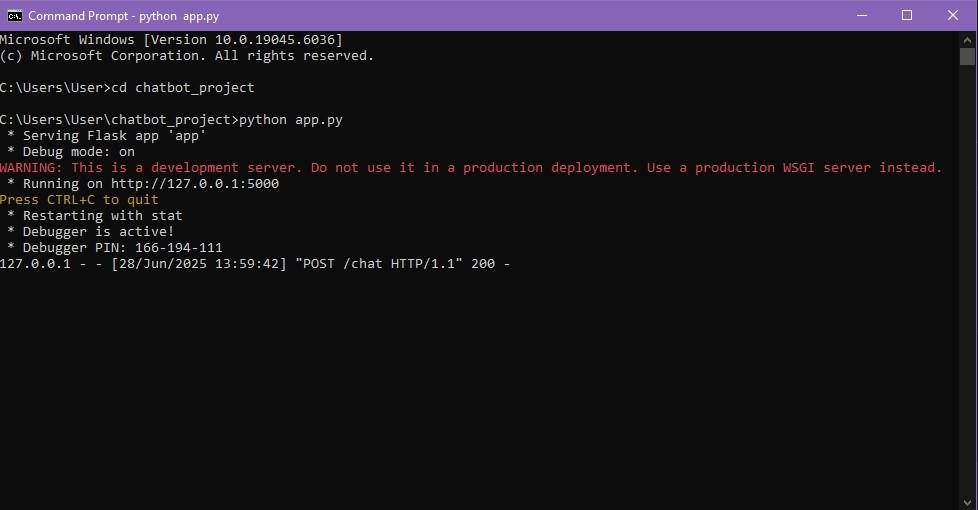
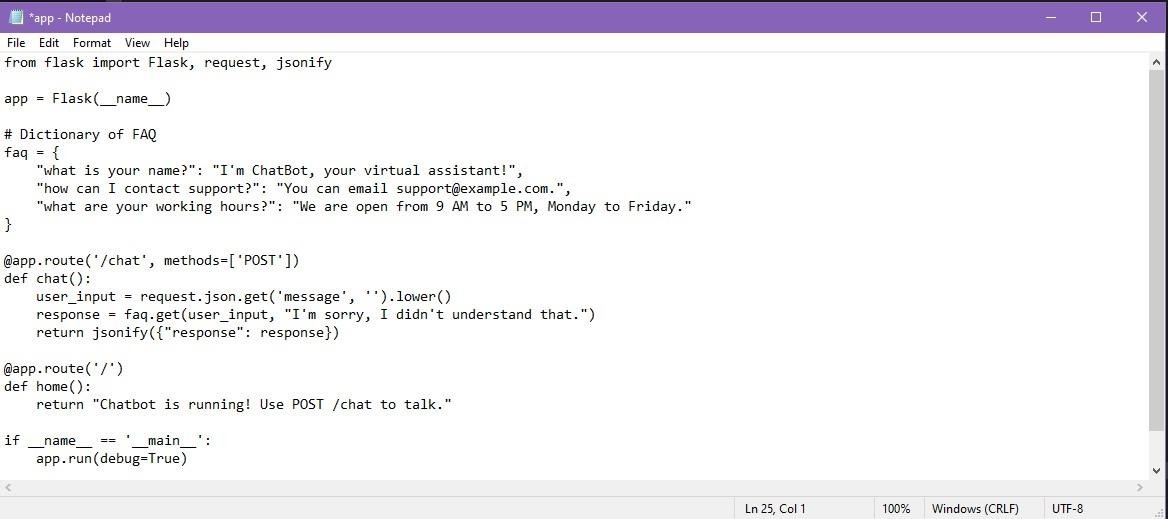
CODE ALPHA INTERNSHIP

Artificial Intelligence Tasks:

# Task #1 Build a Simple Chatbot

1. Installed Python and Flask using pip install flask.
2. Created a project folder chatbot\_project and a Python file app.py.
3. Created a Python file app.py using Notepad by running notepad app.py.
4. Wrote the chatbot code in app.py (shown below).



**Code Used:**

1. from flask import Flask, request, jsonify 2.

