

Project 4 – Portfolio Optimizer

By kepler62f

iShares Core Builder by BLACKROCK



Build a strong core portfolio
Answer 5 quick questions to start.



What do you consider your investment style to be?



Consider risk versus return

Your comfort with risk can depend on your previous investment experience, performance goals, time horizon, and net worth. In general, the higher return you want, the greater risk you'll have to take.



Question 1 of 5

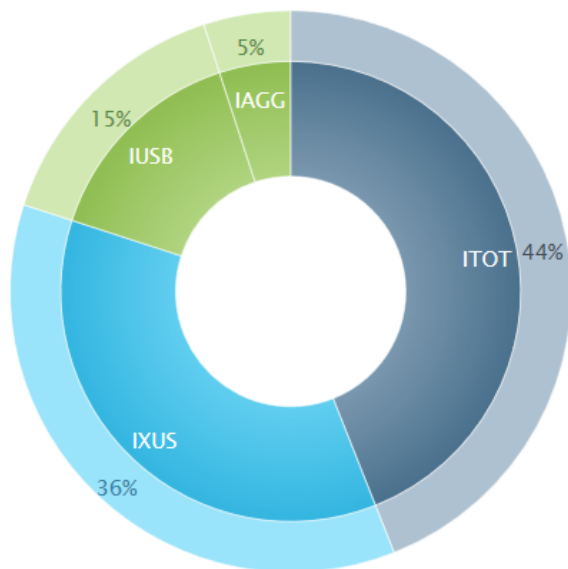
NEXT

+ What are iShares Core Funds?

+ Why iShares?

<https://www.blackrock.com/tools/core-builder/us>

Sample Portfolio Overview



U.S. Stocks

iShares Core S&P Total U.S. Stock Market ETF
Allocation: 44%, Expense Ratio: 0.03%

Change Answers		Change Holdings	
Exposure / Fund Name	Expense Ratio (%)	Value \$	Weight (%)
Broad Market ITOT iShares Core S&P Total U.S. Stock Market ETF	0.03	440.00	44
World IXUS iShares Core MSCI Total International Stock ETF	0.11	360.00	36
Broad Market IUSB iShares Core Total USD Bond Market ETF	0.06	150.00	15
International Bond IAGG iShares Core International Aggregate Bond ETF	0.09	50.00	5
Clear		Show All Funds	
		\$1,000.00	100%
Portfolio Value: \$ 1000		Why does the amount matter?	

Your Unique Portfolio

Create new portfolio

Select asset classes:

Cash	Research	Add	Remove
S&P 500	Research	Add	Remove
MSCI Europe	Research	Add	Remove
MSCI Emerging Market	Research	Add	Remove
ICE U.S. Treasury Core Bond Index	Research	Add	Remove
Gold	Research	Add	Remove

Selection:

Asset Class	Allocation (%)
Cash	<input type="text" value="0"/>
S&P 500	<input type="text" value="0"/>
ICE U.S. Treasury Core Bond Index	<input type="text" value="0"/>

Analyze Portfolio

Research

(Static info of individual asset class will appear here)

Portfolio Analysis

Based on historical monthly returns from July 2012 to Jun 2017 for the selected asset classes, with portfolio target return pegged at 70th percentile of average monthly returns, the following allocation would have produced the best risk adjusted returns on a portfolio basis.

Asset Class	Markowitz Optimized Weights (%)
Cash	40.0000
S&P 500	59.9995
MSCI Europe	0
MSCI Emerging Markets	0
ICE US Core Bond	0.0004
Gold	0

