

Loungebuddy Final Project

Github Repository Link: <https://github.com/phi-h-nguyen/data-focused-python>

Group Members

- Phi Nguyen (hpnguyen)
- Harshil Vora (hvora)
- Carol Shen (xiaoxin2)
- Yasuhiro Kawabata (ykawabat)

Usage

Run `main.py` to use this application. See installation instructions using pipenv below.

Modules

`lounges.py`

Webscrapes www.loungebuddy.com to return a list of all lounges seperated by terminal based on given airport codes.

`aviationstack.py`

Uses the API provided by www.aviationstack.com/ to get real time flight data. Cleans data with pandas / numpy and does visualizations with matplotlib. Note that this API is limited to 100 requests / month, so if you run into any issues querying data, please sign up for a new API key and replace the one used. There are 2 sample datasets included in this repository: PIT -> DFW on 2/24 and PIT -> MDW on 3/1 that replace the need to query new data. Anytime new data is queried, json files are created so that the data can be saved and used in the future.

`weatherAPI.py`

Uses the OpenWeatherMap API to query weather data.

main.py

The file to run. This runs a command-line based menu that allows you to explore flight information such as weather, flight delays, and lounges.

Installation

Uses [pipenv](#) to keep track of requirements and environment. To install, run `pip install --user pipenv`. After installing pipenv, install all dependencies by running `pipenv install`

Virtual Environment

Run `pipenv shell` in your command line to activate this project's virtual environment. If you have more than one version of Python installed on your machine, you can use pipenv's `--python` option to specify which version of Python should be used to create the virtual environment. If you want to learn more about virtual environments, read [this article](#).

Downloading Packages

The repository contains `Pipfile` that declares which packages are necessary for the project. To install packages declared by the `Pipfile`, run `pipenv install` in the command line from the `/app` directory.

To add additional packages, run `pipenv install [PACKAGE_NAME]`, as you would install python packages using `pip`. This will update `Pipfile` and add the downloaded package under `[packages]`. Note that `Pipfile.lock` will also be updated with the specific versions of the dependencies that were installed. Any changes to `Pipfile.lock` should also be committed to your Git repository to ensure that everyone on the team is using the same dependency versions.