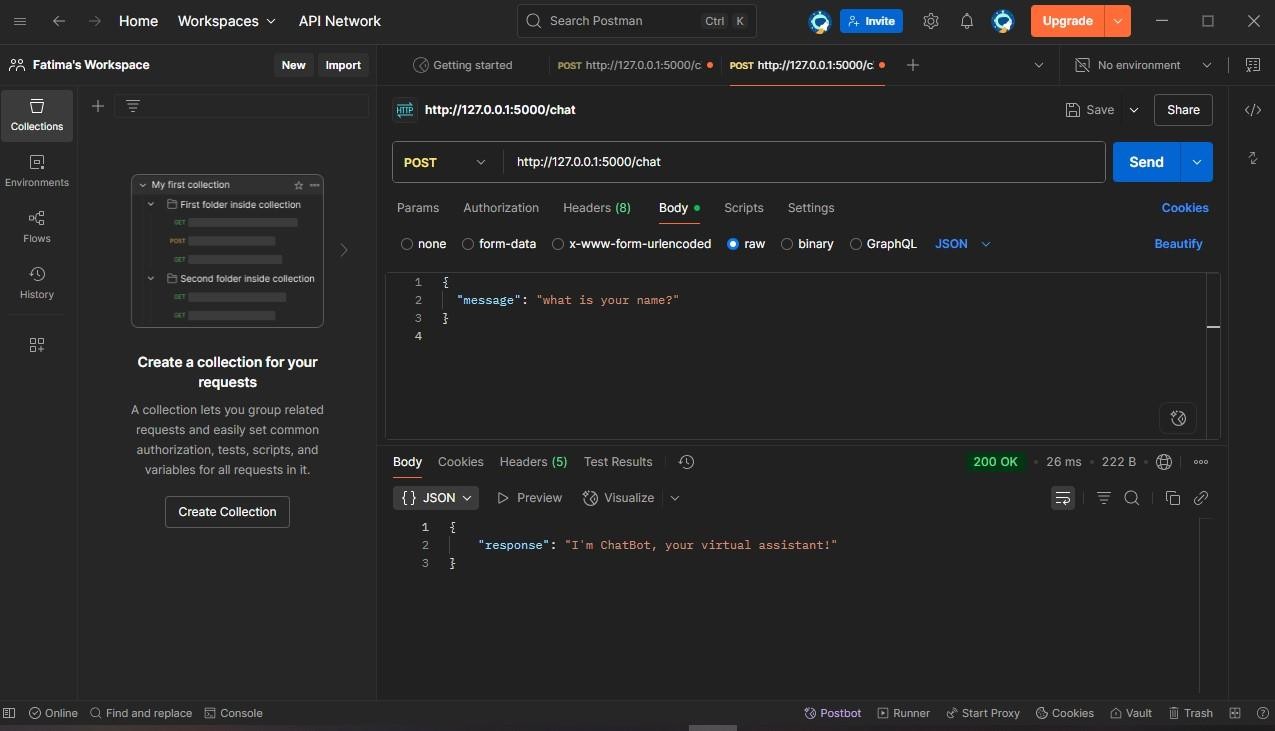
3. app = Flask( name ) 4.

1. faq = {
2. "what is your name?": "I'm ChatBot, your virtual assistant!",
3. "how can I contact support?": "You can email [support@example.com."](mailto:support@example.com),
4. "what are your working hours?": "We are open from 9 AM to 5 PM, Monday to Friday."