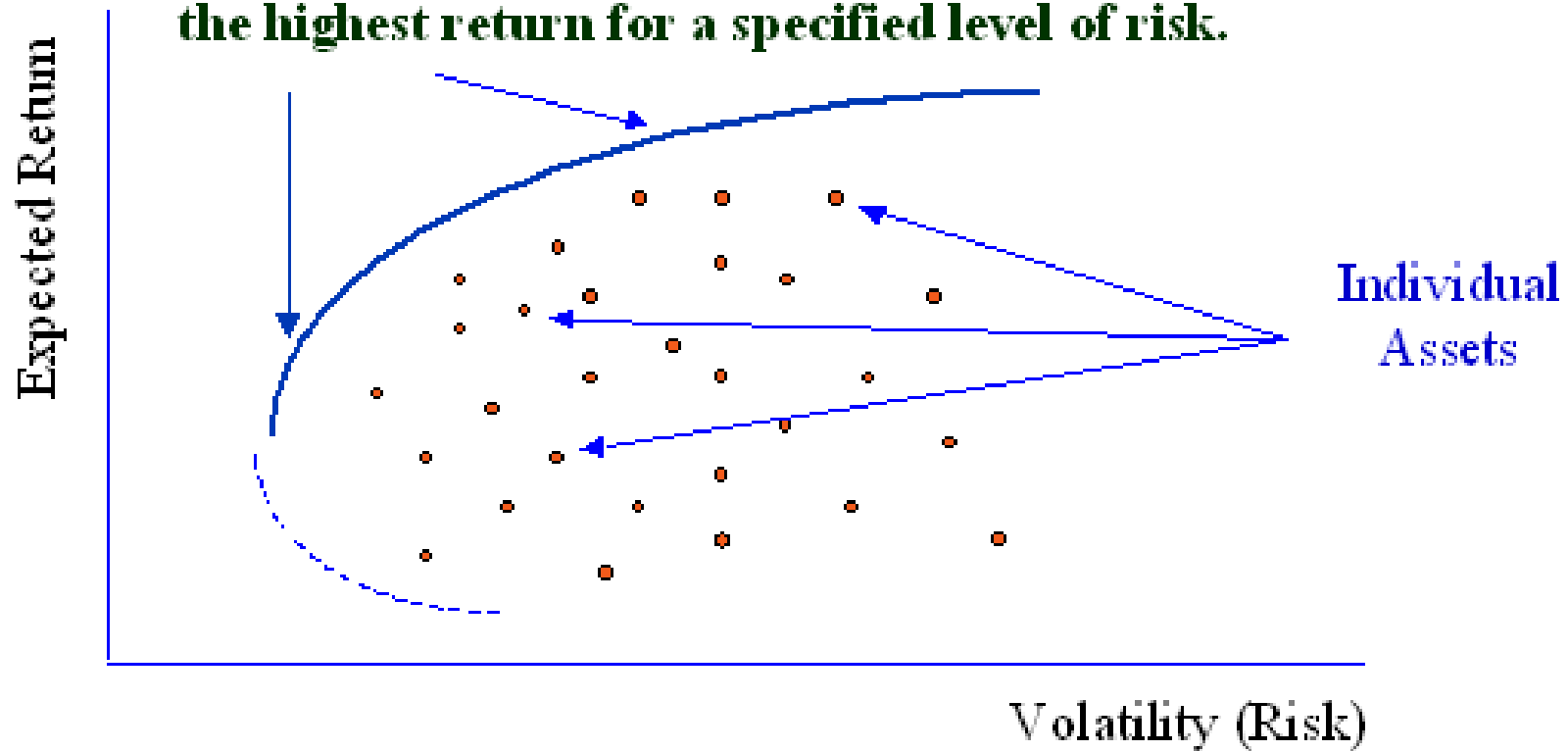
Expected Return: 8.8395 %

Expected Variance: 0.8273 %

<https://kepler62f.github.io/project-4/>

Modern Portfolio Theory...

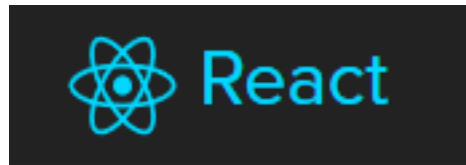
Efficient Frontier : all portfolios that provide the highest return for a specified level of risk.



