

1. Created directory for your file.

```
PS D:\crud_dash_JC>
```

2. I setup a virtual environment with pipenv (if not yet installed go here check this out, <https://pipenv.pypa.io/en/latest/installation/>). I already installed mine as shown here:

```
PS D:\crud_dash_JC> pip install pipenv
Requirement already satisfied: pipenv in c:\users\keanu\appdata\local\programs\python\python311\lib\site-packages (2023.9.8)
Requirement already satisfied: certifi in c:\users\keanu\appdata\local\programs\python\python311\lib\site-packages (from pipenv) (2023.5.7)
Requirement already satisfied: setuptools>=67 in c:\users\keanu\appdata\local\programs\python\python311\lib\site-packages (from pipenv) (68.2.0)
Requirement already satisfied: virtualenv>=20.24.2 in c:\users\keanu\appdata\local\programs\python\python311\lib\site-packages (from pipenv) (20.24.4)
Requirement already satisfied: distlib<1,>=0.3.7 in c:\users\keanu\appdata\local\programs\python\python311\lib\site-packages (from virtualenv>=20.24.2->pipenv) (0.3.7)
Requirement already satisfied: filelock<4,>=3.12.2 in c:\users\keanu\appdata\local\programs\python\python311\lib\site-packages (from virtualenv>=20.24.2->pipenv) (3.12.3)
Requirement already satisfied: platformdirs<4,>=3.9.1 in c:\users\keanu\appdata\local\programs\python\python311\lib\site-packages (from virtualenv>=20.24.2->pipenv) (3.10.0)
PS D:\crud_dash_JC>
```

3. I installed the packages (based on youtube vid you sent) using pipenv (it might take a while).

```
PS D:\crud_dash_JC> pipenv install dash
Creating a virtualenv for this project...
Pipfile: D:\crud_dash_JC\Pipfile
Using default python from C:\Users\Keanu\AppData\Local\Programs\Python\Python311\python.exe (3.11.5) to create virtualenv...
[ ==] Creating virtual environment...created virtual environment CPython3.11.5.final.0-64 in 15353ms
creator CPython3Windows(dest=C:\Users\Keanu\AppData\Local\Programs\Python\Python311\python.exe, clear=False, no_vcs_ignore=False, global=False)
seeder FromAppData(download=False, pip-bundle, setuptools-bundle, wheel-bundle, via=copy, app_data_dir=C:\Users\Keanu\AppData\Local\pip\cache)
added seed packages: pip==23.2.1, setuptools==68.1.2, wheel==0.41.2
activators BashActivator,BatchActivator,FishActivator,NushellActivator,PowerShellActivator,PythonActivator

Successfully created virtual environment!
Virtualenv location: C:\Users\Keanu\AppData\Local\Programs\Python\Python311\python.exe
Creating a Pipfile for this project...
Installing dash...
Resolving dash...
Added dash to Pipfile's [packages] ...
Installation Succeeded
Pipfile.lock not found, creating...
Locking [packages] dependencies...
Building requirements...
Resolving dependencies...
Success!
Locking [dev-packages] dependencies...
Updated Pipfile.lock (2928743c4a048152c5a028e89e433cf67d729eac6c8292e88f82a5463bc1684e)!
Installing dependencies from Pipfile.lock (c1684e)...
To activate this project's virtualenv, run pipenv shell.
Alternatively, run a command inside the virtualenv with pipenv run.
PS D:\crud_dash_JC>
```