9. }

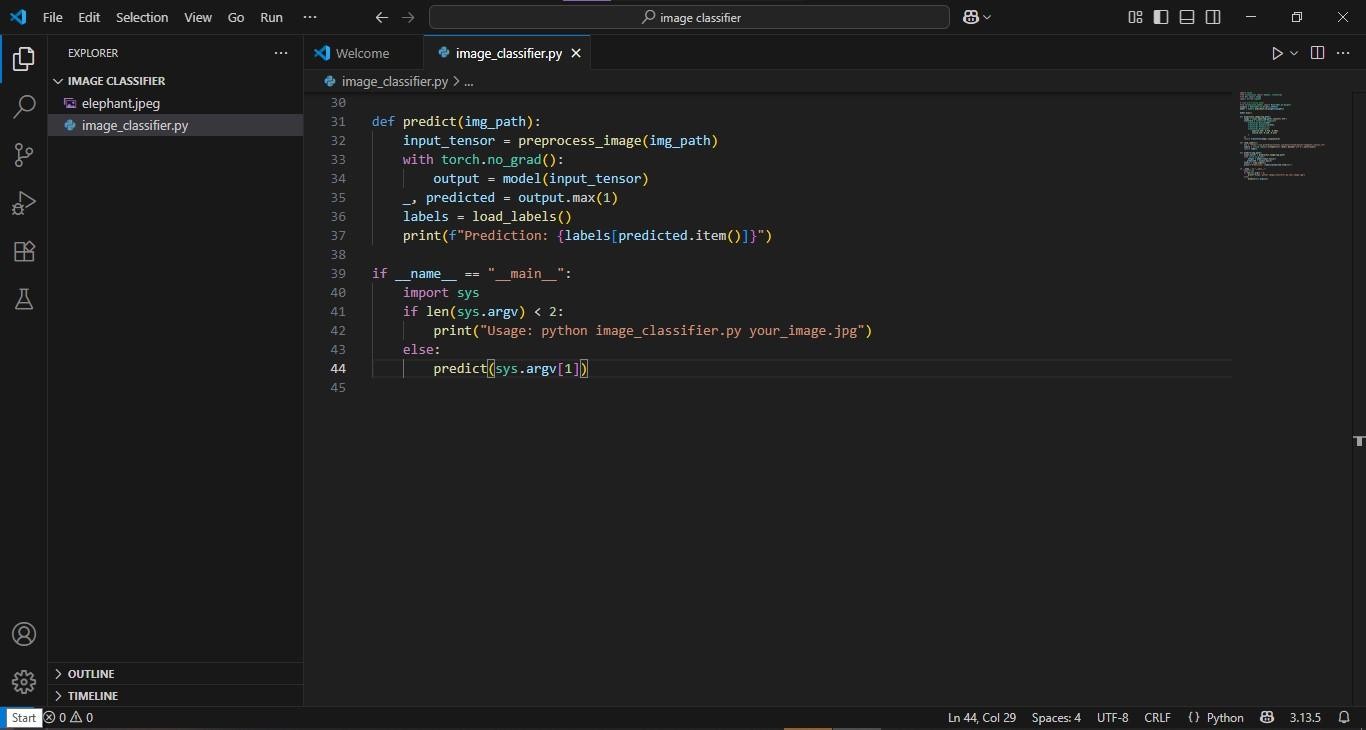
10.

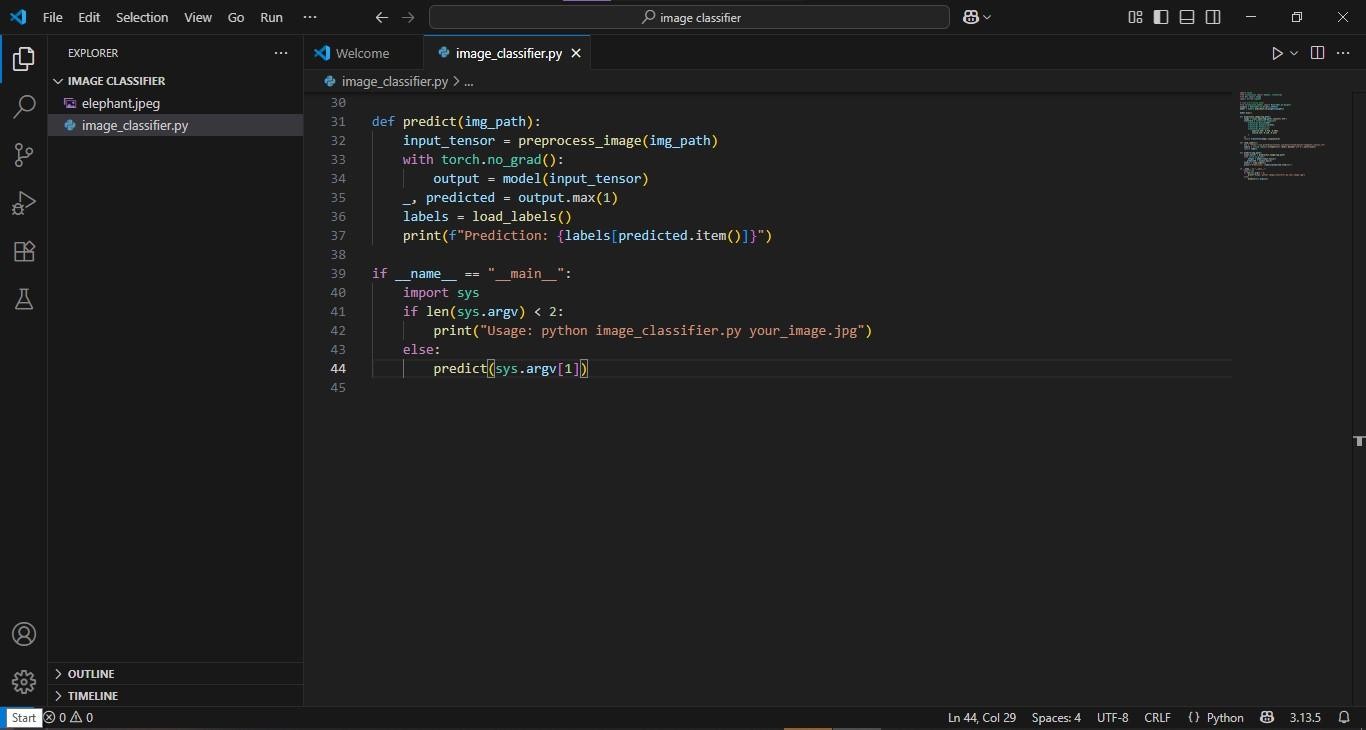
1. @app.route('/chat', methods=['POST'])
2. def chat():
3. user\_input = request.json.get('message', '').lower()
4. response = faq.get(user\_input, "I'm sorry, I didn't understand that.")
5. return jsonify({"response": response}) 16.
6. @app.route('/')
7. def home():
8. return "Chatbot is running! Use POST /chat to talk." 20.
9. if name == ' main ':
10. app.run(debug=True)
11. Ran the chatbot server using python app.py.
12. Accessed the home route <http://127.0.0.1:5000/> to confirm the server is running.
13. Set up a Flask web server with / route (GET) and /chat route (POST) to accept user input.
14. Tested the chatbot using Postman to send POST requests with JSON input.
15. Received prediction in terminal