Modern Portfolio Theory...

$$\begin{array}{ll} \mathcal{M} & \text{minimize} \quad \frac{1}{2} w^T \Sigma w \\ & \text{subject to} \quad m^T w \geq \mu_b, \text{ and } e^T w = 1, \end{array}$$

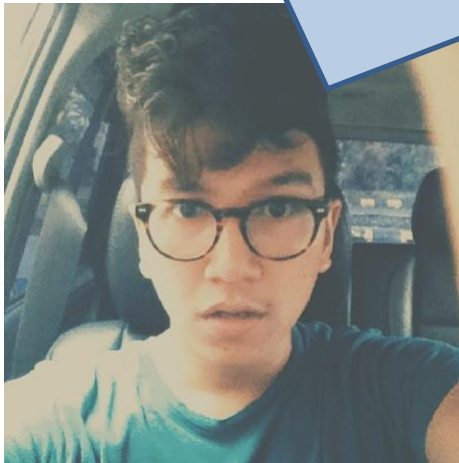
Tech Stack...



PDRP?

WARNING

**Project 4 can be experimental.
But please take into account
the steep learning curve...**



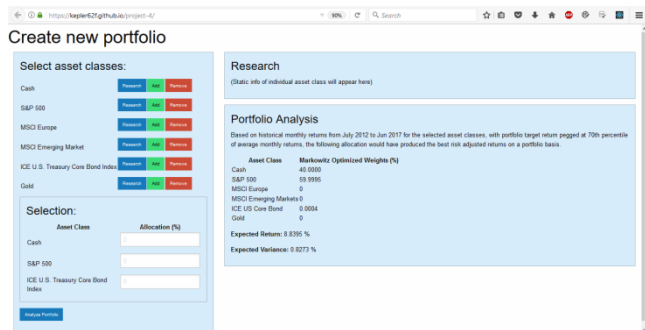
A close-up, high-angle shot of a roulette wheel. The wheel is dark with gold-colored segments and numbers. A white ball is in motion, having just landed in the 25 segment. The wheel is surrounded by a green felt surface.

Django vs Ruby vs Node.js

Django	Ruby	Node.js
Models–Views–Templates	MVC	MVC
python manage.py runserver Makemigrations	rails s, rails g rails db:migrate	yarn start
Built-in ORM	Active Record	Sequelize
pip install	gem	npm, yarn
url(r'^/user/\$', function)	rails routes Get '/user/id	App.get('/user/id', function)

