FE Correlations Test

In this practical, you will design and implement a user interface for exploring correlations between stock tickers, a common task among financial professionals. This challenge is inspired by a previous project at Kensho. We hope it will give you the opportunity to show off your strengths.

Please spend **no more than 3-4 hours** on this practical. Since this is a short timeframe, focus on showcasing your strengths. We do not expect a perfect, production-ready application.

Instructions

We have provided two API endpoints for you to use in this practical. The first endpoint will provide you with the “universe” of tickers that you will consider for this application. The second endpoint will provide pairwise correlations between these tickers. Please see the API Documentation section for details.

Your prototype solution should, at minimum, allow users to select tickers from the “universe” that they wish to explore, then display the correlation data for these tickers in a useful way. The exact inputs and output are left purposefully vague.

In addition, you might choose to focus on one or more of the following, based on your strengths:

1. Following good code standards for maintainability and readability.
2. Handling any error conditions or edge cases.
3. Styling the page in an aesthetically pleasing way.
4. Refining the user experience of the application (see User Profile section)
5. Anything else you think is important.

Feel free to use third party libraries, boilerplate code, or documentation from any source in order to help speed up the implementation process. **Please clearly specify which code you authored for this practical.**

Deliverables

1. Source code needed to run your challenge solution.
2. Installation instructions needed to set up any additional dependencies/requirements.
3. A paragraph explaining what you focused on, and your approach and design.

User Profile

Your user is a busy financial professional who is just interested in the **most** positively and negatively correlated pairs of assets within a list of assets 90% of the time. The other 10% of the time, he wants to get a big-picture view of the correlations between every pair of assets.

API Documentation

/api/tickers

API endpoint that returns a list of tickers. This list is the “universe” of tickers that you should allow your user to choose from in your application.

Sample endpoint:
http://k-fe-practical.herokuapp.com/api/tickers/

Args: None
Returns: Stringified JSON in the following form.

```
{
    "tickers": [ list of ticker strings ]
}
```

/api/correlation

API endpoint that returns the pairwise correlations for a list of tickers. This endpoint will only accept tickers returned from ‘/api/tickers.’

Sample endpoint:
http://k-fe-practical.herokuapp.com/api/correlation/?tickers=AAPL&tickers=MSFT

Args:
Tickers: a list of ticker strings
Returns: Stringified JSON similar to the following example:

```
{
    "tickers": ["AAPL", "GOOG", "MSFT"],
    "correlations": [{
        "pair": ["AAPL", "GOOG"],
        "value": 0.5234
    } for each pair of tickers]
}
```