# Task #2 AI for Image Classification

I used Python, PyTorch + TorchVision on VS Code to do this task.

1. Installed Required Libraries (torch + torchvision + Pillow)
2. Created Image Classification Script in VS Code



Code Used:

1. import torch
2. from torchvision import models, transforms
3. from PIL import Image
4. import urllib.request
5. # Load pre-trained model
6. from torchvision.models import MobileNet\_V2\_Weights
7. weights = MobileNet\_V2\_Weights.DEFAULT
8. model = models.mobilenet\_v2(weights=weights) 10.

11. model.eval()

12.

1. def preprocess\_image(img\_path):
2. image = Image.open(img\_path).convert('RGB')
3. transform = transforms.Compose([
4. transforms.Resize(256),
5. transforms.CenterCrop(224),
6. transforms.ToTensor(),
7. transforms.Normalize(
8. mean=[0.485, 0.456, 0.406],
9. std=[0.229, 0.224, 0.225]

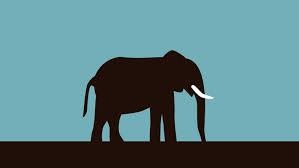
22. )

23. ])

24. return transform(image).unsqueeze(0)

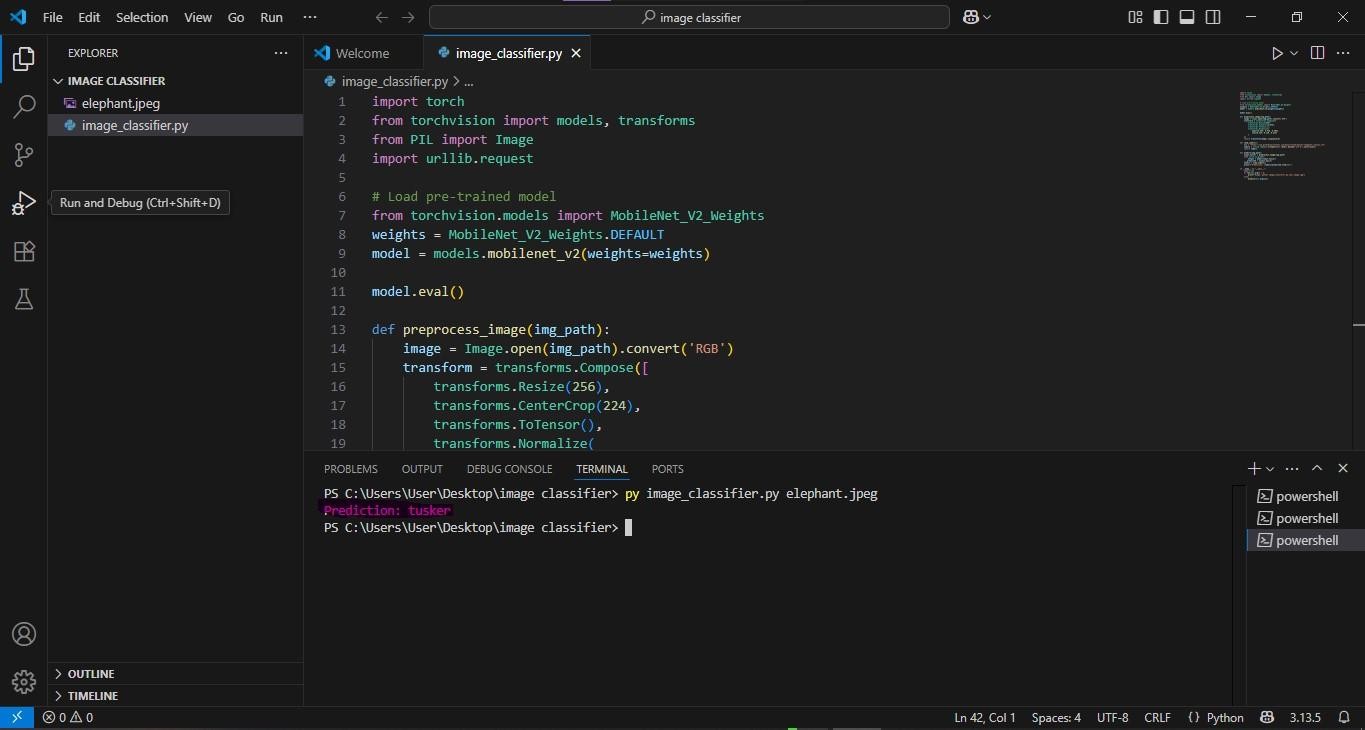
25.

