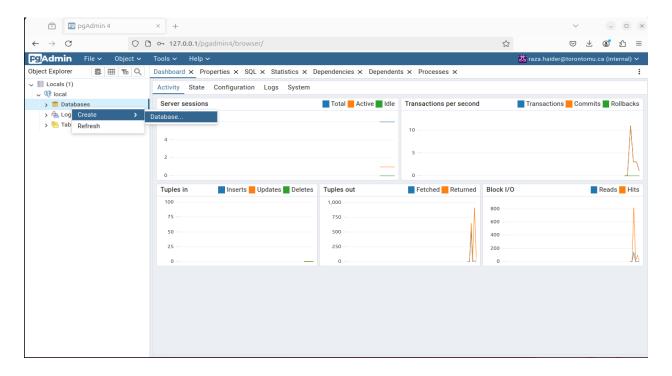
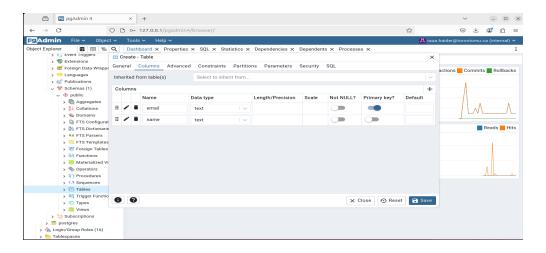
LAB 5 - MIDDLEWARE WITH A RELATIONAL BACKEND

Date: October 29, 2024

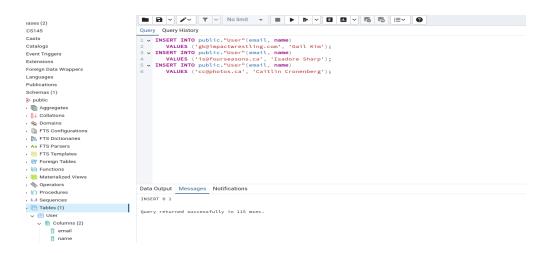
1 Create a database named "CKCS145"



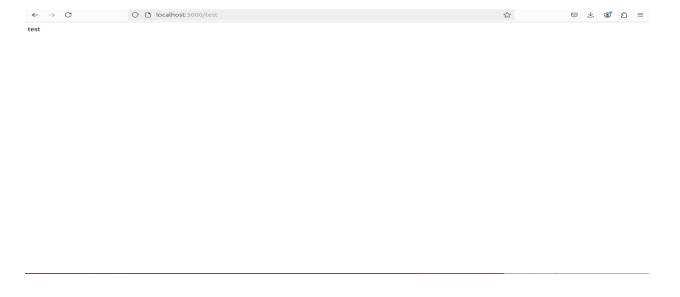
2 Create table two columns "email" and "name"



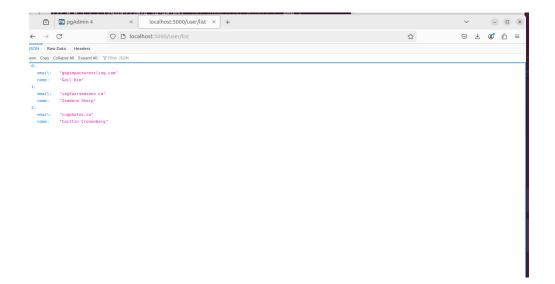
3 Insert data using SQL commands



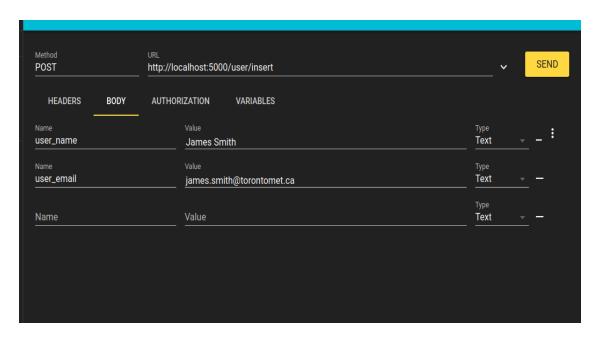
4 Run python code and navigate to http://localhost:5000/test

