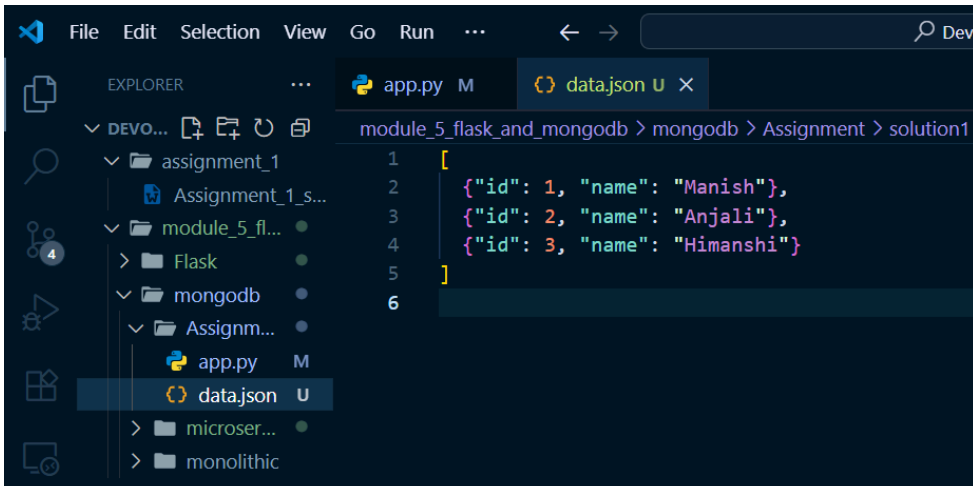
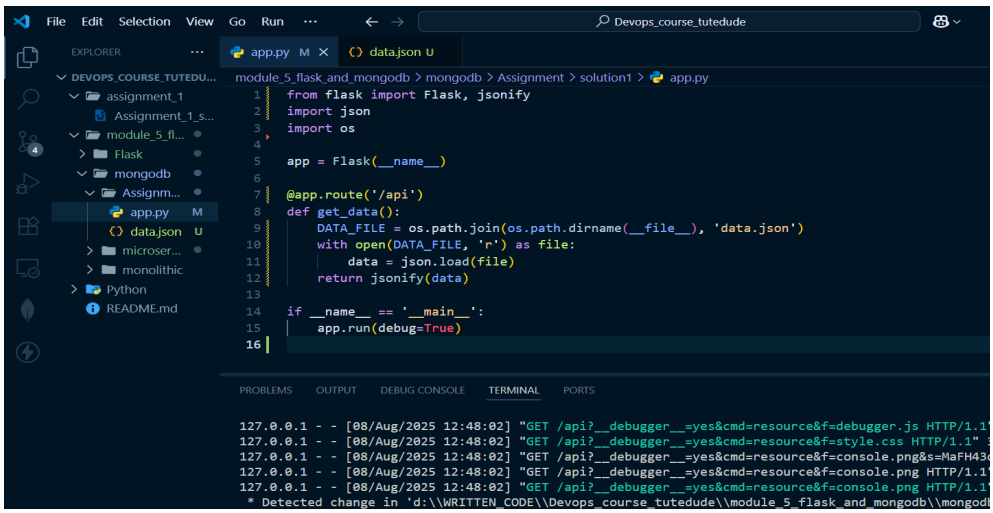
Assignment 3

1. Create a Flask application with an `/api` route. When this route is accessed, it should return a JSON list. The data should be stored in a backend file, read from it, and sent as a response.