1. def load\_labels():
2. url = "https://raw.githubusercontent.com/pytorch/hub/master/imagenet\_classes.txt"
3. labels = urllib.request.urlopen(url).read().decode('utf-8').splitlines()
4. return labels
5. def predict(img\_path):
6. input\_tensor = preprocess\_image(img\_path)
7. with torch.no\_grad():
8. output = model(input\_tensor)
9. \_, predicted = output.max(1)
10. labels = load\_labels()
11. print(f"Prediction: {labels[predicted.item()]}")
12. if name == " main ":
13. import sys
14. if len(sys.argv) < 2:
15. print("Usage: python image\_classifier.py your\_image.jpg")
16. else:
17. predict(sys.argv[1])
18. Added Image (elephant.jpeg) for Classification in the same project folder



1. Ran the Model and Obtained Prediction with the command

1. py image\_classifier.py cat.jpg

The model processed the image and printed a prediction in the terminal.

# Task #3 Text Summarization

I used Python, spaCy library, Counter on Visual Code.

1. Install Required Libraries (Spacy)
2. pip install spacy
3. python -m spacy download en\_core\_web\_sm
4. Created Python File (text\_summarizer.py) and wrote this code

Proper regulations and ethical practices are necessary to ensure AI benefits society as

"""

print("🔹 Summary:\n", summarize\_text(long\_text))

However, there are concerns about job displacement, bias in algorithms, and data

26.

privacy. 27.

a whole. 28.

29.

30.

than humans.

By analyzing vast amounts of data, AI can uncover patterns and make decisions faster

25.

finance, transportation, and more.

AI is becoming increasingly important in today’s world, with applications in healthcare,

24.

are programmed to think and learn like humans.

long\_text = """

Artificial Intelligence (AI) is the simulation of human intelligence in machines that

22.

23.

1. # Test the summarizer
2. if name == " main ":

summary = " ".join([sent.text for sent in top\_sentences])

return summary

17.

18.

19.

reverse=True)[:max\_sentences]

top\_sentences = sorted(sentence\_scores, key=sentence\_scores.get,

doc = nlp(text.lower())

words = [token.text for token in doc if token.is\_alpha and not token.is\_stop] word\_freq = Counter(words)

sentence\_scores = {} for sent in doc.sents:

score = sum(word\_freq[token.text] for token in sent if token.text in word\_freq) sentence\_scores[sent] = score

8.

9.

10.

11.

12.

13.

14.

15.

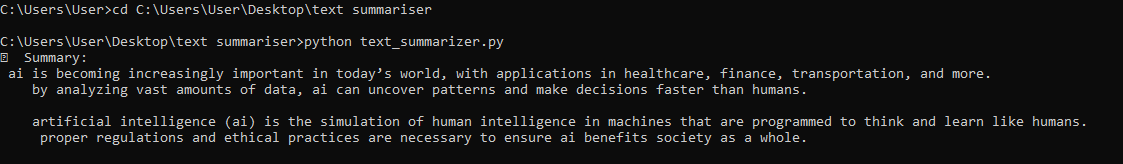
16.

1. import spacy
2. from collections import Counter 3.

4. # Load English NLP model

5. nlp = spacy.load("en\_core\_web\_sm") 6.

7. def summarize\_text(text, max\_sentences=3):

1. Ran the Code in Command Prompt