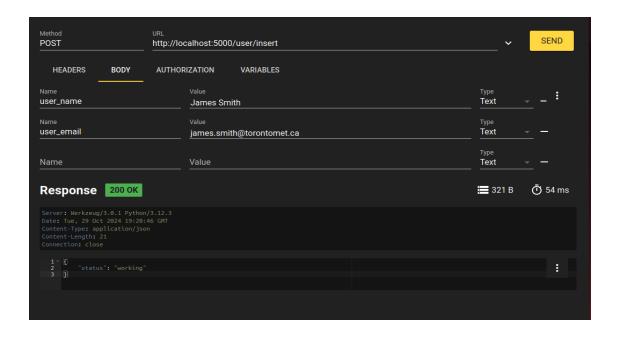


5 Retrieving all data from PostgreSQL by navigating to http://localhost:5000/user/list



6 POST to insert request using RESTer





```
* Serving Flask app 'lab5-appa'

* Debug mode: off

* MARNING: 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

127.0.0.1 - - [29/Oct/2024 15:09:25] "GET /test HTTP/1.1" 200 -

[User(email='gk@impactwrestling.com', name='Gail Kim'), User(email='is@fourseaso ns.ca', name='Isadore Sharp'), User(email='cc@photos.ca', name='Caitlin Cronenberg')]

127.0.0.1 - - [29/Oct/2024 15:09:36] "GET /list HTTP/1.1" 200 -

127.0.0.1 - - [29/Oct/2024 15:10:06] "GET /user/list HTTP/1.1" 200 -

127.0.0.1 - - [29/Oct/2024 15:19:39] "POST /user/insert HTTP/1.1" 404 -

127.0.0.1 - - [29/Oct/2024 15:20:22] "POST /user/insert HTTP/1.1" 404 -

*X^Craza@razahaider:-/Documents/Full-Stack-Developer/lab5/lab5$ python3 lab5-apy

* Serving Flask app 'lab5-appa'

* Debug mode: off

*MARNING: 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

data submitted: James Smith james.smith@torontomet.ca

127.0.0.1 - - [29/Oct/2024 15:20:46] "POST /user/insert HTTP/1.1" 200 -
```

```
→ C | localhost:5000/user/list

JSON Raw Data Headers

Save Copy Collapse All Expand All | Filter JSON

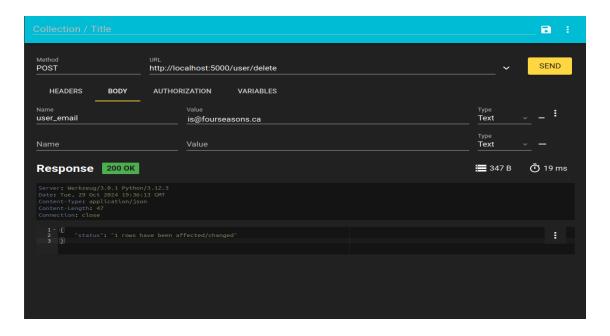
▼ 0:
    email: "gk@impactwrestling.com"
    name: "Gail Kim"

▼ 1:
    email: "is@fourseasons.ca"
    name: "Isadore Sharp"

▼ 2:
    email: "cc@photos.ca"
    name: "Caitlin Cronenberg"

▼ 3:
    email: "james.smith@torontomet.ca"
    name: "James Smith"
```

7 POST to delete using RESTer



```
Save Copy Collapse All Expand All ▼ Filter JSON

▼ 0:

email: "gk@impactwrestling.com"

name: "Gail Kim"

▼ 1:

email: "cc@photos.ca"

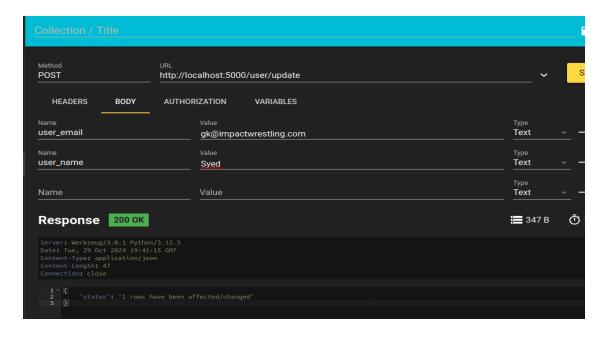
name: "Caitlin Cronenberg"

▼ 2:

email: "james.smith@torontomet.ca"

name: "James Smith"
```

8 POST update using RESTer



9 In this lab, only GET and POST requests are used. Why are PUT and DELETE requests not used?

The lab uses GET and POST for simplicity and compatibility, while PUT and DELETE are not needed for the basic scenarios covered in the lab.