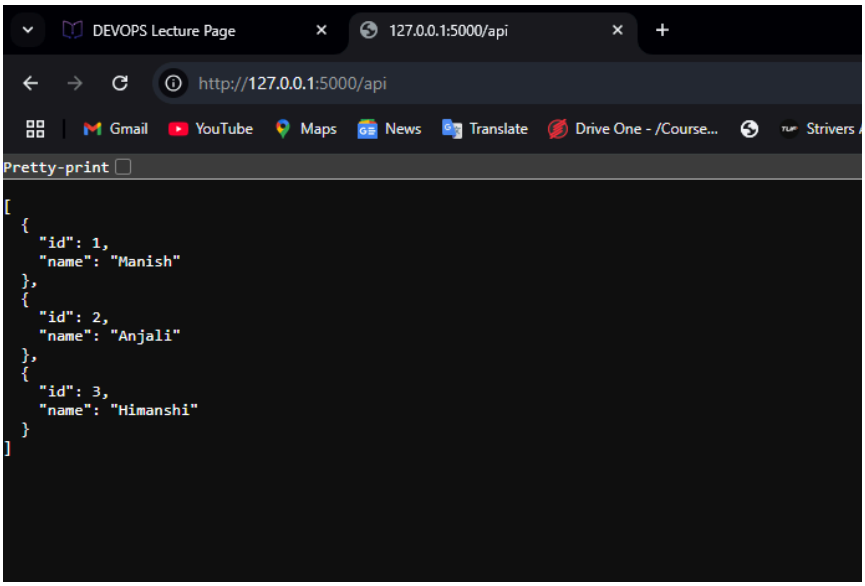
Solution1: **data.json**



app.py



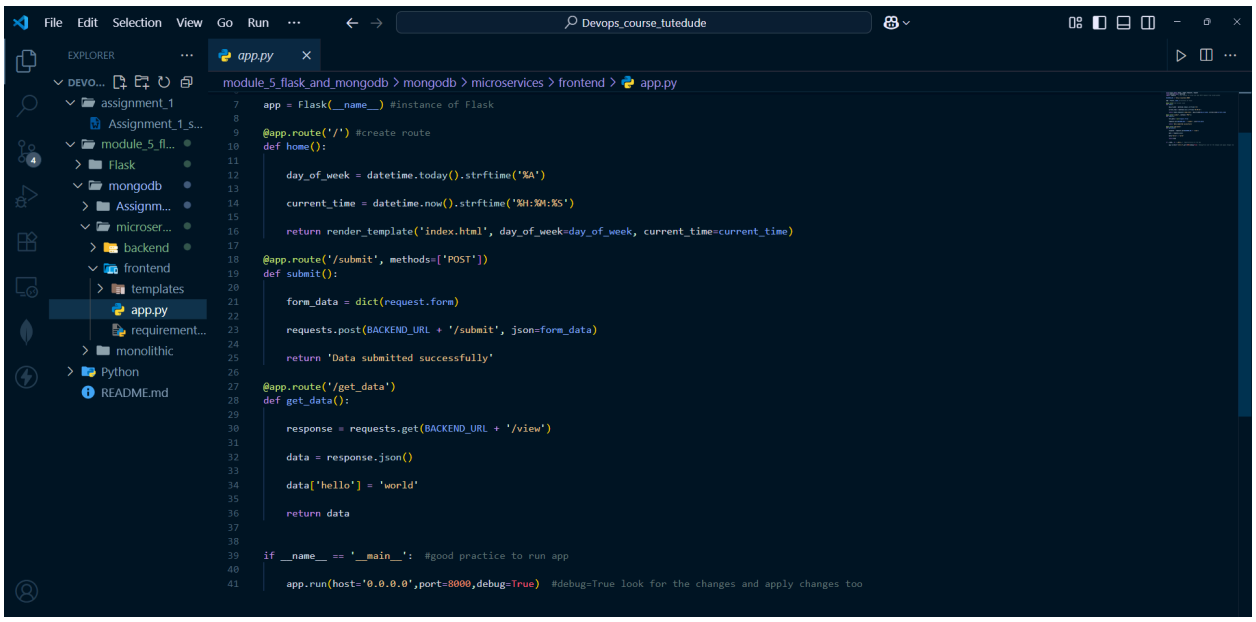
Output :



