

# Week #3 Labs

## 3.1: Python Flask Guestbook

5. Running the code

## 3.2: ag SQL

2. SQL quiz

3. GCP Cloud SQL

7. Cloud SQL from Cloud Shell

15. RDS Test instance

## 3.3: sqlite3 Guestbook

4. Running the code

5. Sqlite3 database

## 3.1: Python Flask Guestbook

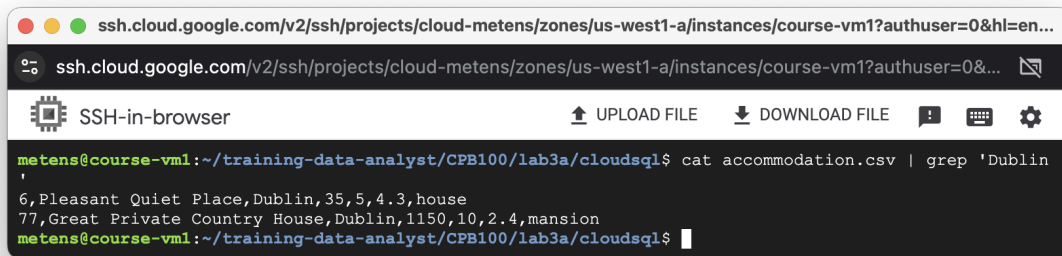
### 5. Running the code

The screenshot displays a terminal window on the left and a web browser on the right. The terminal window, titled 'Terminal - metens@course-vm1: ~/cs430-src/01\_mvc\_p', shows the process of installing dependencies (Werkzeug, Jinja2, itsdangerous, click, blinker, MarkupSafe) and running the Flask application. The output indicates that the application is running on http://127.0.0.1:5000. The web browser, titled 'My Visitors', shows the 'localhost:5000/index.html' page. The page has a title 'Guestbook' and contains three input fields: 'Name:', 'Email:', and 'Message:'. Below the 'Message:' field is a 'Sign' button. Under the 'Sign' button, there is a section titled 'Entries' which displays a list of entries: 'Nathan Metens <metens@pdx.edu> signed on 2024-10-16 python/flask guestbook'.

ssh.cloud.google.com/v2/ssh/projects/cloud-metens/zones/us-west1-a-jn... ssh.google.com/v2/ssh/projects/cloud-metens/zones/us-west1-a-jn... SSH-in-browser UPLOAD FILE DOWNLOAD FILE

```
1 CREATE DATABASE IF NOT EXISTS recommendation_spark;
2
3 USE recommendation_spark;
4
5 DROP TABLE IF EXISTS Recommendation;
6 DROP TABLE IF EXISTS Rating;
7 DROP TABLE IF EXISTS Accommodation;
8 CREATE TABLE IF NOT EXISTS Accommodation
9 (
10     id varchar(255),
11     title varchar(255),
12     location varchar(255),
13     price int,
14     rooms int,
15     rating float,
16     type varchar(255),
17     PRIMARY KEY (ID)
18 );
19
20 CREATE TABLE IF NOT EXISTS Rating
21 (
22     userId varchar(255),
23     accoId varchar(255),
24     rating int,
25     PRIMARY KEY (accoId, userId),
26     FOREIGN KEY (accoId)
27     REFERENCES Accommodation(id)
28 );
29
30 CREATE TABLE IF NOT EXISTS Recommendation
31 (
32     userId varchar(255),
33     accoId varchar(255),
34     prediction float,
35     PRIMARY KEY (userId, accoId),
36     FOREIGN KEY (accoId)
37     REFERENCES Accommodation(id)
38 );
39
40
```

28,2 All 1



```
ssh.cloud.google.com/v2/ssh/projects/cloud-metens/zones/us-west1-a/instances/course-vm1?authuser=0&hl=en...
SSH-in-browser
metens@course-vm1:~/training-data-analyst/CPB100/lab3a/cloudsql$ cat accommodation.csv | grep 'Dublin'
6,Pleasant Quiet Place,Dublin,35,5,4.3,house
77,Great Private Country House,Dublin,1150,10,2.4,mansion
metens@course-vm1:~/training-data-analyst/CPB100/lab3a/cloudsql$
```

