CS356-Lab03:1-Branden-Codd-940428984

1. sqlite3

- To view the code, change into its directory within the repository.

2. gbmodel package

- No screenshots or observations

3. Presenter architecture

- No screenshots or observations

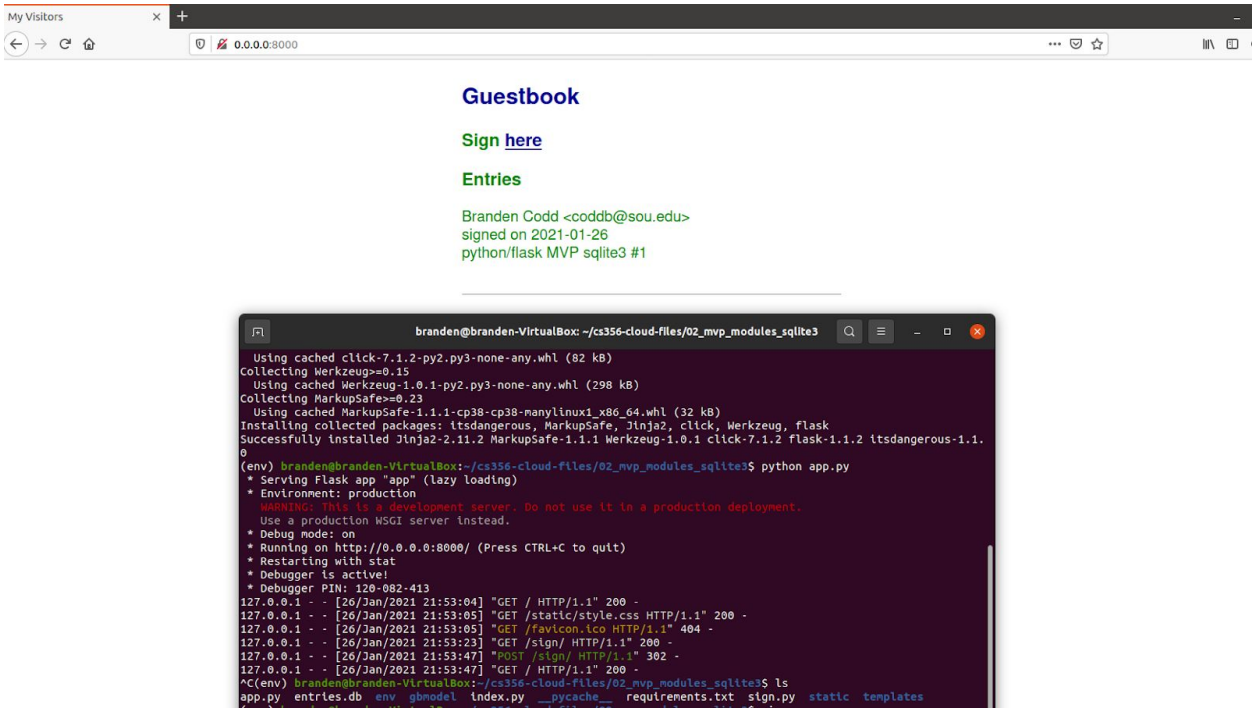
4. Running the code

- Change into the repository that contains the code.

