

Codespaces

Edit localhost - codespaces-blank

potential-space-lamp-699995q6v99j25x54.github.dev

GmailYouTubeAdobe Acrobat

codespaces-blank [Codespaces: potential space lamp]

codespaces-blank [Codespaces: potential space lamp]

DATABASE

localhost

- Eshop 32k
 - Query
 - Tables
 - Views
 - Functions
 - Procedures

app.py

dbservice.py

create-db-template.sql

create-table-template.sql

Edit localhost

Success!

Connection Success! Cost: 43ms

Name

localhost

* Group

Parent/Sub

Advanced

* Scope

Global

Premium Only

Workspace

Note:

Requires a Premium License to unlock all features. (Limited to 3 connections) Database 1/3 , NoSQL 0/3

Server Type

MySQL

MariaDB

PostgreSQL

SQLite

SQL Server

Db2

Oracle

Cassandra

ClickHouse

SSH

Docker

Redis

ElasticSearch

MongoDB

S3

FTP

etcd

Kafka

Hive

More

Config

Main

SSH Tunnel

Socks Proxy

HTTP Proxy

* Host

127.0.0.1

* Port

3306

* Username

Kavina

* Password

.....

Database

Database

Advanced

Socket Path

e.g. /var/run/mysqld/mysqld.sock

Use

Features

Event

Trigger

Use Connection String

SSL

Save

Connect

Close

Codespaces: potential space lamp

0 0 1

localhost

Layout: US

Codespaces

create-db-template.sql - codesp...

potential-space-lamp-699995q6v99j25x54.github.dev

GmailYouTubeAdobe Acrobat

All Bookmarks

codespaces-blank [Codespaces: potential space lamp]

DATABASE

localhost

Eshop 32k

Query

Tables

Views

Functions

Procedures

app.py

dbservice.py

create-db-template.sql

create-table-template.sql

Edit localhost

home > codespace > .dbclient > storage > 1732691260052@@127.0.0.1@3306 > create-db-template.sql > ...

-- Active: 1732691260052@@127.0.0.1@3306localhost MySQL

Run | New Tab

CREATE DATABASE Eshop

DEFAULT CHARACTER SET = 'utf8mb4'; 4ms AffectedRows: 1

Codespaces: potential space lamp

Ln 3, Col 39 Spaces: 4 UTF-8 LF MS SQL Layout: US

Codespaces

create-table-template.sql - cod

potential-space-lamp-699995q6v99j25x54.github.dev

GmailYouTubeAdobe Acrobat

codespaces-blank [Codespaces: potential space lamp]

DATABASE

localhost 8.0.40-0ubu...

Eshop 32k

Query

Tables (2)

Customer 2

Columns

id int

name varchar(250)

mail varchar(250)

Index

Partitions

Empt

Columns

statement varchar(250)

Index

Partitions

Views

Functions

Procedures

app.py

dbservice.py

create-db-template.sql

create-table-template.sql X

Edit localhost

home > codespace > .dbclient > storage > 1732691260052@@127.0.0.1@3306@Eshop > create-table-template.sql > ...

1

-- Active: 1732691260052@@127.0.0.1@3306@Eshoplocalhost MySQL

2

CREATE TABLE Customer(

3

id INT,

4

name VARCHAR(250),

5

mail VARCHAR(250)

6

);

7

Run | Reset

8

INSERT INTO Customer

9

(id, name, mail) VALUES

10

(1,'kavi','kavi@gmail'),

11

(2,'kavina','kavina@gmail');

12

Run | Reset | JSON

13

SELECT * FROM Customer;

14

Run | Reset

15

CREATE Table Empt(statement VARCHAR(250));

16

INSERT INTO Empt(statement) VALUES('NO DATA FOUND');