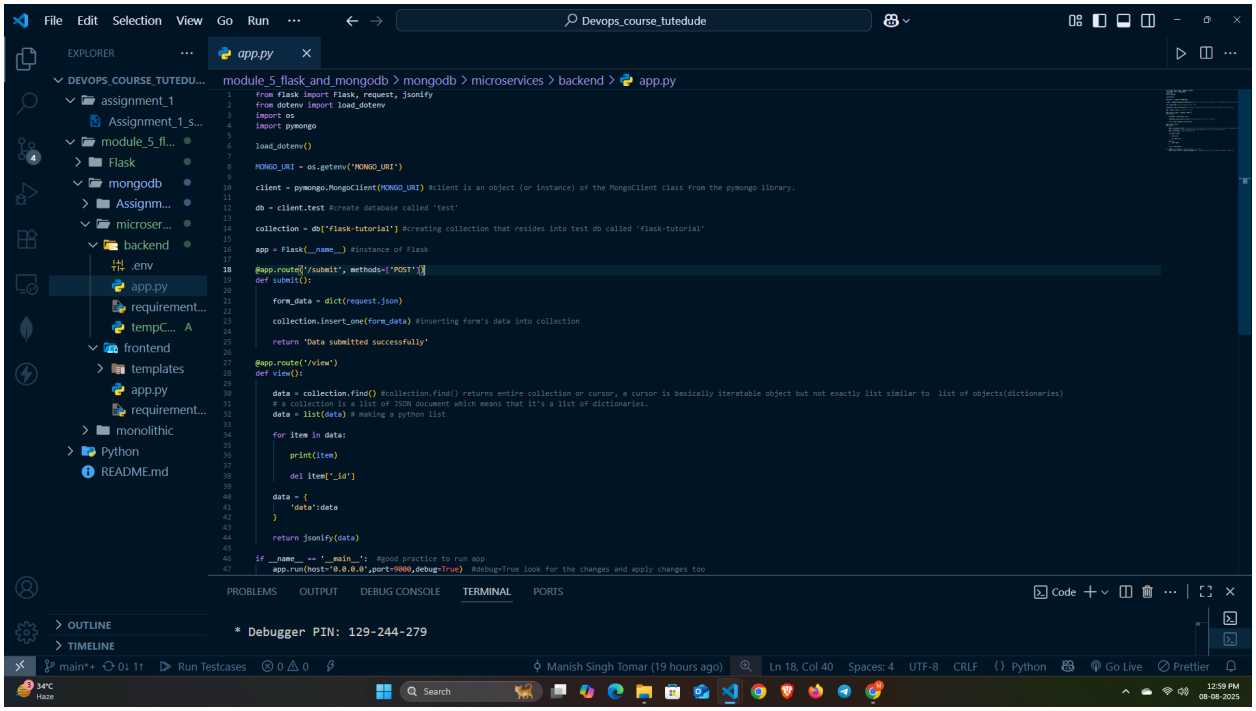
2. Create a form on the frontend that, when submitted, inserts data into MongoDB Atlas. Upon successful submission, the user should be redirected to another page displaying the message **"Data submitted successfully"**. If there's an error during submission, display the error on the same page without redirection.