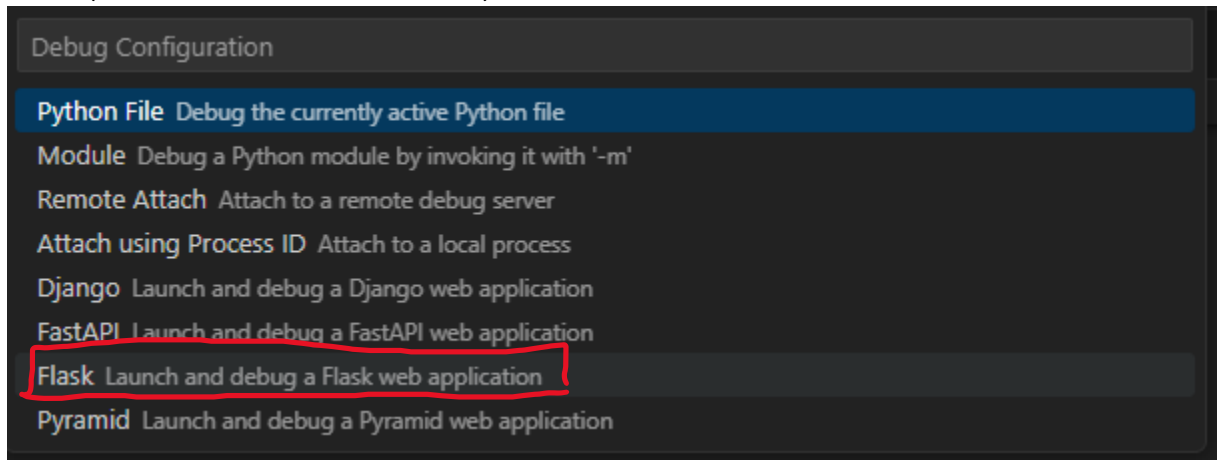
Take note of the virtual env location (especially the “f76HiDQj” for this instance)

A Pipfile & pipfile.lock will be created once you installed atleast 1 package. You can visit the pipfile to check the packages you installed on your virtual environment. For your reference, here are the packages I installed.

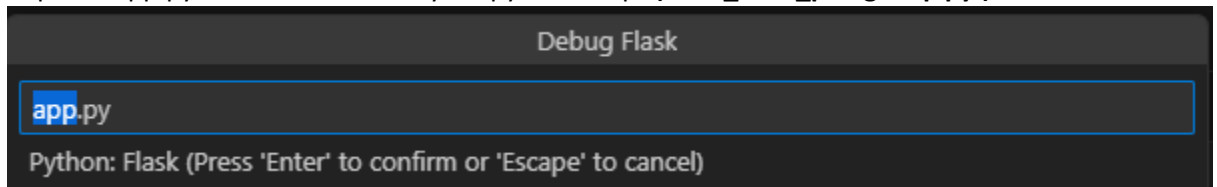
```
[packages]
dash = "*"
pandas = "*"
flask-sqlalchemy = "*"
psycopg2 = "*"
plotly = "*"
numpy = "*"
unicorn = "*"
```

Also, if someone sent you a pipfile, it will automatically install all the packages after installing atleast one with “pipenv install”.

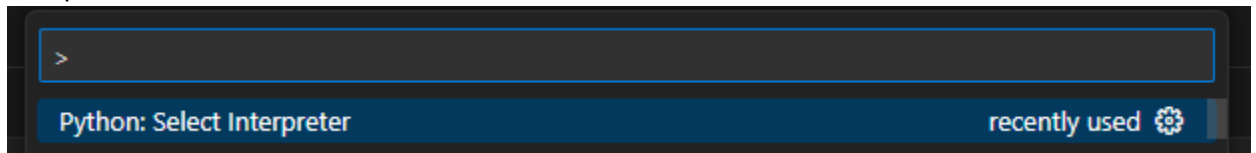
4. **Debug file.** Go to Run and Debug (Ctrl + Shift + D) and click 'create a launch.json file'. Choose 'Flask' (as Dash is related to Flask server).



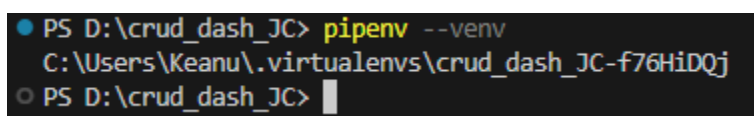
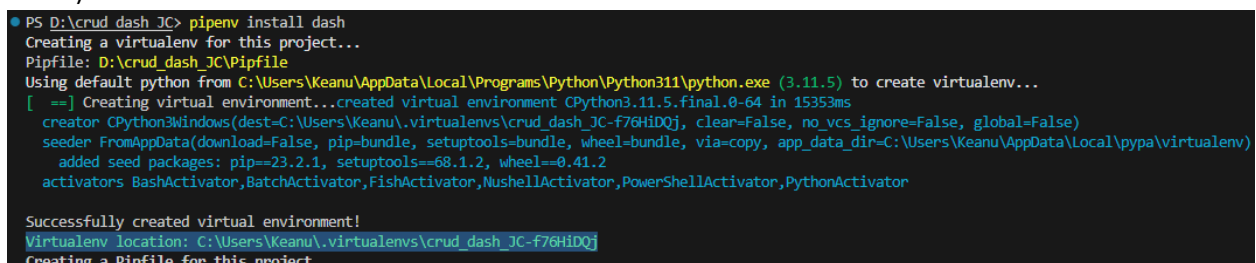
Replace 'app.py' to the filename of your python script ('crud\_dash\_postgresql.py')