Architecture

React Frontend



<https://kepler62f.github.io/project-4/>

REST API response

REST API request

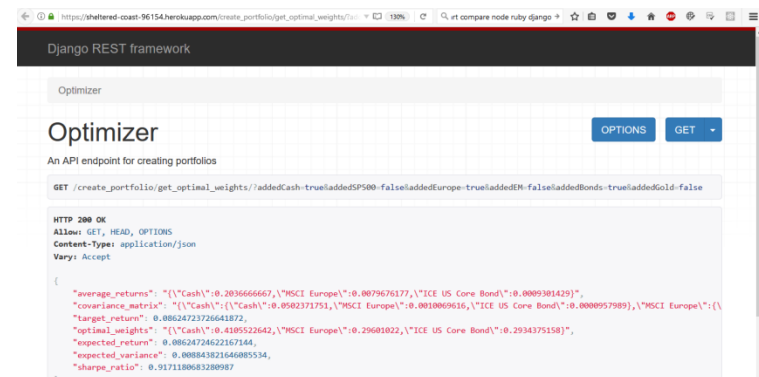
Postgresql Server

Django ORM

Pandas

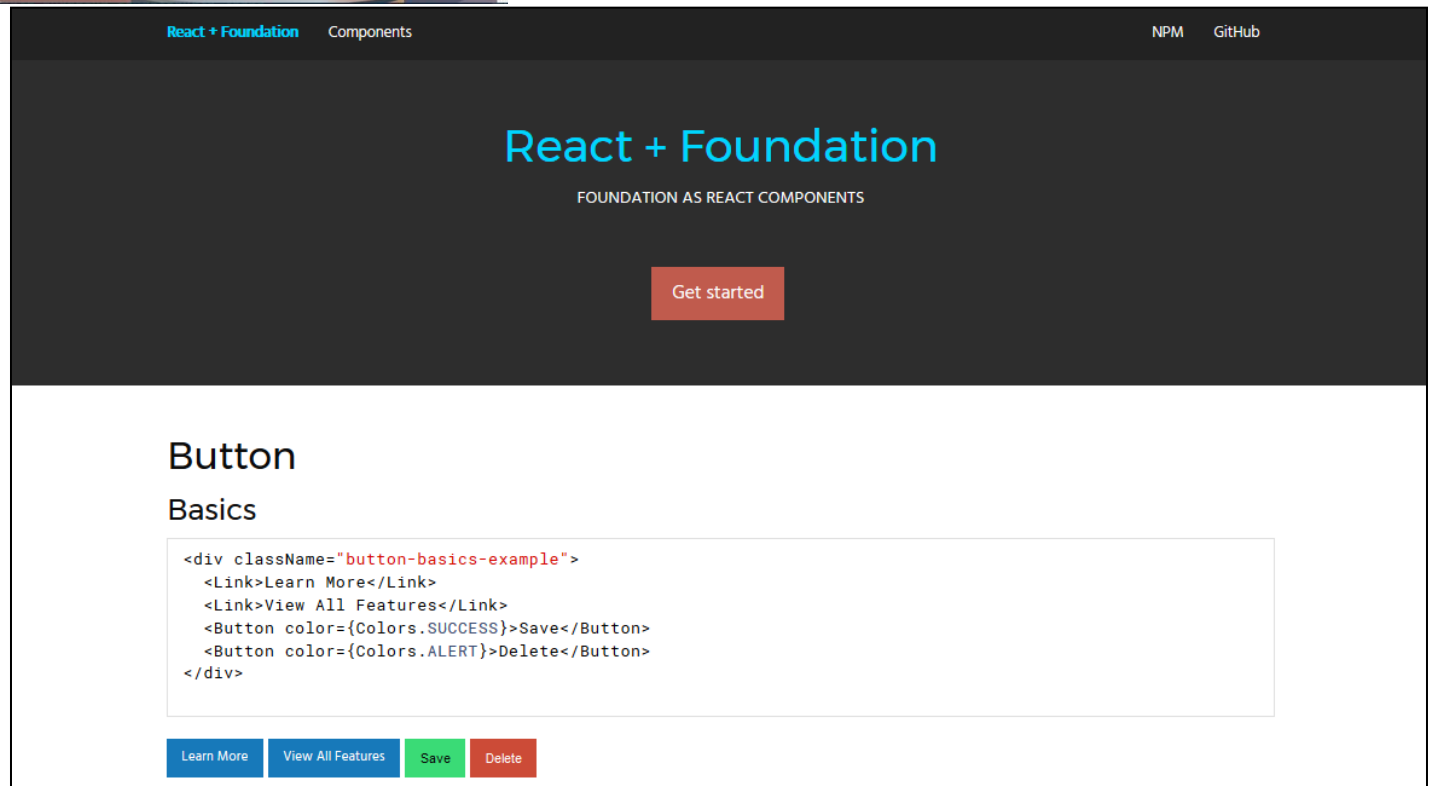
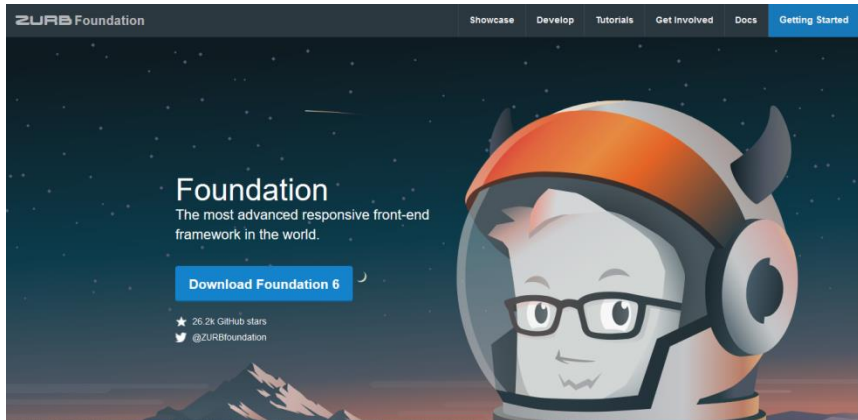
Portfolioopt, numpy