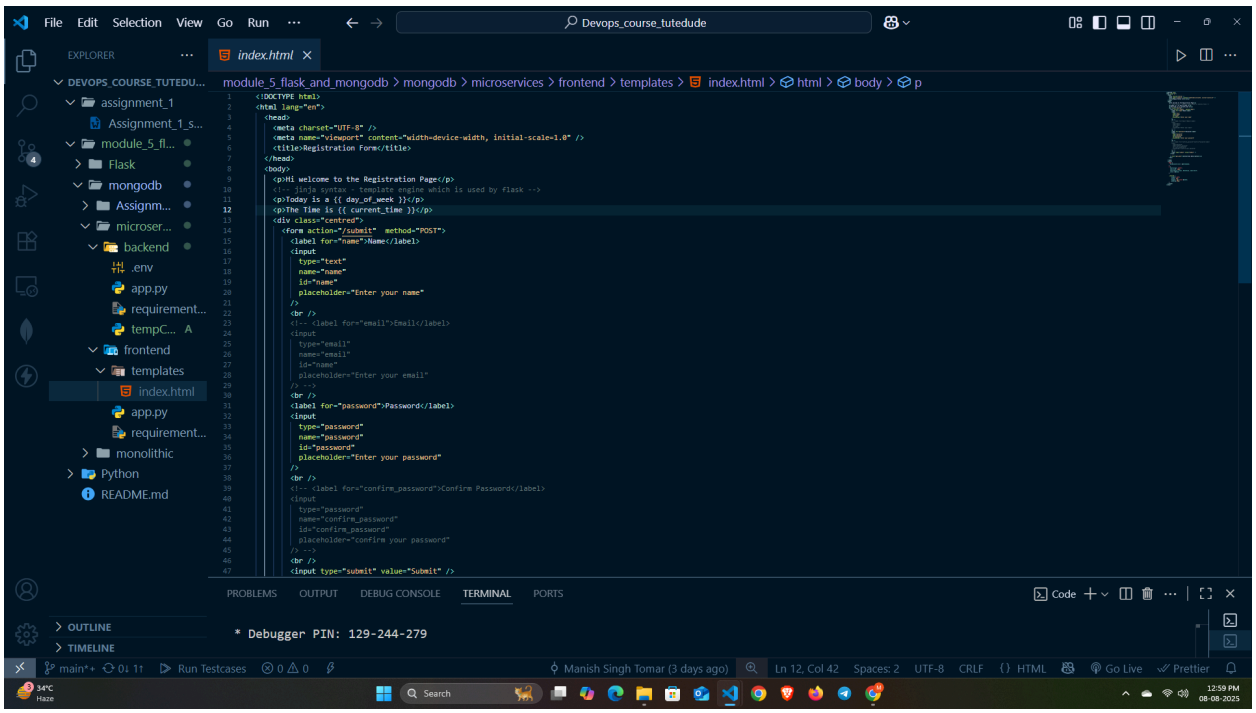
Solution 2:

Frontend:



Backend:


File Edit Selection View Go Run
Devops_course_tutedude
app.py
module_5_flask_and_mongodb > mongodb > microservices > backend > app.py
from flask import Flask, request, jsonify
from dotenv import load_dotenv
import os
import pymongo
load_dotenv()
MONGO_URI = os.getenv('MONGO_URI')
client = pymongo.MongoClient(MONGO_URI) #client is an object (or instance) of the MongoClient class from the pymongo library.
db = client.test #create database called 'test'
collection = db['flask-tutorial'] #creating collection that resides into test db called 'flask-tutorial'
app = Flask(__name__) #instance of Flask
@app.route('/submit', methods=['POST'])
def submit():
form_data = dict(request.json)
collection.insert_one(form_data) #inserting form's data into collection
return 'Data submitted successfully'
@app.route('/view')
def view():
data = collection.find() #collection.find() returns entire collection or cursor, a cursor is basically iteratable object but not exactly list similar to list of objects(dictionaries)
# a collection is a list of JSON document which means that it's a list of dictionaries.
data = list(data) # making a python list
for item in data:
print(item)
del item['_id']
data = {
'data':data
}
return jsonify(data)
if __name__ == '__main__': #good practice to run app
app.run(host='0.0.0.0',port=9000,debug=True) #debug=True look for the changes and apply changes too
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
* Debugger PIN: 129-244-279
Manish Singh Tomar (19 hours ago)
Ln 18, Col 40 Spaces: 4 UTF-8 CRLF Python Go Live Prettier
12:59 PM 08-08-2025


File Edit Selection View Go Run
Devops_course_tutedude
index.html
module_5_flask_and_mongodb > mongodb > microservices > frontend > templates > index.html > html > body > p
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<title>Registration Form</title>
</head>
<body>
Hi welcome to the Registration Page
<!-- Jinja syntax - template engine which is used by flask -->
Today is a {{ day_of_week }}
The Time is {{ current_time }}

