# **RECIPIES**

<ul><li>O Created</li></ul>	@May 23, 2025 5:46 PM
Tags	

Create a venv

## Backend:

MySQL setup:

db name: recipe\_db

Database:

Default   Extra   
NULL
NULL   NULL   NULL   NULL
NULL
- - - -

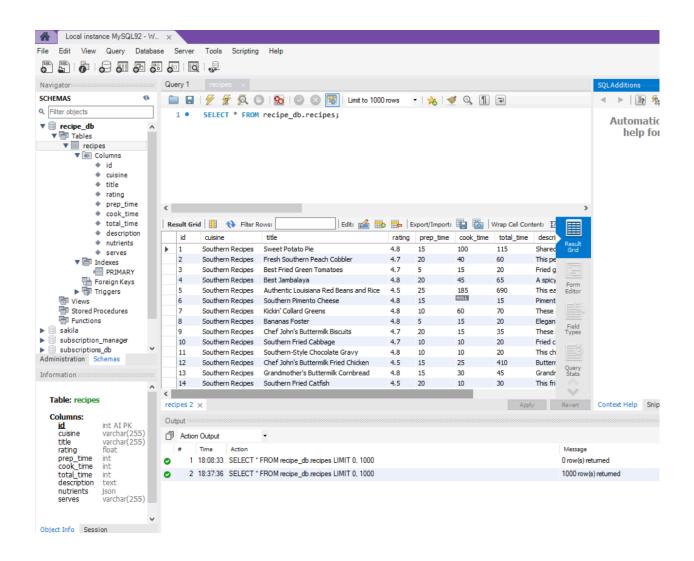
Data Preprocessing:

under data folder

- takes recipe.json as input preprocesses data as said and according to the data and stores them a new json file: "clean\_recipe.json"
- Python file: preprocess.py

#### Loading data to database:

- use mysql-connector-python to connect to mysql database and store the contents in the database.
- python file: database.py
- · data successfully updated my sql



API TESTING: (2 API endpoints)

Basic API endpoint:

## 1. Paginated and sorted display

```
Pretty-print 2

(***page** 2)
***Color: 100.

(***page** 2)
**Color: 100.
```

## 2. Search based on various fields

3. Implemented pagination for search api

## TO RUN:

 $\underline{preprocess.py} \rightarrow \text{once to create a cleaned json file}$ 

data - folder

python preprocess.py

<u>databse.py</u> → to load the cleaned input to the MySQL database

python database.py

main.py → Define API endpoints to display the data from the DB

uvicorn main:app —reload

API ENDPOINT 1: <a href="http://127.0.0.1:8000/api/recipes?page=2&limit=20">http://127.0.0.1:8000/api/recipes?page=2&limit=20</a>

(Page and limit, sorted based on the rating)

API ENDPOINT 2: <a href="http://127.0.0.1:8000/api/recipes/search?calories="https://127.0.0.1:8000/api/recipes/search?calories="https://127.0.0.1:8000/api/recipes/search?calories="https://127.0.0.1:8000/api/recipes/search?calories="https://127.0.0.1:8000/api/recipes/search?calories="https://127.0.0.1:8000/api/recipes/search?calories="https://127.0.0.1:8000/api/recipes/search?calories="https://127.0.0.1:8000/api/recipes/search?calories="https://127.0.0.1:8000/api/recipes/search?calories="https://127.0.0.1:8000/api/recipes/search?calories="https://127.0.0.1:8000/api/recipes/search?calories="https://127.0.0.1:8000/api/recipes/search?calories="https://127.0.0.1:8000/api/recipes/search?calories="https://127.0.0.1:8000/api/recipes/search?calories="https://127.0.0.1:8000/api/recipes/search?calories="https://127.0.0.1:8000/api/recipes/search?calories="https://127.0.0.1:8000/api/recipes/search?calories="https://127.0.0.1:8000/api/recipes/search?calories="https://127.0.0.1:8000/api/recipes/search?calories="https://127.0.

