Data Engineering Challenge ReadMe

Introduction:

First of all, I would like to thank you for considering me for this task. Also, thanks to Stephen Kazibwe for spending time to grade this assessment challenge. I really enjoyed every part of this task! 😊

Used Technologies:

- Anaconda Package (which includes Jupyter Notebook)
- Jupyter Notebook as IDE
- Postgres for Windows
- psycopg2 connector for Postgres
- Flask to render python elements to HTML

Approach:

After installation of Anaconda, Postgres and Psycopg2:

1. I created a free account on Coin.io and got the free API authentication key.
2. I used this key to retrieve data from 1 Jan 2016 till now However I got only 4 months.
3. I initialized connection to DB from Python then I created the two tables as recommended and then I inserted the value of the first table (curren) manually.
4. Then I inserted the values for historical data along with insertion timestamp with a loop. This loop simply loops over the responses dictionary and builds SQL Statement and then execute it into postgres.
5. Then I selected, from the historical table, aggregated average trade count by month and year.
6. I used Month_Year as label and average count as the value since you did not specify which values should I show in the chart.
7. I used flask to render my values in to the HTML template which is located at /templates
8. Also I've downloaded Charts.min.js and move it to /sources as it should be done according to flask documentation

Final output:

BTC Bitcoin Monthly Average Number of Trades
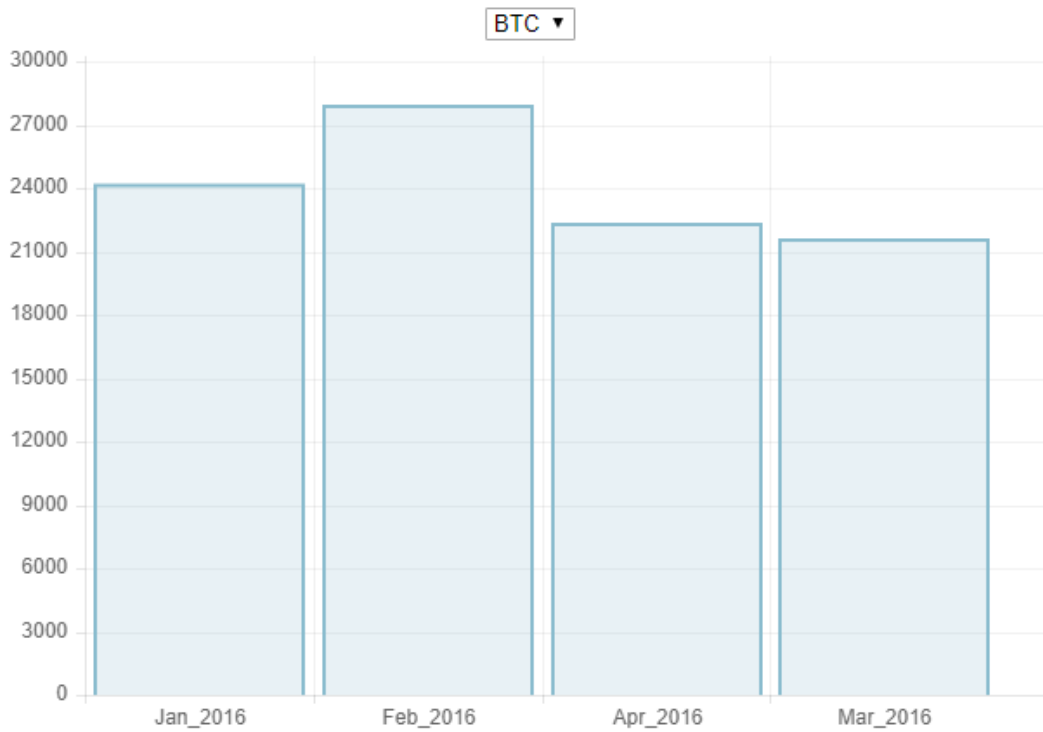
Note:

The maximum chart value is calculated based on the maximum output of the select query.

Note:

I've implemented google analytics but it's now showing bitcoin most visited coin despite that I've added the right tags to the dropdown list as shown below:

```
<select name="bitc" onChange="ga('send', 'event', 'Form Event', 'Select Changed', 'BitCoin',this.options[this.selectedIndex].text);
    <option value="{{bitcoins[0]}}" selected>{{bitcoins[0]}}</option>
    {% for bitc in bitcoins[1:] %}
```

I found another way by adding custom JavaScript code but I did not want to copy a ready-made code.
However, the it's showing website total visit count as below:
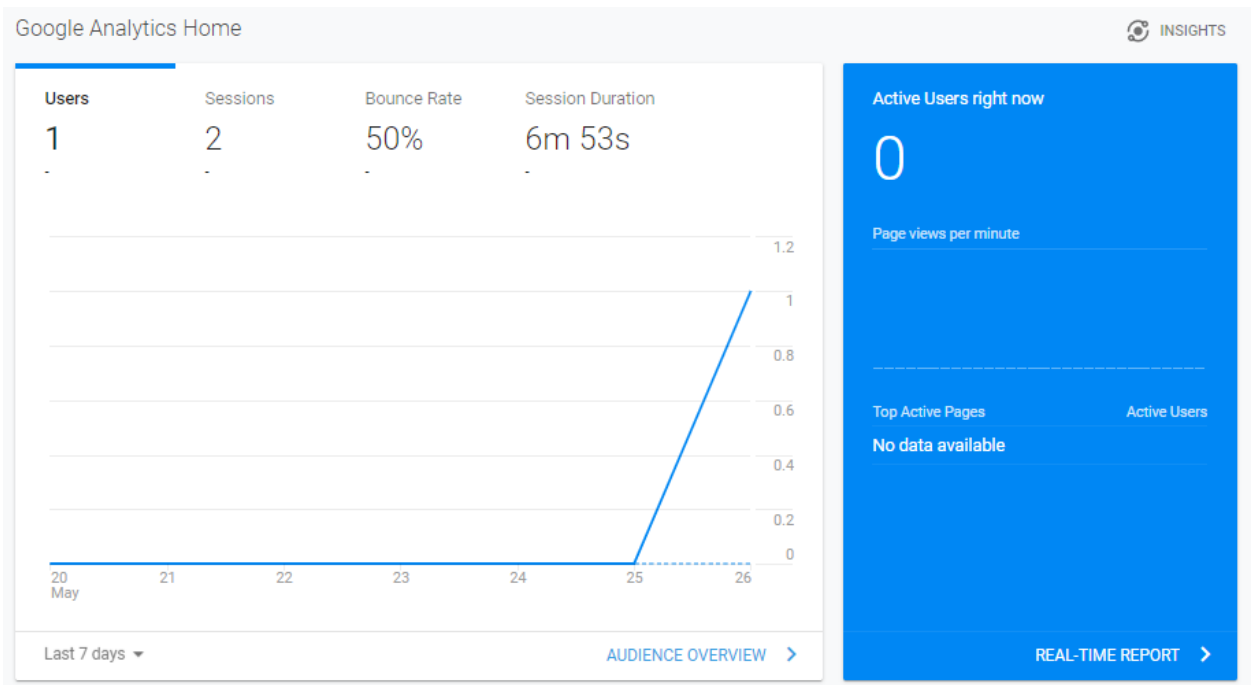
How to run this code:

There are 2 ways to run this code after installing postgres and include all the imported packages:

1. Install anaconda which includes Jupyter and then run **.ipynb** file
2. Run **.py** file directly

<center>Thanks for your time</center>