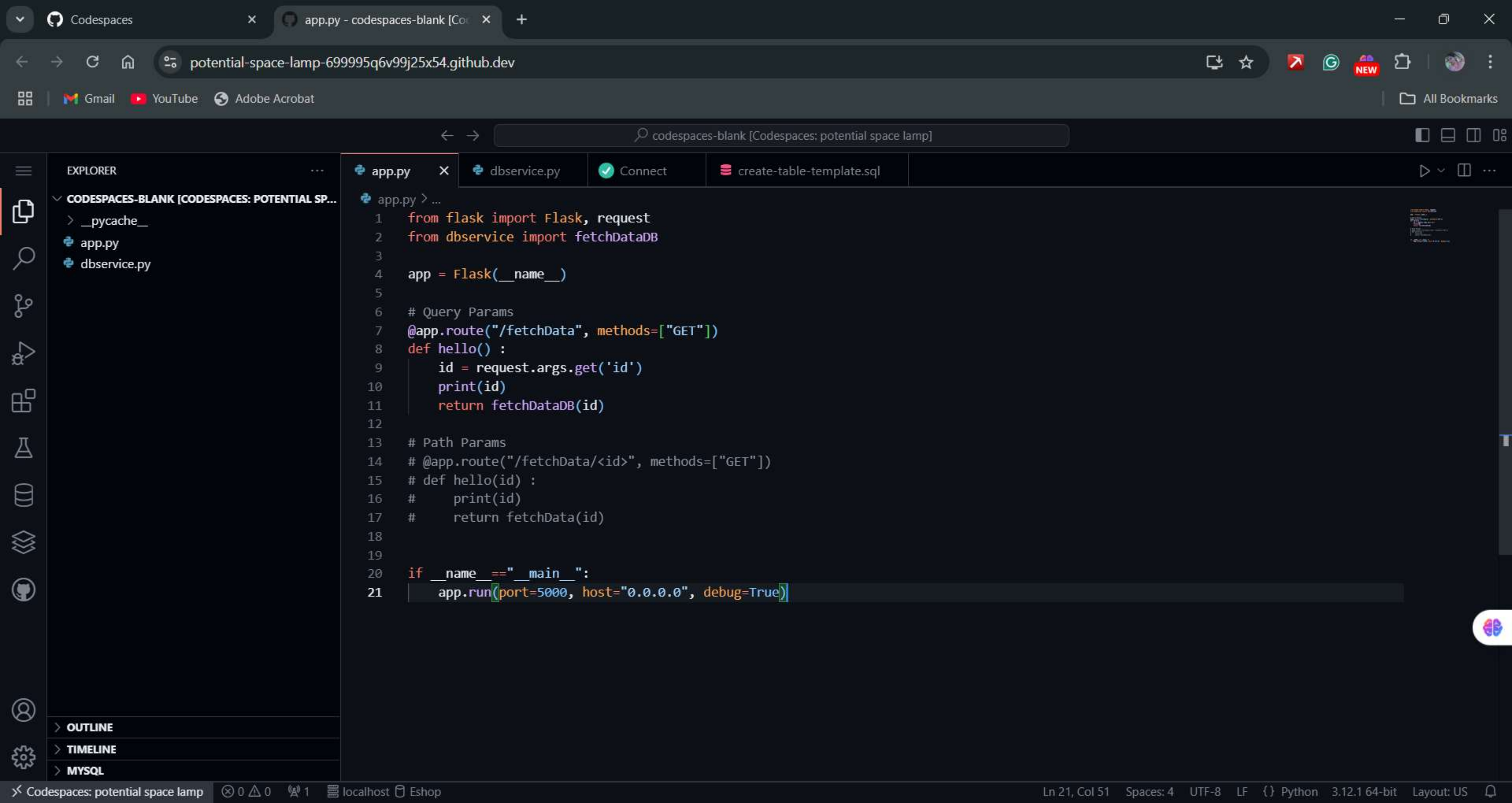
17

SELECT * FROM Empt;