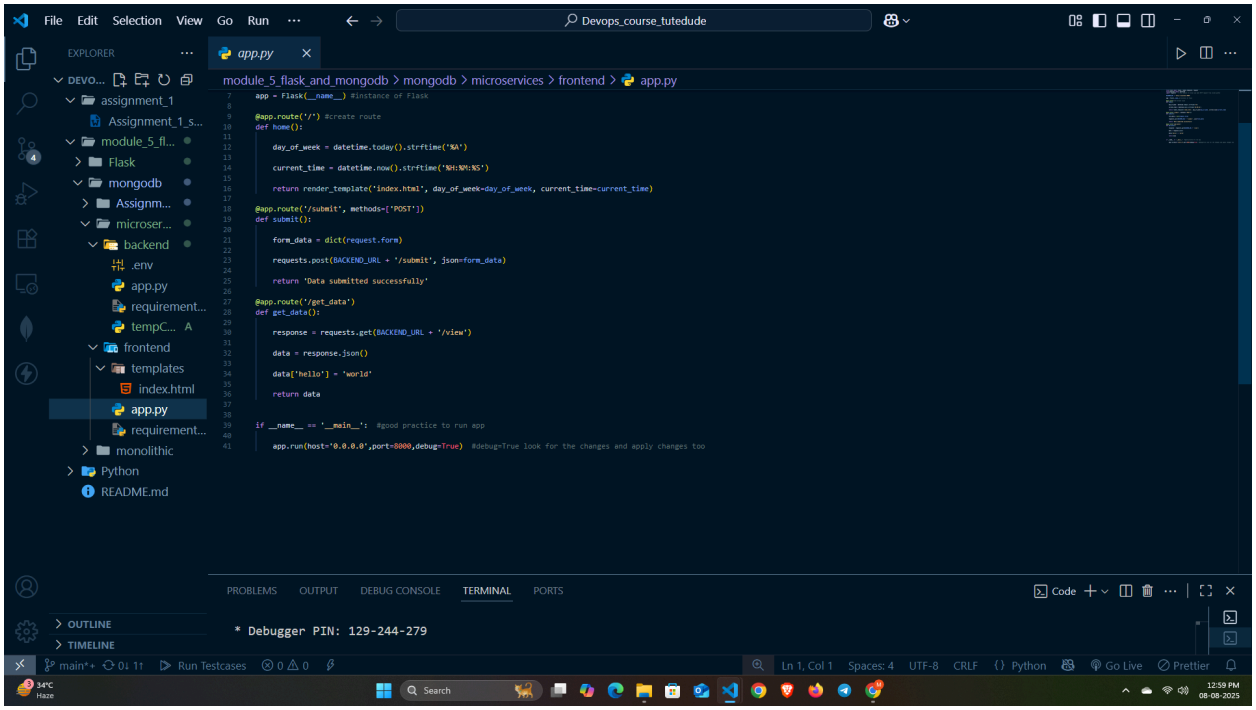
<form action="/submit" method="POST">
<label for="name">Name</label>
<input
type="text"
name="name"
id="name"
placeholder="Enter your name"
/>