Django Backend (API server)



`/create_portfolio/get_optimal_weights/?addedCash=true&addedSP500=false&addedEM=true&addedBonds=true&addedGold=false`

CSS Framework



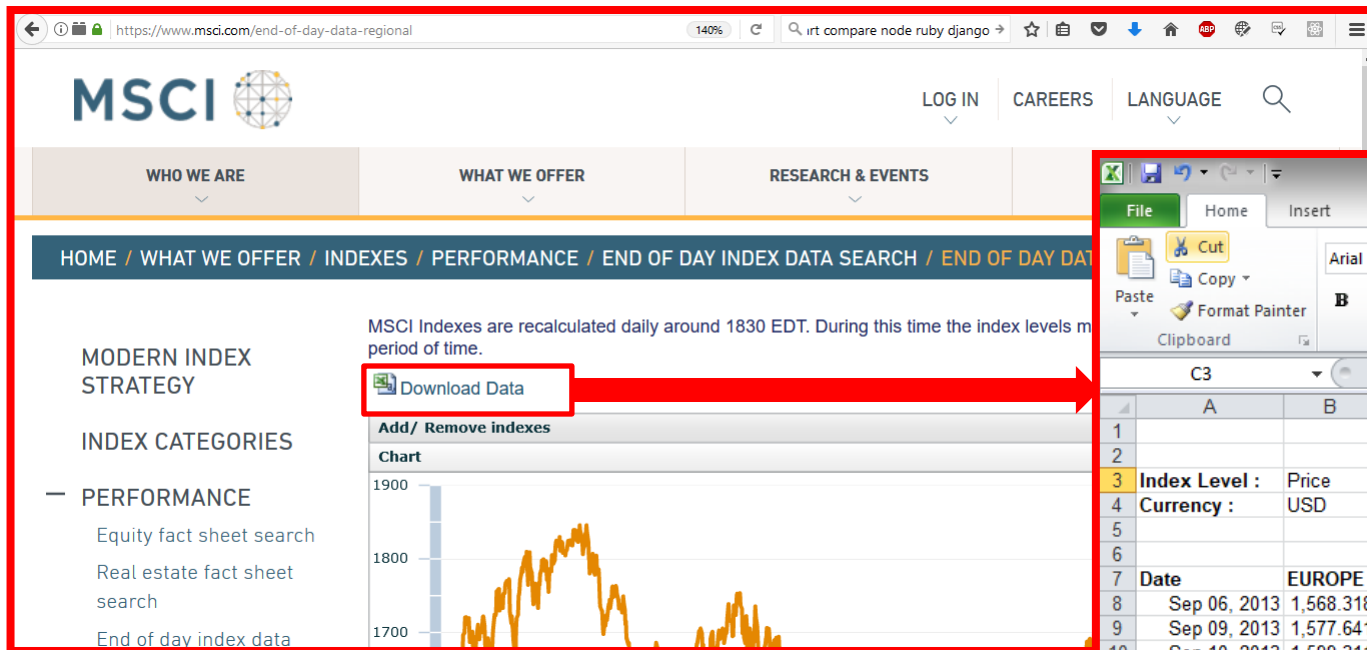
Challenges

- Authentication
 - django.contrib.auth (= Node passport, Ruby device)
 - Django-rest-auth (with json web token)
 - Reactjs + djang-rest-auth