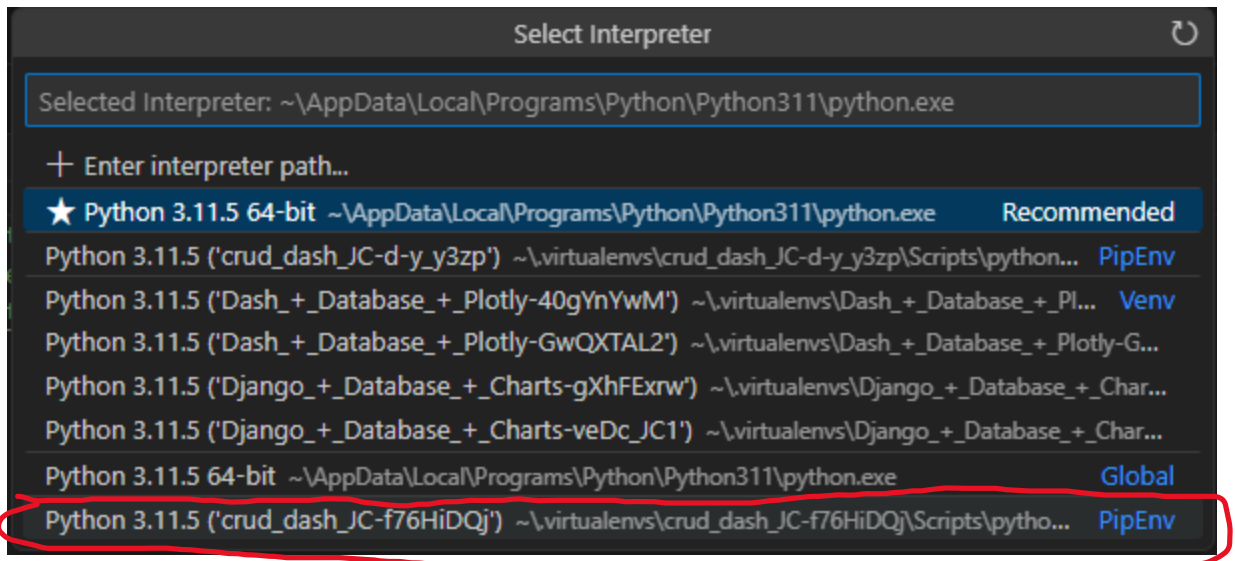


5. **Select interpreter.** Open the command palette (Ctrl + Shift + P) and choose 'Python: Select Interpreter'



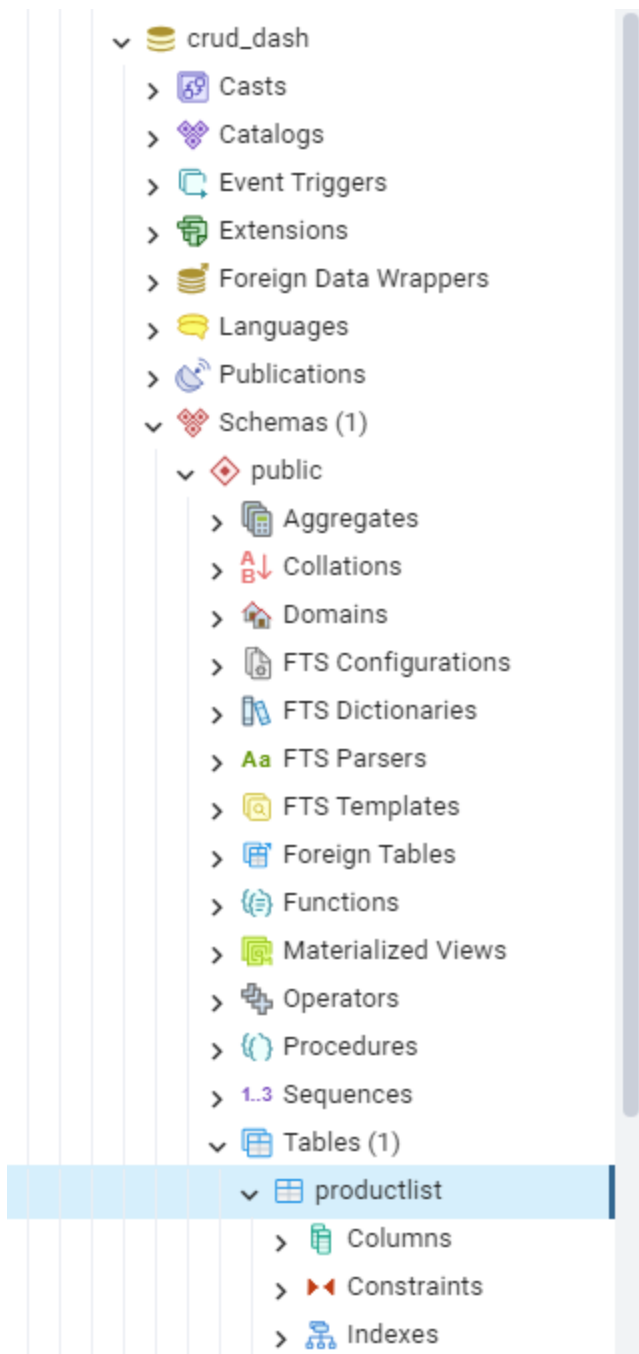
Choose the interpreter with your virtual environment. (You can see it earlier when you setup the virtual environment, such as this screenshot or you may enter 'pipenv --venv' on the terminal to check)