18

Codespaces: potential space lamp

Ln 18, Col 1 Spaces: 4 UTF-8 LF MS SQL Layout: US



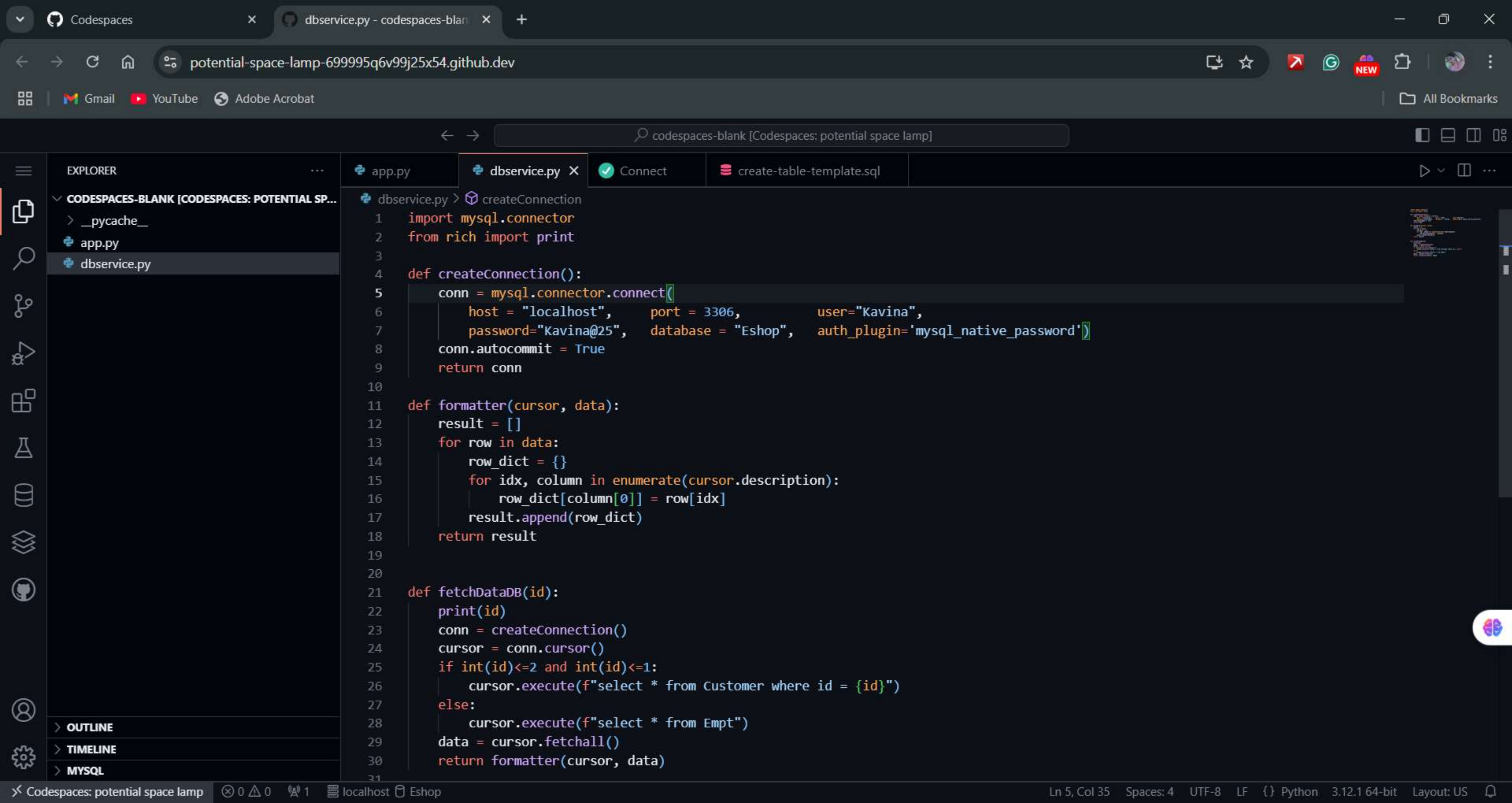
```
from flask import Flask, request
from dbservice import fetchDataDB

app = Flask(__name__)

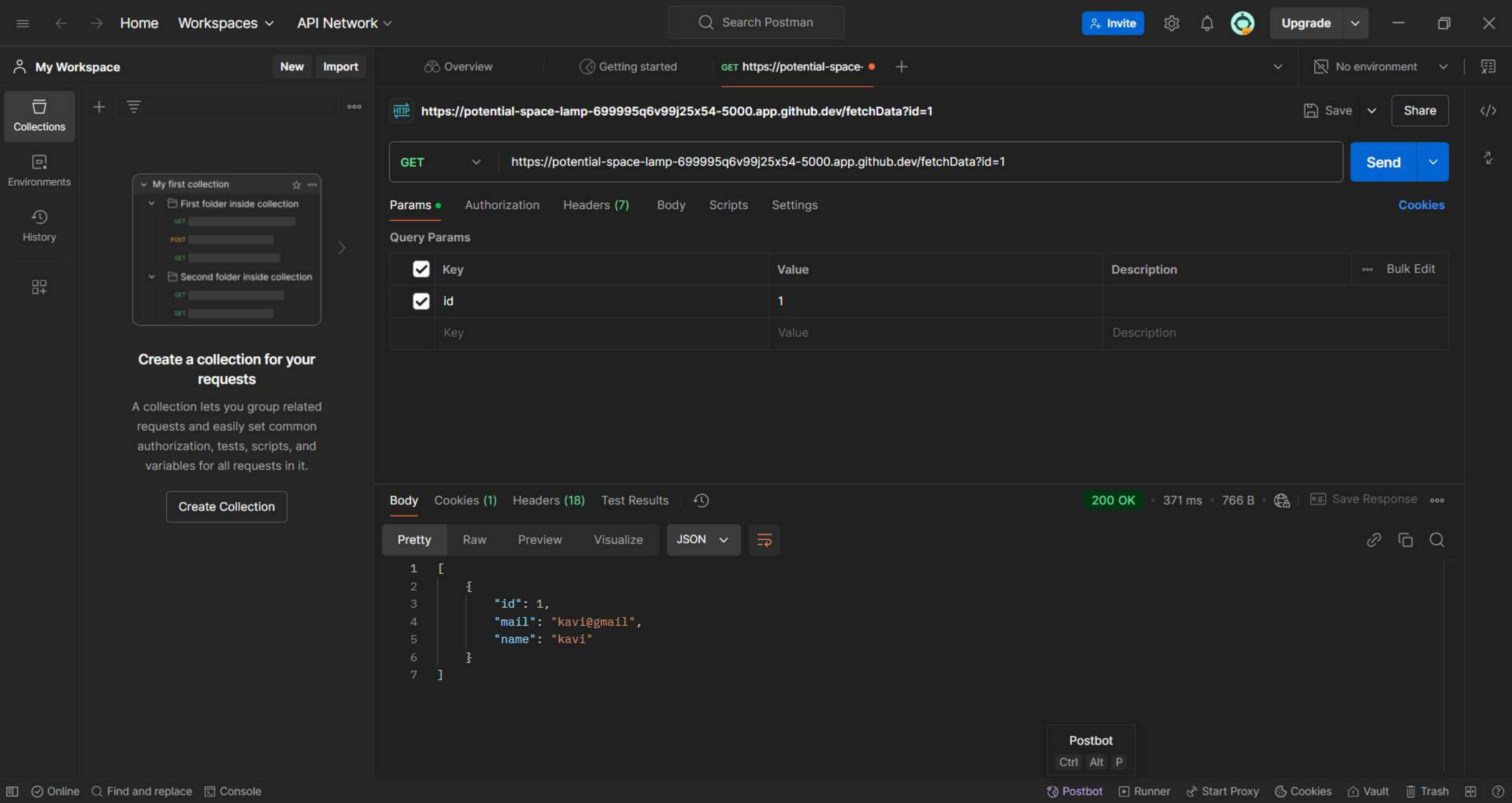
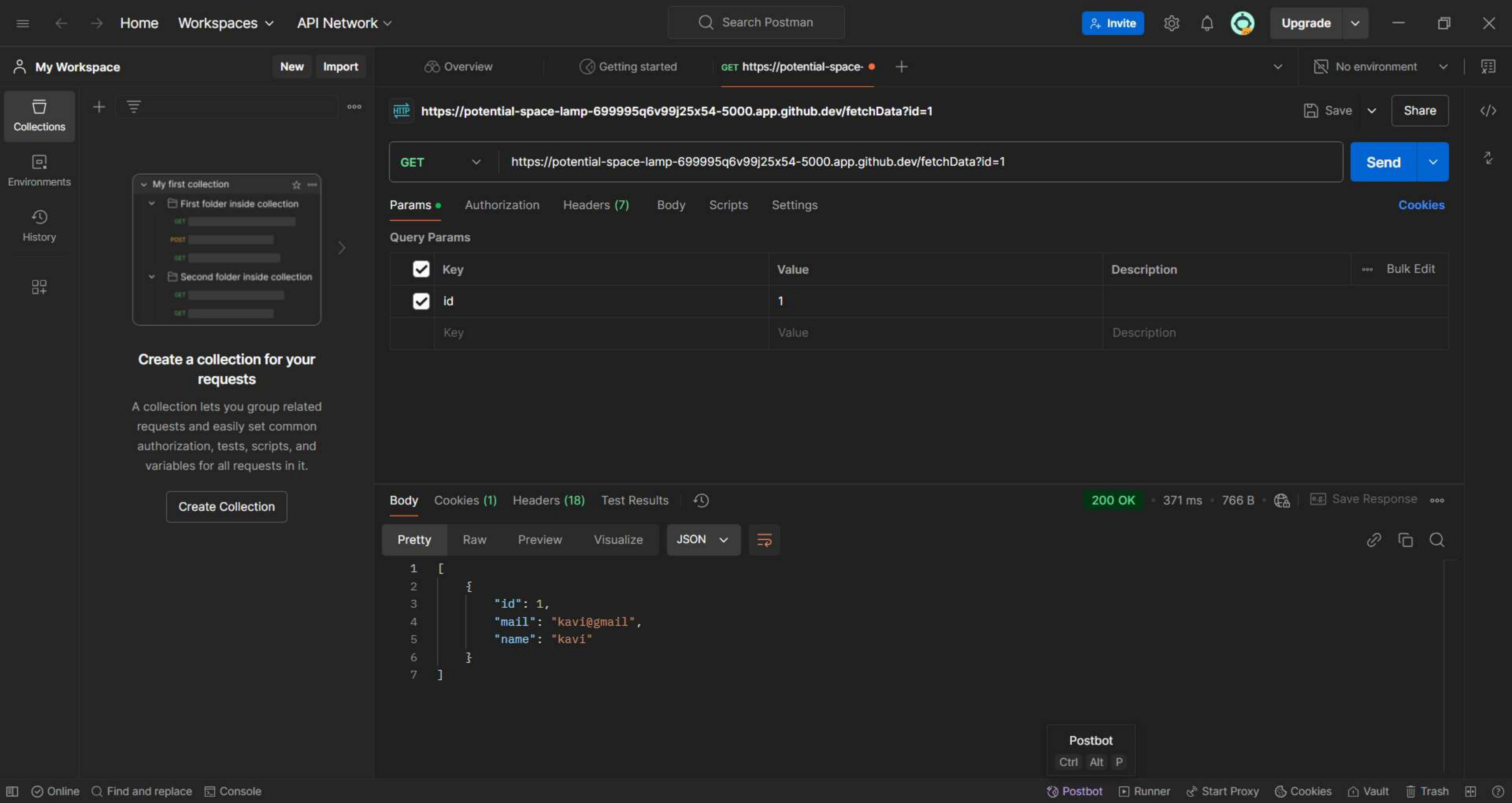
# Query Params
@app.route("/fetchData", methods=["GET"])
def hello() :
    id = request.args.get('id')