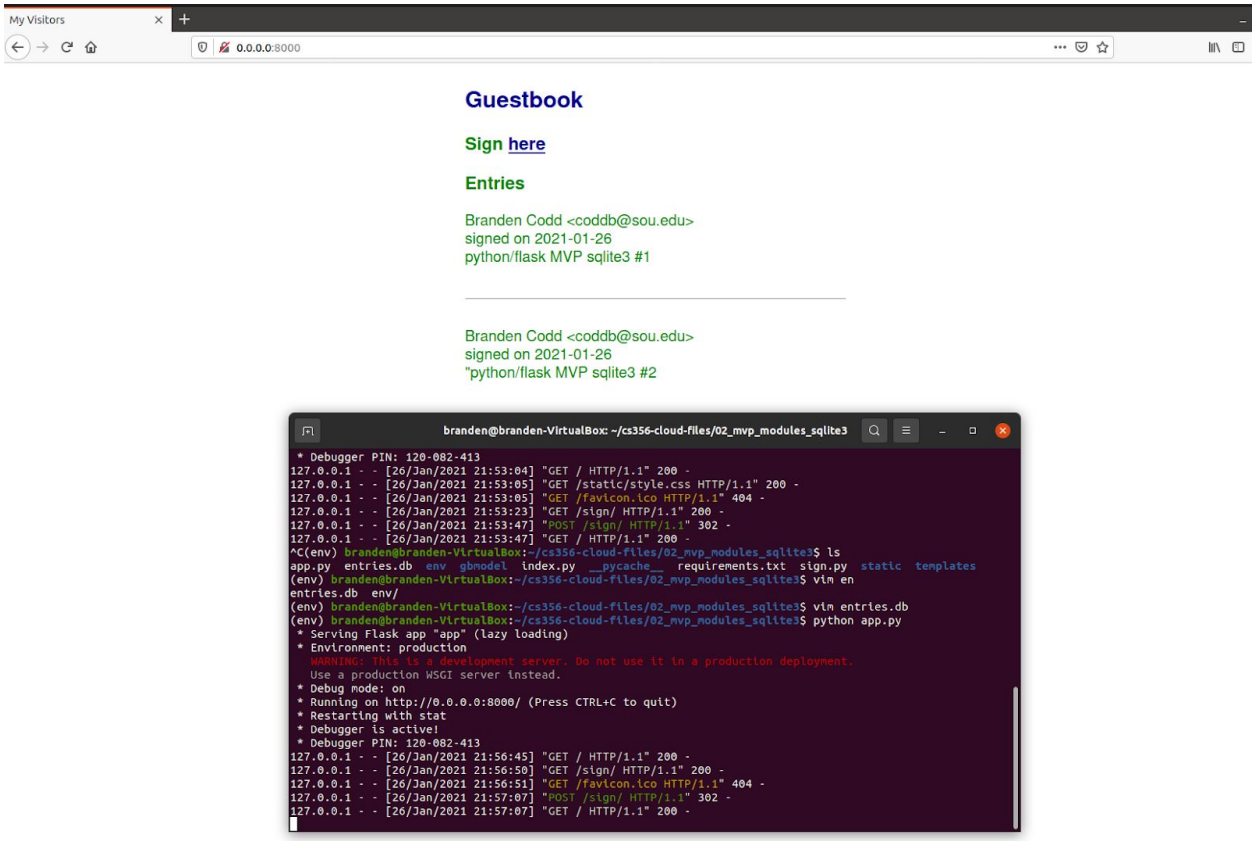
- **create a Python 3 virtual environment and install the packages specified in `requirements.txt` (e.g. `flask`)**

```
(env) branden@branden-VirtualBox:~/cs356-cloud-files/02_mvp_modules_sqlite3$ pip install -r requirements.txt
Collecting flask
  Using cached Flask-1.1.2-py2.py3-none-any.whl (94 kB)
Collecting itsdangerous>=0.24
  Using cached itsdangerous-1.1.0-py2.py3-none-any.whl (16 kB)
Collecting Jinja2>=2.10.1
  Using cached Jinja2-2.11.2-py2.py3-none-any.whl (125 kB)
Collecting click>=5.1
  Using cached click-7.1.2-py2.py3-none-any.whl (82 kB)
Collecting Werkzeug>=0.15
  Using cached Werkzeug-1.0.1-py2.py3-none-any.whl (298 kB)
Collecting MarkupSafe>=0.23
  Using cached MarkupSafe-1.1.1-cp38-cp38-manylinux1_x86_64.whl (32 kB)
Installing collected packages: itsdangerous, MarkupSafe, Jinja2, click, Werkzeug, flask
Successfully installed Jinja2-2.11.2 MarkupSafe-1.1.1 Werkzeug-1.0.1 click-7.1.2 flask-1.1.2 itsdangerous-1.1.
0
(env) branden@branden-VirtualBox:~/cs356-cloud-files/02_mvp_modules_sqlite3$ python app.py
 * Serving Flask app "app" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Running on http://0.0.0.0:8000/ (Press CTRL+C to quit)
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 120-082-413
```

