

DOPPELGÄNGER

SYNTHETIC DATA GENERATOR

Application Installation Guide










1. Localize GITHUB Directory URL -

<https://github.com/DoppelgangerSyntheticData/Doppelganger>

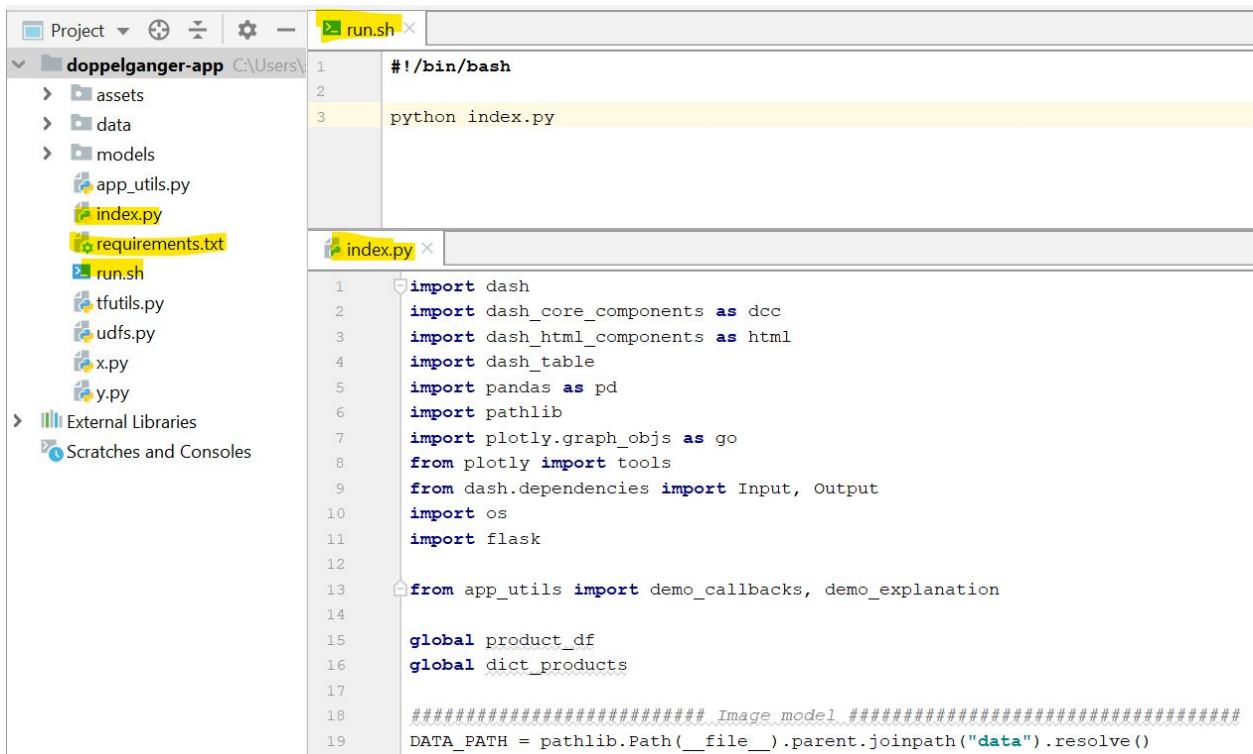
The screenshot shows the GitHub repository page for **DoppelgangerSyntheticData / Doppelganger**. The repository is marked as **Private**. The navigation bar includes links for **Code**, **Issues**, **Pull requests**, **Actions**, **Projects**, **Security**, **Insights**, and **Settings**. Below the navigation bar, it shows **main** branch selected, **3 branches**, and **0 tags**. There are buttons for **Go to file** and **Add file**. The commit history table shows the following entries:

Commit Message	Commit Hash	Time
prashgithub updated user guide	6ebbb30	13 minutes ago
Documentation	updated user guide	
doppelganger-app	updated user guide	
doppelganger-ipynb	added IPYNB	
initial_brainstorming	moved from prashgithub	
GROUP 19 PROJECT.docx	moved from prashgithub	
GROUP 19 PROJECT.docx	moved from prashgithub	
LICENSE	Initial commit	
README.md	Update README.md	

2. Below is the folder structure of **doppelganger-app**

 .idea	updated
 __pycache__	updated
 assets	updated
 data	updated
 models	updated
 app_utils.py	updated
 index.py	updated
 requirements.txt	updated
 run.sh	updated

