\_account\_assoc — link between customer and cust\_account
- currency — the currencies relevant for this project
- acct\_transaction - buy and sell history

## MySQL views

- customer\_balance — joins customer, currency and account tables
- transaction\_log — for future use. Not yet displayed on pages

See Entity Relationship Diagram and table creation scripts on Github

jenbar\_err.pdf  
jenbar\_table\_create.sql

## Main Screen (index.html)

Spot Prices as of <date time>

*Spot prices come from Coinbase api through python get\_price function*

```
@app.route('/')  
def main_page():
```

Image of BitCoin bitcoin.png	Image of LiteCoin litecoin.png	Image of Ethereum ethereum.png
Spot price of BTC-USD centered	Spot price of LTC-USD centered	Spot price of ETH-USD centered
Coindesk hyperlink with price history	Coindesk hyperlink with price history	Coindesk hyperlink with price history

## Buy Screen (buy.html)

Buy Prices as of <date time>

*Buy prices come from Coinbase api through python get\_price function*

```
@app.route('/buy', methods=['GET', 'POST'])  
def buy():
```

Image of BitCoin bitcoin.png	Image of LiteCoin litecoin.png	Image of Ethereum ethereum.png
Buy price of BTC-USD centered	Buy price of LTC-USD centered	Buy price of ETH-USD centered
Desired quantity	Desired quantity	Desired quantity
Price * Quantity	Price * Quantity	Price * Quantity

## Sell Screen (sell.html)

Buy Prices as of <date time>

*Buy prices come from Coinbase api through python get\_price function*

```
@app.route('/sell', methods=['GET', 'POST'])
def sell():
```

Image of BitCoin bitcoin.png	Image of LiteCoin litecoin.png	Image of Ethereum ethereum.png
Sell price of BTC-USD centered	Sell price of LTC-USD centered	Sell price of ETH-USD centered
Price * Quantity	Price * Quantity	Price * Quantity

\*Python checks to see if user inventory is  $\geq$  what they have to sell. If not, transaction does not happen.

*User feedback message not yet implemented.*

*Confirmation action not yet implemented.*

*Submit executes the sale and writes it to the database.*

## Registration Screen (form.html)

Lets a new user create accounts.

After entering user information four accounts get created in MySQL

Bitcoin Account - balance 0

Litecoin Account - balance 0

Ethereum Account - balance 0

USD Account to be used for purchases. Balance \$1,000,000

*pretend the money was deposited through other mechanism*

The MySQL Customer table auto-generates a unique customer id

*Feedback to user confirming account creation and what their customer number is not yet implemented.*

```
@app.route('/form', methods=['GET', 'POST'])
def form_login():
```

1. Create account form - write to customer database table and generate unique customer id / account number. User name, customer id, credit card account or bank, expiration date.

## Account Screen (view\_account.html)

Buy Prices as of <date time>

*Buy prices come from Coinbase api through python get\_price function*

```
@app.route('/sell', methods=['GET', 'POST'])
```

```
def sell():
```

Image of BitCoin bitcoin.png	Image of LiteCoin litecoin.png	Image of Ethereum ethereum.png	Image of Dollar ethereum.png
Quantity in account	Quantity in account	Quantity in account	Quantity of US dollars in Cash Account
Additional details tbd	Additional details tbd	Additional details tbd	Additional details tbd

*Display of transaction logs not yet implemented. In MySQL only now.*

*Market value (appropriate price \* quantity) not yet implemented.*

## Key Python Functions in app.py

```
def get_connection():  
    #returns a connection object
```

```
def add_customer(first_name, middle_name, last_name, user_name,  
email_address):  
    #create an customer with 4 accounts—one for USD cash and  
one for each crypto
```

```
def get_price(currency_type, price_type):  
    my_url = 'https://api.coinbase.com/v2/prices/'  
    my_url = my_url+currency_type+"/"+price_type  
    response=requests.get(my_url)  
    data = response.json()  
    price = data["data"]["amount"]  
    return price
```

```
def make_trade(customer_id, account_id, currency_short_name,  
currency_id, quantity, side):  
    #update the transaction table and the account table
```

Use for buy or sell. Gets latest price

```
def get_account_balance(customer_id, currency_id):  
    #find out how much user has in each account  
    sql = ("Select account_id, currency_symbol, quantity,  
last_update from customer_balance where customer_id=%s and  
currency_id=%s")
```

## **Enhancements and Corrections**

The fundamental elements of The CryptoCurrency Store are in place, including:

- Home page with live feed of updated prices for all three currencies, BTC, ETH,LITE
- Nav Bar available on all pages which connects to the Buy and Sell Transaction pages, and to the Register (Create Account) form. The Registration process assigns an Customer ID to each Customer, that must be input at every transaction for security purposes.
- Buy and Sell transaction pages automatically update the balances in all three currencies and in USD.

Going forward, here are the enhancements that are in development to be addeed:

### **A. Transaction Enhancements**

#### **1. Confirmation Loop on the Transaction pages (Buy, Sell):**

Before completing the transaction, a pop-up window will ask "are you sure you want to (buy or sell) (the specifc currency)?"

#### **2. Error Control:**

There will also be an "error control" function to ensure the seller has enough crypto to sell or enough dollars to buy, through the use of a drop down window that will say: "You do not have enough (specific currency) for this transaction. You have (number) (crypto currency coins) now worth approximately (\$XXX).

This "error control" function requires an active and customized connection to the transaction database where the User's account information is tracked, a live price feed, and the equation to multiply out the number of coins by the current price.

The "error control" function will also alert the customer if there are insufficient funds through a dropdown window: You need \$dollar amount for this transaction and you have \$dollar amount in your account now."

### **B. Account Enhancements**

#### **1. Cost basis**

We will add a cost basis calculator and a gain/loss tracking column, so the User can quickly see the increase or decrease of the coin's value from a. purchase, b. same day, and c. current year (for tax purposes).

### **C. Registration Enhancements**

The Customer will receive his private and exclusive customer number via email after registering.

The Customer will be able to log-in via the Customer ID Number, assigned automatically upon registration, which is required on each page currently for security reasons.

Going forward, there will be additional password protection and customized User names.

### **D. UI Enhancements**

The current large logo buttons representing each currency will become active links to choices to " buy, sell, see your cost basis, see price history, see current price, see news articles."

#### E. Nav Bar

An Explore option will be added to the Nav Bar that will link to Price History, Current New Articles, Our Customer Blog, and an AP Newswire feed).

When appropriate, there will be an information section at the bottom of the pages in mice-type that will include Contact Us info, Management info, etc.

#### F. Data Base Expansion

The Store will be able to add more customers and more currencies.....someday.