3.1 Introduction

The purpose of Sprint #2 is for you to write Python functions which will load the raw data into both a Pandas Data Frame as well as into SQL. You will also be required to write a number of *unit tests* which verify that the SQL and Pandas data share some properties.

The code needs to be written in Python 3.6+ and use standard libraries.[1]

As an example of what is being looked for, the code `BART.py` was provided to you. This code is an extensive example and includes a number of features which are *not* required to receive full credit. While reading over this assignment, I strongly recommend taking time to identify where in the `BART.py` code

3.2 Requirements

You and your group will write two Python functions. The first one will load the data into a pandas Data Frame and will be called `load_pandas` and can take a single argument of the *base directory* of where to find the required files. The other function, `load_sql` will do the same thing, but will load the data into the database itself.

Both functions should take a *string* directory which is the directory that the function will work in. All temporary files, downloaded data, etc. should be stored in this base directory. DO NOT USE any other directory than this or a subdirectory of this base directory. Any subdirectory used should be created by the function itself.

The function should be able to run multiple times without issue. In other words, it needs to clean up after itself as well as make sure that files which need to be empty / not-exist do so.

In order to receive full credit on this assignment, you will need to upload a text file which contains these two functions. The file should look something like the template below.

```
import ...

def load_pandas(base_dir):
    ...

def load_sql(base_dir):
    ...
```

You should have multiple functions within your uploaded Python file and those functions can have any name you wish. The two functions, `load_sql` and `load_pandas` however, are required.

Your code should be easy to read and generally follow good coding practices. There should be very little duplication (e.g. lines which are repeated in the code), variables should be named properly and the functions should be clearly defined. Global variables should be avoided.

[1] In particular if you can install the library using either `pip| or conda| then it is acceptable.`

IMPORTANT: this need to be a text file, NOT a jupyter notebook.

The Pandas Function

The function `load_pandas` should load the data and then return a Data Frame containing the loaded data. If your dataset is multiple tables, then it should return a dictionary containing your tables, where the names of the Data Frames are the keys to the dictionary.

As stated above, this is allowed to call other functions within your file and allowed to use any standard Python library. Since you will be writing code to both return a Pandas Data Frame and load the data into SQL your code should be functional along this dimension. Specifically your code should not repeat the same logic to clean the data – there should be a function which implements this logic that can be called by both the Pandas function and the SQL function.

The SQL Function

This function `load_sql` should load the data into the SQL database that we have been using in class. In order to do this you will need to implement a few different pieces of logic:

1. If the table exists, then DROP it.
2. CREATE a table, with the proper data types.
3. COPY the data in from a properly formatted file.

In class we went over how to do these three operations so feel free to refer to the videos. Code was also provided which implemented these operations.

Each group will be given a separate schema in which to operate. You are allowed to use any tablename within that schema, but no one else's schema. You should also not use the schema `public`, `cls` or `stocks`.

BE CAREFUL WHEN USING THESE OPERATIONS. DO NOT SIMPLY COPY / PASTE CODE FROM STACKOVERFLOW OR FROM WHAT WAS PROVIDED. IF YOU RUN CODE WHICH DELETES SOMEONE ELSE'S WORK YOUR GRADE WILL BE PENALIZED.

3.3 Data

Please note, in the comments of your code, any data that needs to be downloaded and placed in the base directory. For example, if you need to download a file from a website (or multiple files), please describe how those files can be downloaded.

Importantly, the input to these files should be "raw" data from the internet. To receive full credit, the data needs to be directly downloaded from the internet. If you need to modify the underlying data files in order to have your code operate, this will negatively impact your

grade. Note that the code itself doesn't need to complete the downloading, you can write a comment in the code describing where you downloaded the data from.

If you received your file from a non-public source, please describe to me how I can receive that data (or send it to me via Canvas/DropBox/email).

3.4 Unit Tests

You must also write two unit tests to verify that the data in SQL matches the data in Pandas. To complete the assignment, you are required to write at least two tests, described below. You are free (and encouraged) to think about other tests that you could run to verify that your data was copied properly.

Test #1: Verify the number of rows

The first test you need to run should count the number of rows in each of your SQL tables and verify that there are the same number of rows in the Data Frame. In particular, you need to make sure that the data loaded in by your `load_sql` command has the same number of rows as the data loaded by your `load_pandas` command.

In order to complete the test, please use a command similar to `assert` which returns an error or `raise` an `Exception` to alert the user that the loading task did not complete successfully.

Test #2: Value count verification

The second test you need to run should pick the most important column (to your questions) in your most important table and verify that the distribution of values is the same in your SQL table and your Pandas Data Frame

3.5 Grading

In order to receive a "B" on this assignment, your code needs to do all of the above. To receive an "A" on the assignment, all of the previous requirements need to be met, but also one of the following extensions needs to be completed:

1. **Data From Sources:** In this extension you need to write code that will download your data from its original source. If you scrapped data from the internet, this should include that scrapping. If you simply downloaded a file, then your code should complete that download. This should be a separate function that your `load_sql` and `load_pandas` command should call.

2. **Test all column value counts:** In this extension you need to write code which will automatically test *all* of your tables and columns and verify that the distribution of values is the same. To complete this extension you need to write a separate function which will undertake this operation.

3. **Create a Python library:** In this extension you should create a Python library which allows you to import the data. Specifically, you should be able to run the following code:

```
from urlibrary import data_handler

df = data_handler.load_data()
```

This code should check to see if the data is available in a local file and if the data exists in either it will load it from there. If the data does not already exist it should then run the `load_pandas` command and return the data that way.