For Dublin:

id=6, title='Pleasant Quiet Place', location=Dublin, price=35, rooms=5, rating=4.3, type=house  
id=77, title='Great Private Country House', location=Dublin, price=1150, rooms=10, rating=2.4,  
type=mansion

## 7. Cloud SQL from Cloud Shell



```
ssh.cloud.google.com/v2/ssh/projects/cloud-metens/zones/us-west1-a/instances/course-vm...
SSH-in-browser
MySQL [recommendation_spark]> select * from Accommodation where price between 50 and 100 and type='house'
-> ;
+-----+-----+-----+-----+-----+-----+
| id | title | location | price | rooms | rating | type |
+-----+-----+-----+-----+-----+-----+
| 12 | Beautiful Peaceful Villa | Seattle | 90 | 2 | 2.1 | house |
| 22 | Pleasant Peaceful House | Auckland | 50 | 5 | 3.5 | house |
| 3 | Agreeable Calm Place | London | 65 | 4 | 4.8 | house |
| 38 | Big Private House | San Francisco | 70 | 4 | 2.9 | house |
| 39 | Beautiful Calm Villa | Vancouver | 50 | 3 | 3.5 | house |
| 49 | Big Private Villa | NYC | 90 | 2 | 4.8 | house |
| 59 | Large Peaceful Place | Tokyo | 55 | 5 | 1.2 | house |
| 61 | Large Calm Place | NYC | 60 | 2 | 1.3 | house |
| 66 | Beautiful Private Villa | London | 80 | 2 | 2.4 | house |
| 72 | Beautiful Calm Place | Paris | 80 | 4 | 2.1 | house |
| 75 | Large Private Place | Berlin | 50 | 4 | 3.6 | house |
| 86 | Large Quiet House | London | 100 | 4 | 4 | house |
| 99 | Pleasant Quiet Place | NYC | 80 | 4 | 3.2 | house |
+-----+-----+-----+-----+-----+-----+
13 rows in set (0.002 sec)

MySQL [recommendation_spark]>
```

## 15. RDS Test instance

```
}
}
}
eee_W_3679152@runweb140458:~$ ls
eee_W_3679152@runweb140458:~$
eee_W_3679152@runweb140458:~$ mysql -h aws-rds-lab.ctulgl14yuuhx.us-east-1.rds.amazonaws.com -P
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 27
Server version: 8.0.35 Source distribution

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> 
```

## 3.3: sqlite3 Guestbook

### 4. Running the code

The screenshot shows a terminal window on the left and a web browser on the right. The terminal window, titled 'Terminal - metens@course-vm1: ~/cs430-src/02\_mvp\_modules\_sqlite3', displays the output of running a Flask application. It shows the app starting on http://127.0.0.1:5000 and receiving several requests. The web browser, titled 'My Visitors', shows the URL '127.0.0.1:5000' and displays the 'Guestbook' application. The application has a 'Sign here' link and a list of 'Entries'. The first entry is by Nathan Metens, signed on 2024-10-17, with the text 'python/flask MVP sqlite3 #1'. The second entry is also by Nathan Metens, signed on 2024-10-17, with the text 'python/flask MVP sqlite3 #2'.

```
app.py  env  index.py  sign.py  templates
(env) metens@course-vm1:~/cs430-src/02_mvp_modules_sqlite3$ vim entries
(env) metens@course-vm1:~/cs430-src/02_mvp_modules_sqlite3$ python app.py
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production de
Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://10.138.0.3:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 736-389-029
127.0.0.1 - - [17/Oct/2024 23:41:28] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [17/Oct/2024 23:41:28] "GET /static/style.css HTTP/1.1" 304 -
127.0.0.1 - - [17/Oct/2024 23:41:29] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [17/Oct/2024 23:41:29] "GET /static/style.css HTTP/1.1" 304 -
127.0.0.1 - - [17/Oct/2024 23:41:31] "GET /sign HTTP/1.1" 200 -
127.0.0.1 - - [17/Oct/2024 23:41:31] "GET /static/style.css HTTP/1.1" 304 -
127.0.0.1 - - [17/Oct/2024 23:41:37] "POST /sign HTTP/1.1" 302 -
127.0.0.1 - - [17/Oct/2024 23:41:37] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [17/Oct/2024 23:41:37] "GET /static/style.css HTTP/1.1" 304 -
```