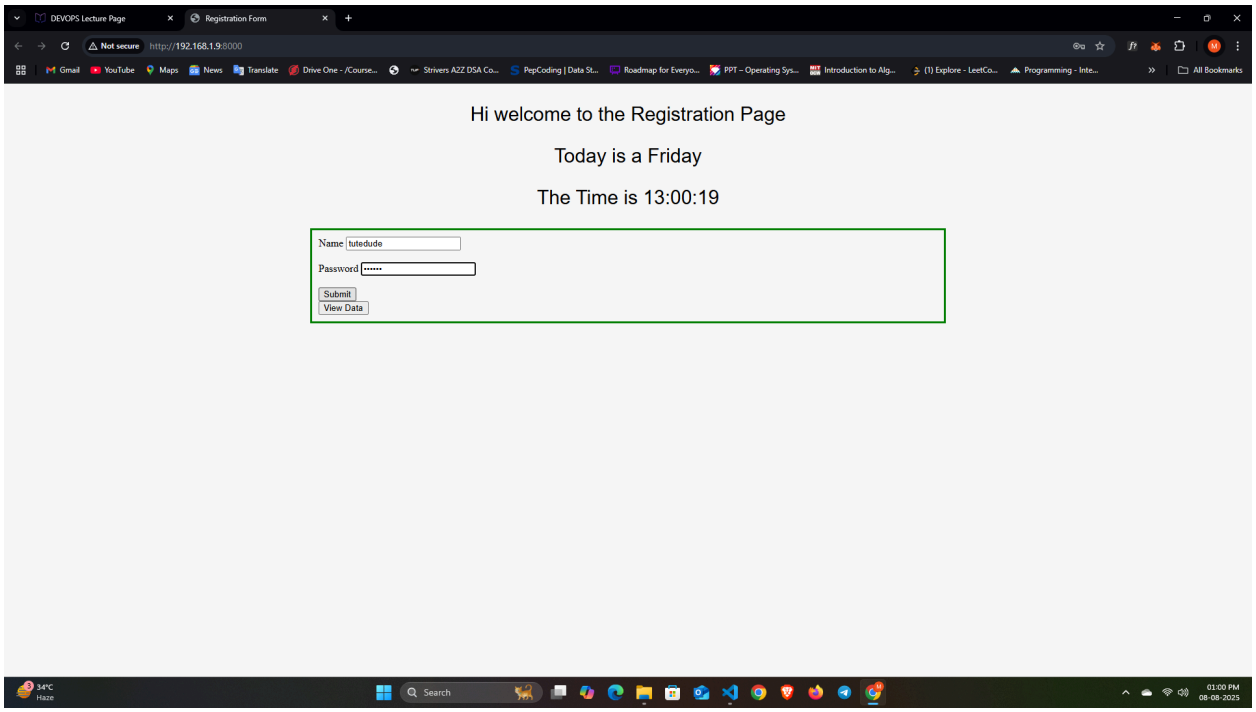
<!-- <label for="email">Email</label>
<input
type="email"
name="email"
id="name"
placeholder="Enter your email"
/> -->

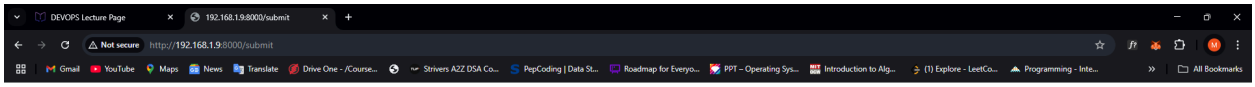
<label for="password">Password</label>
<input
type="password"
name="password"
id="password"
placeholder="Enter your password"
/>

<!-- <label for="confirm_password">Confirm Password</label>
<input
type="password"
name="confirm_password"
id="confirm_password"
placeholder="confirm your password"
/> -->