The screenshot shows the GitHub repository page for `ZachLiuGIS / reactjs-auth-django-rest`. The repository is described as "A ReactJS authentication demo site with django-rest-auth backend". It has 16 commits, 1 branch, 1 release, 2 contributors, and is licensed under MIT. The latest commit, `ef29ee3`, was made on April 22, 2016, by ZachLiuGIS, with the message "fixed reset user password.". The repository structure includes a `project` directory, a `react_src` directory, and a `templates/project` directory. The `project` and `react_src` directories were last updated a year ago, while `templates/project` was last updated 2 years ago.

GitHub repository page for `ZachLiuGIS / reactjs-auth-django-rest`. The repository is described as "A ReactJS authentication demo site with django-rest-auth backend". It has 16 commits, 1 branch, 1 release, 2 contributors, and is licensed under MIT. The latest commit, `ef29ee3`, was made on April 22, 2016, by ZachLiuGIS, with the message "fixed reset user password.". The repository structure includes a `project` directory, a `react_src` directory, and a `templates/project` directory. The `project` and `react_src` directories were last updated a year ago, while `templates/project` was last updated 2 years ago.

Challenges – Index Data – Can't scrape?



MSCI

LOG IN CAREERS LANGUAGE

WHO WE ARE WHAT WE OFFER RESEARCH & EVENTS

HOME / WHAT WE OFFER / INDEXES / PERFORMANCE / END OF DAY INDEX DATA SEARCH / END OF DAY DATA

MODERN INDEX STRATEGY

INDEX CATEGORIES

PERFORMANCE

Equity fact sheet search

Real estate fact sheet search

End of day index data

MSCI Indexes are recalculated daily around 1830 EDT. During this time the index levels may fluctuate for a short period of time.

