Coding challenge solved by

Guilherme Martins

The document aim is to show the necessary requirements and dependencies to run the script as well as explain how to run.

Overall Process:

1. Extracting data:

Collect the data from Mongo DB and then save the json file at the folder "1. Dados Recebidos".

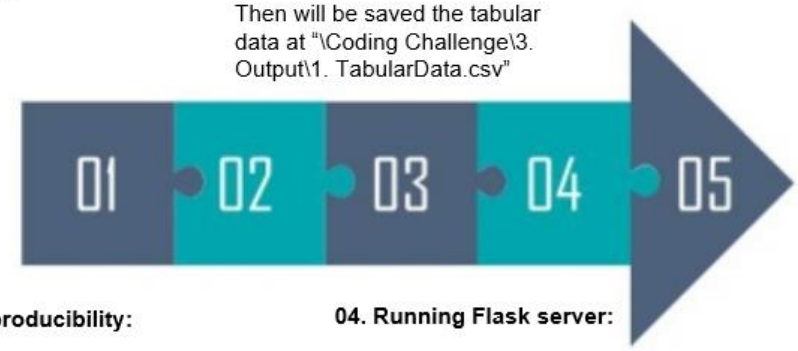
03. Writting tabular data:

Run the script at "\Coding Challenge\2. Scripts\1. TabularData.py".

Then will be saved the tabular data at "\Coding Challenge\3. Output\1. TabularData.csv"

05. Chaining tabular data into LivUP ETL :

Run the script at "\Coding Challenge\2. Scripts\3. Read_html.py" in order to get tabular data from API as **GET METHOD**.



02 Reproducibility:

At the Coding Challenge folder and using CMD prompt ...

Create virtual enviroment using "python –m venv livup_virtual_env".

After activate the enviroment using "livup_virtual_env\Scripts\activate"

And install all necessary libraries with "pip install –r "\2. Scripts\requirements.txt"

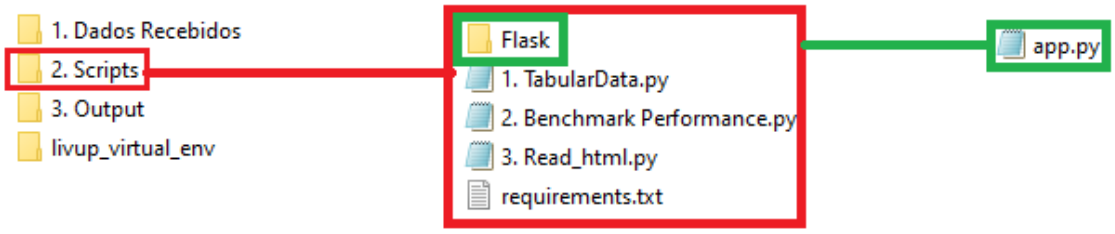
04. Running Flask server:

Run the script at "\Coding Challenge\2. Scripts\Flask\app.py"

We can access the API using
http://127.0.0.1:5000/api/v1/sales
Or filtering direct in the URL
http://127.0.0.1:5000/api/v1/sales/pick_up

Folder Structure:

In the root folder (Coding Challenge folder) there are 4 subfolders as shown in the picture below.



Note: There is script called “2. Benchmark Performance.py” which was used to evaluate how well the programming was running for 1000 sales json file.

Specs used in the project:

- Windows 10
- Python 3.7.1 64-bit
- Visual Studio Code

Libraries used in the project:

```
Click==7.0
Flask==1.1.1
itsdangerous==1.1.0
Jinja2==2.11.1
MarkupSafe==1.1.1
numpy==1.18.1
pandas==1.0.1
pycodestyle==2.5.0
python-dateutil==2.8.1
pytz==2019.3
schedule==0.6.0
six==1.14.0
Werkzeug==1.0.0
```

Note: There is requirements.txt file at “\Coding Challenge\2. Scripts”.