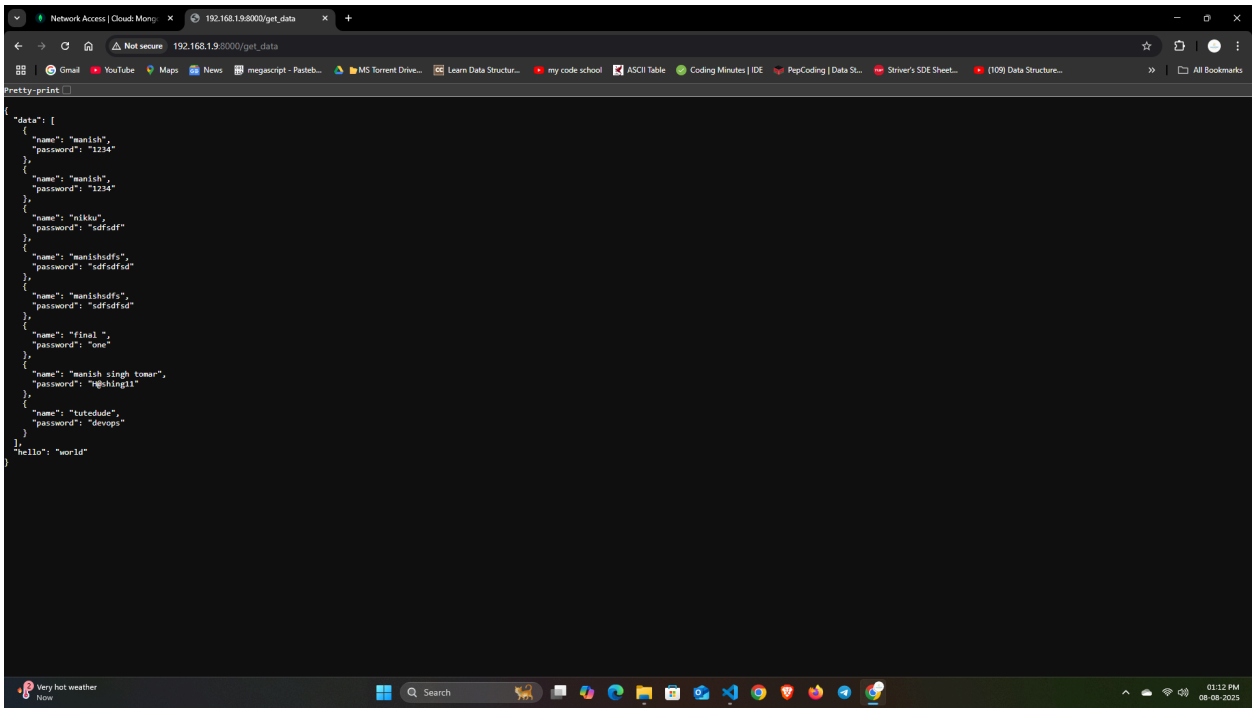
<input type="submit" value="Submit" />
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
* Debugger PIN: 129-244-279
Manish Singh Tomar (3 days ago)
Ln 12, Col 42 Spaces: 2 UTF-8 CRLF HTML Go Live Prettier
12:59 PM 08-08-2025


File Edit Selection View Go Run
Devops_course_tutedude
app.py
module_5_flask_and_mongodb > mongodb > microservices > frontend > app.py
app = Flask(__name__) #instance of Flask
@app.route('/') #create route
def home():
day_of_week = datetime.today().strftime('%A')
current_time = datetime.now().strftime('%H:%M:%S')
return render_template('index.html', day_of_week=day_of_week, current_time=current_time)
@app.route('/submit', methods=['POST'])
def submit():
form_data = dict(request.form)
requests.post(BACKEND_URL + '/submit', json=form_data)
return 'Data submitted successfully'
@app.route('/get_data')
def get_data():
response = requests.get(BACKEND_URL + '/view')
data = response.json()
data['hello'] = 'world'
return data
if __name__ == '__main__': #good practice to run app
app.run(host='0.0.0.0',port=8000,debug=True) #debug=True look for the changes and apply changes too
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
* Debugger PIN: 129-244-279


DEVOPS Lecture Page
Registration Form
Not secure http://192.168.1.9:8000
Hi welcome to the Registration Page
Today is a Friday
The Time is 13:00:19
Name tutedude
Password
Submit
View Data



Data submitted successfully




```
{
  "data": [
    {
      "name": "manish",
      "password": "1234"
    },
    {
      "name": "manish",
      "password": "1234"
    },
    {
      "name": "nikku",
      "password": "sdfsdf"
    },
    {
      "name": "manishsdfs",
      "password": "sdfsdfsd"
    },
    {
      "name": "manishsdfs",
      "password": "sdfsdfsd"
    },
    {
      "name": "final ",
      "password": "one"
    },
    {
      "name": "manish singh tomar",
      "password": "Hashing11"
    },
    {
      "name": "tutedude",
      "password": "devops"
    }
  ],
  "hello": "world"
}
```