Download Data

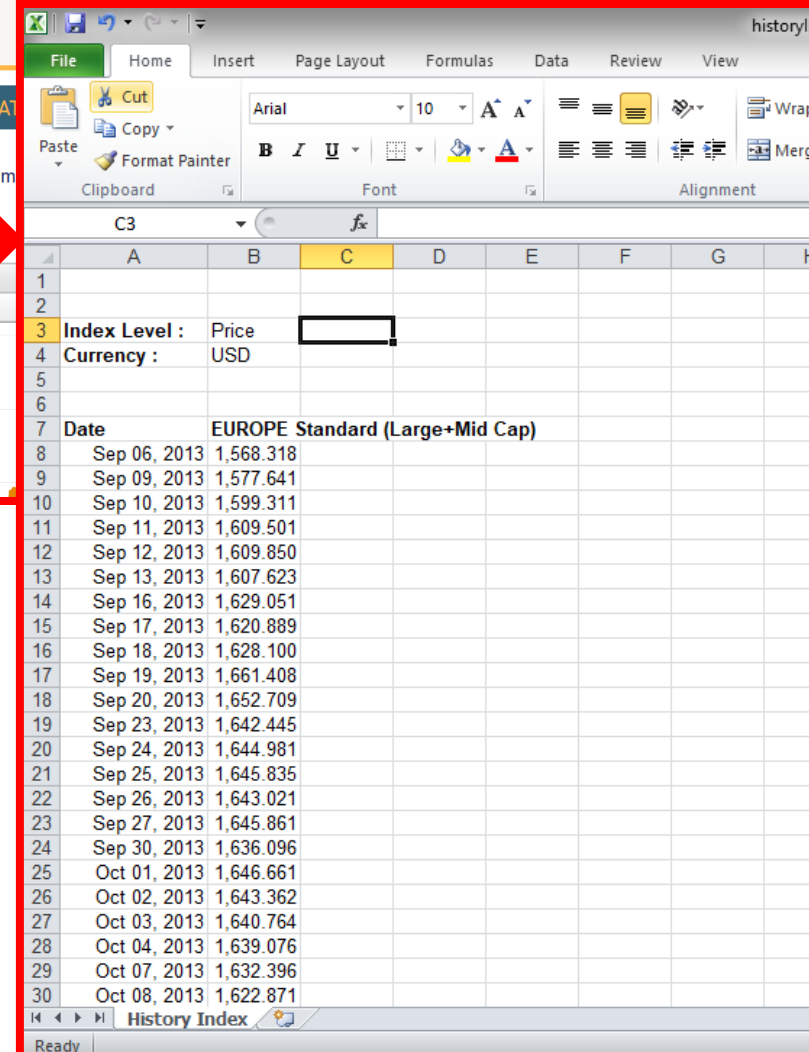
Add/ Remove indexes

Chart

1900

1800

1700



Date	Index Level	Price	Currency
Sep 06, 2013	1,568.318		USD
Sep 09, 2013	1,577.641		USD
Sep 10, 2013	1,599.311		USD
Sep 11, 2013	1,609.501		USD
Sep 12, 2013	1,609.850		USD
Sep 13, 2013	1,607.623		USD
Sep 16, 2013	1,629.051		USD
Sep 17, 2013	1,620.889		USD
Sep 18, 2013	1,628.100		USD
Sep 19, 2013	1,661.408		USD
Sep 20, 2013	1,652.709		USD
Sep 23, 2013	1,642.445		USD
Sep 24, 2013	1,644.981		USD
Sep 25, 2013	1,645.835		USD
Sep 26, 2013	1,643.021		USD
Sep 27, 2013	1,645.861		USD
Sep 30, 2013	1,636.096		USD
Oct 01, 2013	1,646.661		USD
Oct 02, 2013	1,643.362		USD
Oct 03, 2013	1,640.764		USD
Oct 04, 2013	1,639.076		USD
Oct 07, 2013	1,632.396		USD
Oct 08, 2013	1,622.871		USD

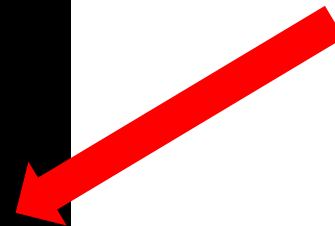
Data Munging.....

- Manual download
- Clean in excel
- Save as .csv ('YYYY-MM-DD' for pandas)
- Create DB table with Pgadmin
- psql \copy
- Resample daily -> monthly (pandas dataframes)

Challenges – setState on nested states

```
16   this.state = {
17     selectedAssets: {
18       addedCash: false,
19       addedSP500: false,
20       addedEurope: false,
21       addedEM: false,
22       addedBonds: false,
23       addedGold: false
24     },
25     proposed_allocation: [0,0,0,0,0,0],
26     portfolioAnalysis: [{
27       average_returns: [
28         {"Cash":0},
29         {"S&P 500":0}
30       ],
31       covariance_matrix: [0,0],
32       target_return: [0,0],
33       optimal_weights: [
34         {"Cash": 0},
35         {"S&P 500": 0},
36         {"MSCI Europe": 0},
37         {"MSCI Emerging Markets": 0},
38         {"ICE US Core Bond": 0},
39         {"Gold": 0}
40       ],
41       expected_return: 0,
42       expected_variance: 0,
43       sharpe ratio: 0
```

Need to wrap in
array else can't
compile render



Can't reach deep state variables...solution: Immutability-helper....

```
Object.keys(weights).map((weight) => {
  console.log(weight)
  console.log(weights[weight])

  if (weight === "Cash") {

    console.log("before, weight is: ", this.state.portfolioAnalysis[0].optimal_weights[0].Cash)

    let newState = update(this.state, {
      portfolioAnalysis: {
        0: {
          optimal_weights: {
            0: {
              "Cash": {
                $set: (weights[weight]*100).toFixed(4)
              }
            }
          }
        }
      }
    })

    this.setState(newState,
      () => console.log('after change this.state', this.state.portfolioAnalysis[0].optimal_weight
    )
  } // if cash
```