- **Visit the site as before and add an entry that includes your PSU e-mail address in it and the message "python/flask MVP sqlite3 #1". Then, type "Ctrl+c" to stop the server. Perform a directory listing to see that the sqlite3 database file entries.db has been created.**



- **After restart**

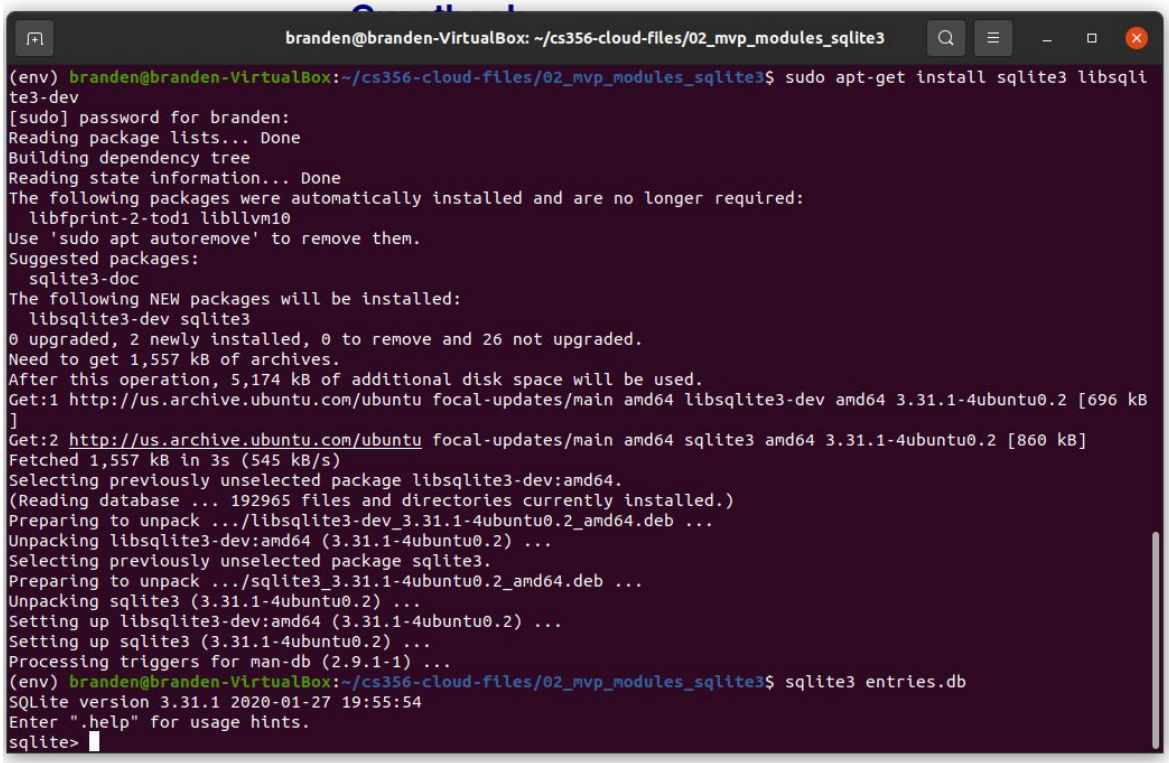


5. sqlite3 database

- **install the CLI by running the command: `sudo apt-get install sqlite3 libsqlite3-dev`**

  Bring up the entries.db database within sqlite3 via the following command:

  sqlite3 entries.db



Then, within the sqlite client, perform the following commands to:

- **List the tables in the database and note the table name**

  `sqlite> .tables`
- **Then, output the schema for the table via its name**

  `sqlite> .schema <table_name>`
- **Finally, perform a SQL query to dump out all rows in the table**

  `sqlite> select * from <table_name>;`

  Take a screenshot of the output of the above commands and include it in your lab notebook.