6. Edit the psql. Attached in this image reflects on the database I setup. "PGPassword001" is my psql password; "crud\_dash" is the Database I created; and "productlist" is the name of the table I created inside psql (see screenshot below).

```
crud_dash_postgresql.py x launch.json
crud_dash_postgresql.py > ...
12
13
14 # app requires "pip install psycopg2" as well
15
16 server = Flask(__name__)
17 app = Dash(__name__, server=server, suppress_callback_exceptions=True)
18 app.server.config['SQLALCHEMY_TRACK_MODIFICATIONS'] = False
19
20 # for your home PostgreSQL test table
21 app.server.config["SQLALCHEMY_DATABASE_URI"] = "postgresql://postgres:PGPassword001@localhost/crud_dash"
22
23 # for your live Heroku PostgreSQL database
24 # app.server.config["SQLALCHEMY_DATABASE_URI"] = "postgres://kcfwfqwnavpjg:9473936daf43bff3d17c1dd8ab2c28144dfbf677\
25 # 14cb30622e3017bbe55cdeac@ec2-34-197-188-147.compute-1.amazonaws.com:5432/d9eat64jon4dti"
26
27 db = SQLAlchemy(app.server)
28
29
30 class Product(db.Model):
31     __tablename__ = "productlist"
32
33     Phone = db.Column(db.String(40), nullable=False, primary_key=True)
34     Version = db.Column(db.String(40), nullable=False)
35     Price = db.Column(db.Integer, nullable=False)
36     Sales = db.Column(db.Integer, nullable=False)
```



- All done. You may click 'Ctrl+F5' to run your script (as it was already setup on Step 4). **OR** you may run `'pipenv run crud_dash_postgresql.py'` **OR** you may run `'pipenv shell'` to enter the virtual environment and run `'py crud_dash_postgresql.py'`

```
PS D:\crud_dash_JC> & "C:/Users/Keanu/.virtualenvs/crud_dash_JC-7470100j/scripts/activate.ps1"
(crud_dash_JC) PS D:\crud_dash_JC> & "C:/Users/Keanu/.virtualenvs/crud_dash_JC-7470100j/scripts/python.exe" "C:/Users/Keanu/.vscode/extensions/ms-python.python-2023.16.0/pythonFiles/lib/python/debugpy/adapter/.../debugpy/launcher" "51513" "-m" "Flask" "run" "--no-debugger" "--no-reload"
* Serving Flask app "crud_dash_postgresql.py"
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
```

```
⊗ PS D:\crud_dash_JC> pipenv run crud_dash_postgresql.py
Dash is running on http://127.0.0.1:8050/

* Serving Flask app 'crud_dash_postgresql'
* Debug mode: on

Aborted!
○ PS D:\crud_dash_JC> pipenv shell
Launching subshell in virtual environment...
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\crud_dash_JC> py crud_dash_postgresql.py
Dash is running on http://127.0.0.1:8050/

* Serving Flask app 'crud_dash_postgresql'
* Debug mode: on
```