    print(id)
    return fetchDataDB(id)

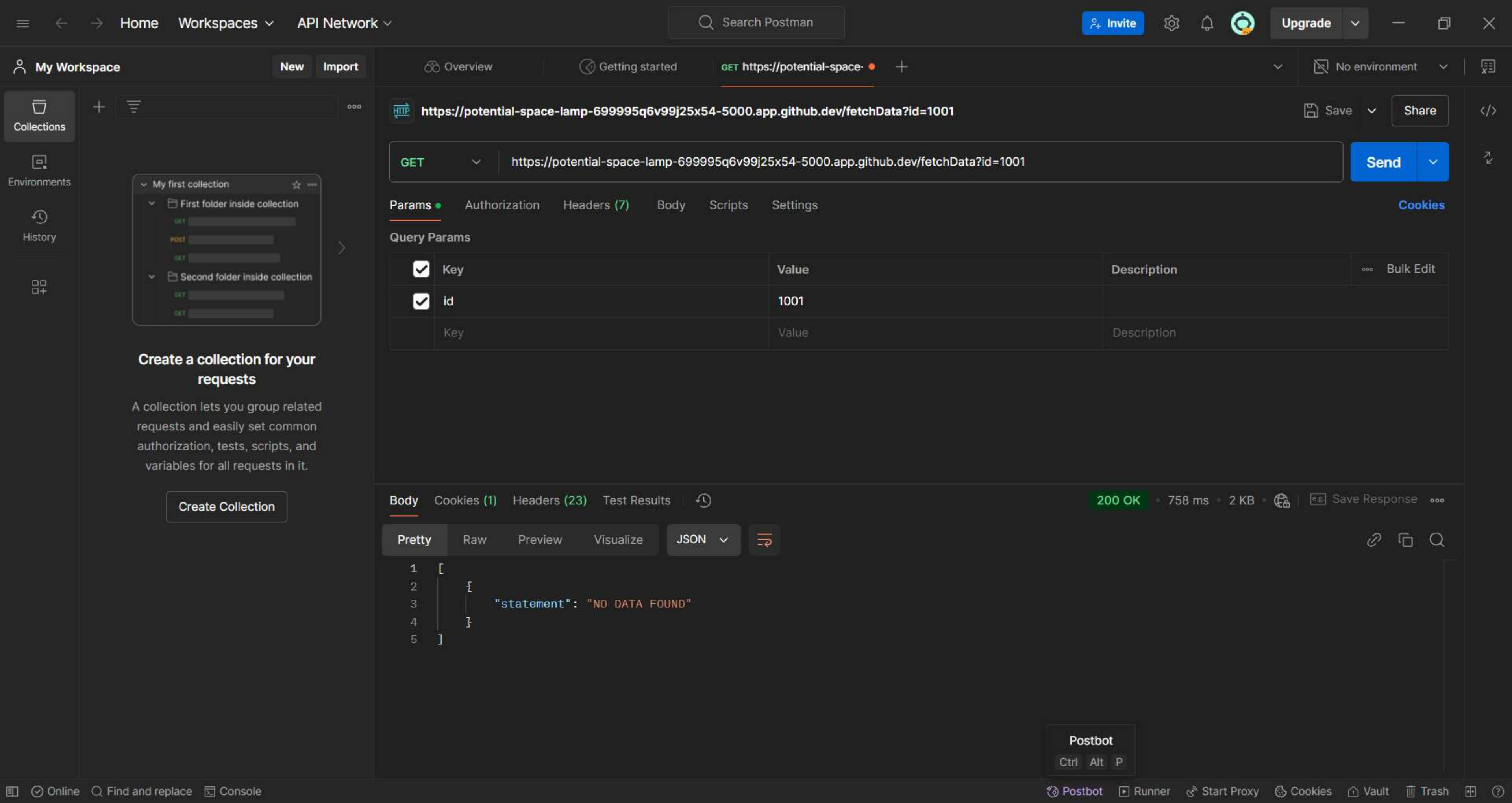
# Path Params
# @app.route("/fetchData/<id>", methods=["GET"])
# def hello(id) :
#     print(id)
#     return fetchData(id)

if __name__ == "__main__":
    app.run(port=5000, host="0.0.0.0", debug=True)
```

```
dbservice.py > createConnection
1 import mysql.connector
2 from rich import print
3
4 def createConnection():
5     conn = mysql.connector.connect(
6         host = "localhost", port = 3306, user="Kavina",
7         password="Kavina@25", database = "Eshop", auth_plugin='mysql_native_password')
8     conn.autocommit = True
9     return conn
10
11 def formatter(cursor, data):
12     result = []
13     for row in data:
14         row_dict = {}
15         for idx, column in enumerate(cursor.description):
16             row_dict[column[0]] = row[idx]
17         result.append(row_dict)
18     return result
19
20
21 def fetchDataDB(id):
22     print(id)
23     conn = createConnection()
24     cursor = conn.cursor()
25     if int(id)<=2 and int(id)<=1:
26         cursor.execute(f"select * from Customer where id = {id}")
27     else:
28         cursor.execute(f"select * from Empt")
29     data = cursor.fetchall()
30     return formatter(cursor, data)
31
```





