Running Code

There are 2 webserver to run. A backend and a frontend. - Python https://www.python.org/ - Next.js https://nextjs.org/

Python

- avoid using pip install
- to update python run the following from terminal in vscode
  - conda update python -y
  - conda update –all -y
  - conda clean –all -y
- to install the necessary extensions, from terminal in vscode run
  - conda install python-dotenv flask flask-cors flask-jwt-extended sqlalchemy psycopg2 plotly ipkernals flask8 autopep8 -y
  - with terminal in the backend run the following to start the backend web server
    - python run.py
- After working with Graeme to get github permissions execute the following from terminal in vscode to pull the code repository down
  - conda install git -y
  - git clone https://github.com/JLBlueDom/backend
- create a file called .env in the root directory of the new backend folder
-

add the following content, replacing xxx with the correct password, and making any other appropriate

```
DB_CONNECT=postgresql+psycopg2://wexpython:xxx@slcplwell01/wexprobi
PD_CONNECT=postgresql+psycopg2://wexpython:xxx@slcplwell01/pason_drlg
SC_CONNECT=mssql+pyodbc://wexpython:xxx@slcpwsql02/wexengscada?driver=SQL Server&MARS
ENG_CONNECT=mssql+pyodbc://wexpython:xxx@slcpwsql02/wexeng?driver=SQL Server&MARS_Con
CLIENT_URL=http://slcplwell01
IMPORT_PATH=/mnt/slcdata02/WEXGE/wexpython_automation/
BACKEND_SRV=http://slcplwell01/
FRONTEND_URL=http://slcplwell01/
JWT_SECRET_KEY=secret_key
MAIL_SERVER=smtpmta.dominionnet.com
MAIL_PORT=25
MAIL_USERNAME=Wexpythonnoreply
SECURITY_EMAIL_SENDER=wexpythonnoreply@dominionenergy.com
MAIL_DEFAULT_SENDER=wexpythonnoreply@dominionenergy.com
MSSQL_DRV=SQL Server
```

At this point Python can be used, and is ready to go

- the sqlalchemy model file that lays out the Db structure is in
  - db/wexprobi.py
- in vscode search from db.wexprobi import for usage examples

Running the backend web server

- To run the backend web server execute the following in the vscode terminal
  - python run.py
  - this is the only thing you will need to execute in the future to run the backend server
- in order to get code updates in the future run the following from terminal
  - git discard
    - this will discard any code changes allowing a code update
    - you may need to save off any changes
    - pushing changes to server will overwrite any changes others have made too
  - git pull

React

- After making sure you have github access from the previous step
- to install the necessary extensions, from terminal in vscode run
  - conda install nodejs -y
- After working with Graeme to get github permissions execute the following from terminal in vscode to pull the code repository down
  - git clone https://github.com/JLBlueDom/client_react.git
-

create a file called .env.local in the root directory of the new client_react folder with the following content

```
BACKEND_API=http://localhost:5001/
BASE_PATH=
SECRET=secret key
```

- from vscode terminal in the client_react directory run the following install and upgrade yarn
  - npm i -g yarn
  - yarn set version berry
  - yarn install
    - this needs ran anytime there are updates to the package.json file
- To run the client_react web server execute the following in the vscode terminal
  - yarn dev

  - this is the only thing you will need to execute in the future to run the future

- the dev server runs slowly, alternately you can compile the code, and start the production server for greater client speed
  - yarn build
  - yarn start

- in order to get code updates in the future run the following from terminal
  - git discard
    * this will discard any code changes allowing a code update
    * you may need to save off any changes
    * pushing changes to server will overwrite any changes others have made too
  - git pull