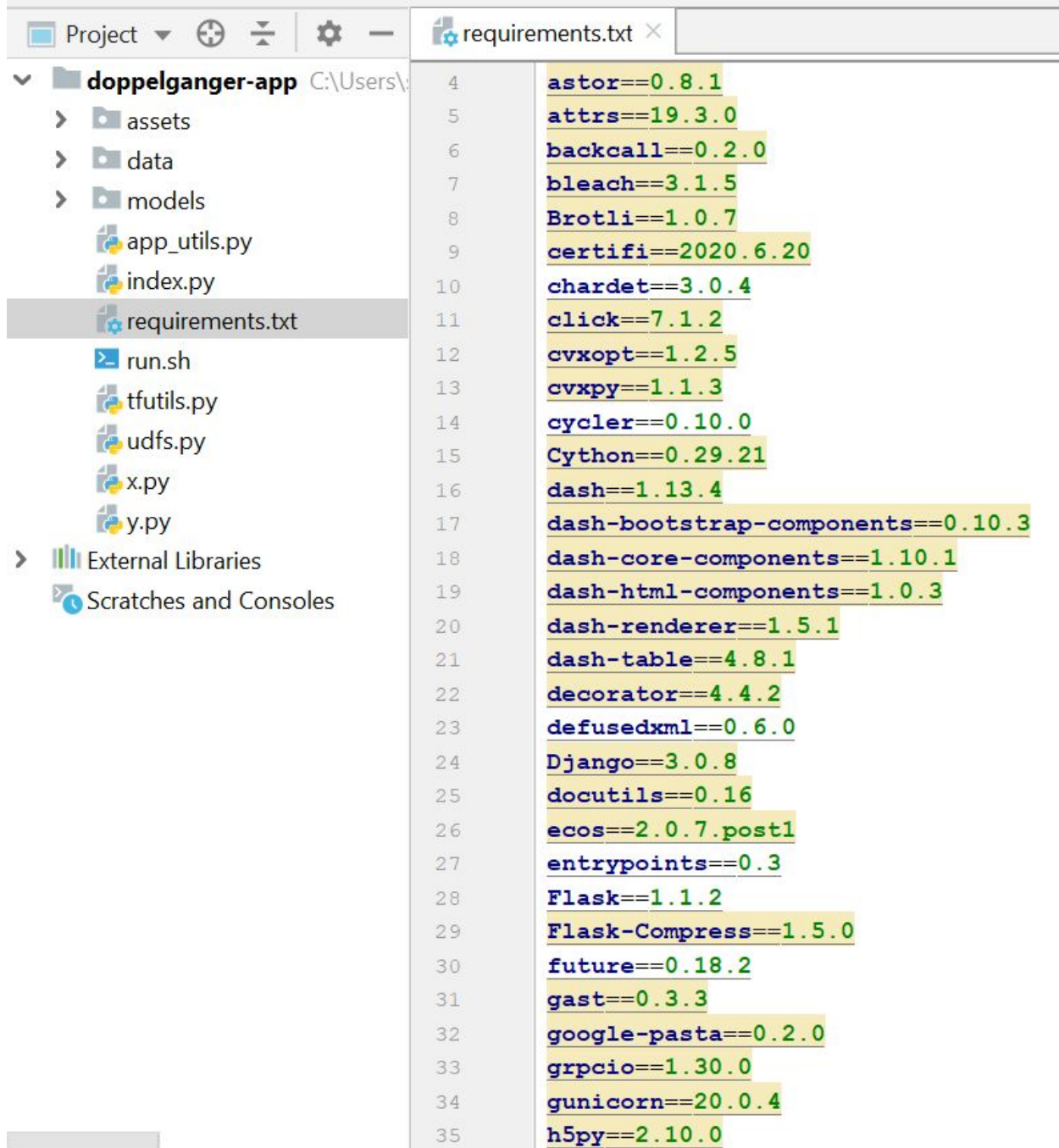
Under Project setup -



```
run.sh
1  #!/bin/bash
2
3  python index.py

index.py
1  import dash
2  import dash_core_components as dcc
3  import dash_html_components as html
4  import dash_table
5  import pandas as pd
6  import pathlib
7  import plotly.graph_objs as go
8  from plotly import tools
9  from dash.dependencies import Input, Output
10 import os
11 import flask
12
13 from app_utils import demo_callbacks, demo_explanation
14
15 global product_df
16 global dict_products
17
18 ##### Image_model #####
19 DATA_PATH = pathlib.Path(__file__).parent.joinpath("data").resolve()
```

- a. Main command file is - **run.sh**
 - To start the python shell execute this command (given python env is already setup)
- b. The application starts from - **index.py**
 - The main starting point of the Dash/Flask Python application
 - It starts the server and loads all backend data files and functions
- c. Required dependencies for Python stays in - **requirements.txt**



```
4  astor==0.8.1
5  attrs==19.3.0
6  backcall==0.2.0
7  bleach==3.1.5
8  Brotli==1.0.7
9  certifi==2020.6.20
10 chardet==3.0.4
11 click==7.1.2
12 cvxopt==1.2.5
13 cvxpy==1.1.3
14 cycler==0.10.0
15 Cython==0.29.21
16 dash==1.13.4
17 dash-bootstrap-components==0.10.3
18 dash-core-components==1.10.1
19 dash-html-components==1.0.3
20 dash-renderer==1.5.1
21 dash-table==4.8.1
22 decorator==4.4.2
23 defusedxml==0.6.0
24 Django==3.0.8
25 docutils==0.16
26 ecos==2.0.7.post1
27 entrypoints==0.3
28 Flask==1.1.2
29 Flask-Compress==1.5.0
30 future==0.18.2
31 gast==0.3.3
32 google-pasta==0.2.0
33 grpcio==1.30.0
34 gunicorn==20.0.4
35 h5py==2.10.0
```

Any missing dependencies can be resolved by using standard Python/pip installation commands.

Run time -

Go to App folder and execute "run.sh"-

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.18363.1139]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\simpl>cd C:\Users\simpl\Documents\GitHub\Doppelganger\doppelganger-app

C:\Users\simpl\Documents\GitHub\Doppelganger\doppelganger-app>run.sh

C:\Users\simpl\Documents\GitHub\Doppelganger\doppelganger-app>
```

Python Dash Server comes up with URL to hit -

```
Running on http://127.0.0.1:8050/
Debugger PIN: 466-190-964
* Serving Flask app "index" (lazy loading)
* Environment: production
  WARNING: This is a development server. Do
  Use a production WSGI server instead.
* Debug mode: on
Running on http://127.0.0.1:8050/
Debugger PIN: 909-291-490
```

-- End --