Collections

Environments

History

My first collection

First folder inside collection

POST

GET

Second folder inside collection

GET

GET

Create a collection for your requests

A collection lets you group related requests and easily set common authorization, tests, scripts, and variables for all requests in it.

Create Collection

https://potential-space-lamp-699995q6v99j25x54-5000.app.github.dev/fetchData?id=1001

GET

Send

Params

Authorization

Headers (7)

Body

Scripts

Settings

Cookies

Query Params

<input checked="" type="checkbox"/>	Key	Value	Description	Bulk Edit
<input checked="" type="checkbox"/>	id	1001		
	Key	Value	Description	

Pretty

Raw

Preview

Visualize

JSON

```
1 [
2   {
3     "statement": "NO DATA FOUND"
4   }
5 ]
```

Codespaces

app.py - codespaces-blank [Code Editor]

potential-space-lamp-699995q6v99j25x54.github.dev

Gmail

YouTube

Adobe Acrobat

All Bookmarks

codespaces-blank [Codespaces: potential space lamp]

app.py

dbservice.py

Connect

create-table-template.sql

EXPLORED

CODESPACES-BLANK [CODESPACES: POTENTIAL SP...]

> __pycache__

app.py

dbservice.py

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

@cmkavi2504 →/workspaces/codespaces-blank \$ python3 app.py

* Serving Flask app 'app'

* 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 all addresses (0.0.0.0)

* Running on http://127.0.0.1:5000

* Running on http://10.0.1.82:5000

Press CTRL+C to quit

* Restarting with stat

* Debugger is active!

* Debugger PIN: 128-883-263

1001

1001

127.0.0.1 - - [27/Nov/2024 09:34:01] "GET /fetchData?id=1001 HTTP/1.1" 200 -

1

1

127.0.0.1 - - [27/Nov/2024 09:34:15] "GET /fetchData?id=1 HTTP/1.1" 200 -

@cmkavi2504 →/workspaces/codespaces-blank \$

bash

Python

OUTLINE

TIMELINE

MYSQL

Codespaces: potential space lamp

0 0 1

localhost Eshop

Ln 22, Col 5 Spaces: 4 UTF-8 LF {} Python 3.12.1 64-bit Layout: US