Guestbook

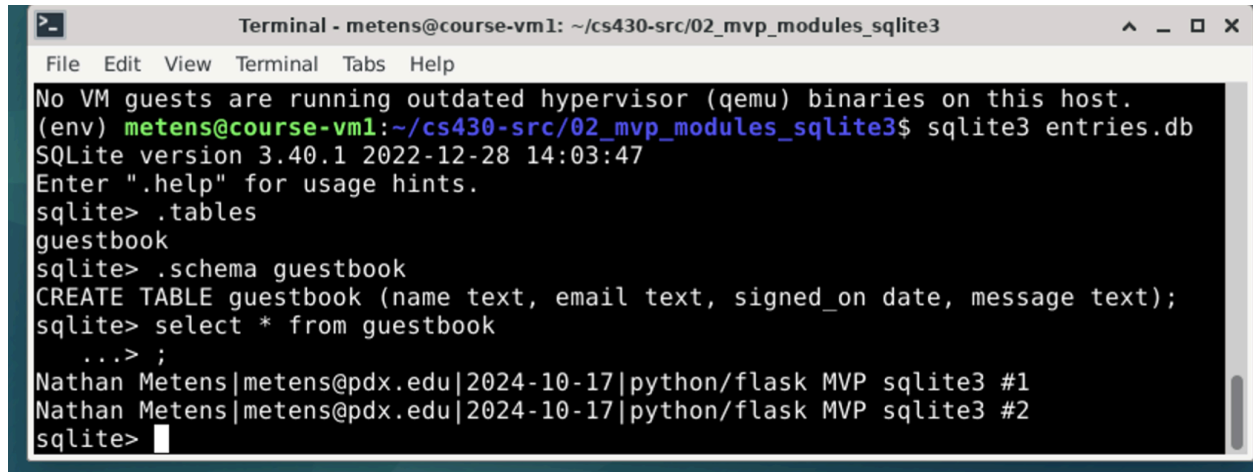
[Sign here](#)

Entries

Nathan Metens <metens@pdx.edu>  
signed on 2024-10-17  
python/flask MVP sqlite3 #1

Nathan Metens <metens@pdx.edu>  
signed on 2024-10-17  
python/flask MVP sqlite3 #2

## 5. Sqlite3 database

A terminal window titled "Terminal - metens@course-vm1: ~/cs430-src/02\_mvp\_modules\_sqlite3". The window contains a menu bar with "File", "Edit", "View", "Terminal", "Tabs", and "Help". The terminal output shows the execution of the sqlite3 command on a file named "entries.db". It displays the SQLite version (3.40.1) and a timestamp. The user enters ".help" for usage hints, ".tables" to list tables (showing "guestbook"), ".schema guestbook" to show the table schema (CREATE TABLE guestbook (name text, email text, signed\_on date, message text);), and "select \* from guestbook" to retrieve data. The output shows two rows of data for "Nathan Metens".

```
> Terminal - metens@course-vm1: ~/cs430-src/02_mvp_modules_sqlite3
File Edit View Terminal Tabs Help
No VM guests are running outdated hypervisor (qemu) binaries on this host.
(env) metens@course-vm1:~/cs430-src/02_mvp_modules_sqlite3$ sqlite3 entries.db
SQLite version 3.40.1 2022-12-28 14:03:47
Enter ".help" for usage hints.
sqlite> .tables
guestbook
sqlite> .schema guestbook
CREATE TABLE guestbook (name text, email text, signed_on date, message text);
sqlite> select * from guestbook
...> ;
Nathan Metens|metens@pdx.edu|2024-10-17|python/flask MVP sqlite3 #1
Nathan Metens|metens@pdx.edu|2024-10-17|python/flask MVP sqlite3 #2
sqlite> 
```