 $<=400\&title=pie\&rating\Rightarrow=4.5$ 

(Search based on the various fields)

#### FRONTEND:

React+Vite:

npm create vite@latest recipe-frontend --template react cd recipe-frontend npm install

npm install axios

#### FRONTEND AND BACKEND CONNECTED using CORS:

MIddleware:

app.add\_middleware(
 CORSMiddleware,
 allow\_origins=["http://localhost:5173"], # Frontend Vite dev server

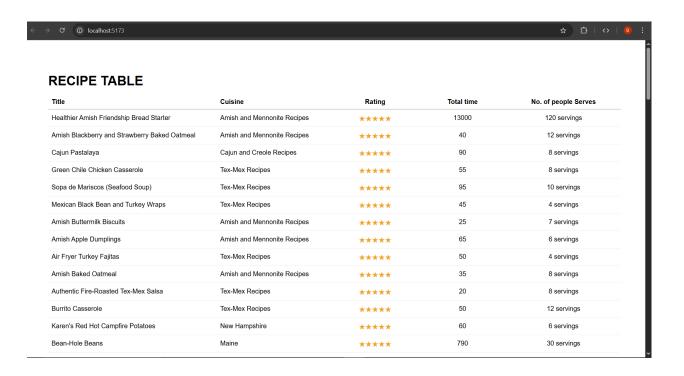
```
allow_credentials=True,
allow_methods=["*"],
allow_headers=["*"],
)
```

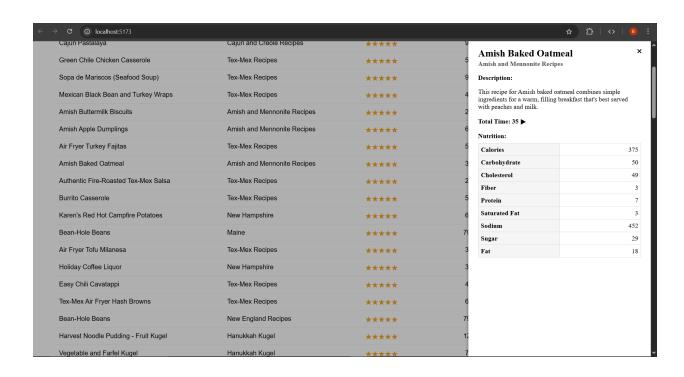
## Vite Config file:

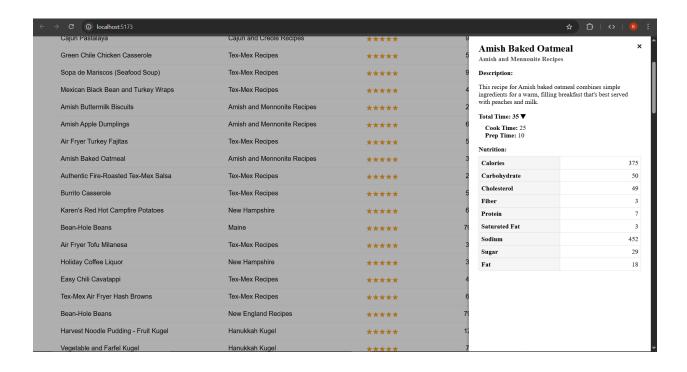
```
import { defineConfig } from "vite";
import react from "@vitejs/plugin-react";

export default defineConfig({
  plugins: [react()],
  server: {
    proxy: {
        // Proxy /api requests to your backend server
        "/api": {
        target: "http://127.0.0.1:8000", // change to your backend URL and port changeOrigin: true,
        secure: false,
        },
     },
    },
},
```

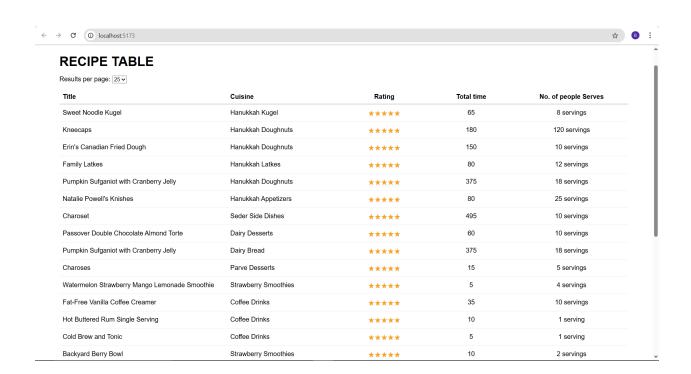
#### RECIPE TABLE PAGE:







#### Customizable page limit:



## Search Feature Integration:

