SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE		DEPARTMENT OF COMPUTER SCIENCE ENGINEERING	
Program Name: B. Tech		**Assignment Type: Lab**	**AcademicYear:**2025-2026
Course Coordinator Name		Venkataramana Veeramsetty	
Instructor(s)Name		1. Dr. Mohammed Ali Shaik 2. Dr. T Sampath Kumar 3. Mr. S Naresh Kumar 4. Dr. V. Rajesh 5. Dr. Brij Kishore 6. Dr Pramoda Patro 7. Dr. Venkataramana 8. Dr. Ravi Chander 9. Dr. Jagjeeth Singh	
CourseCode	24CS002PC215	**CourseTitle**	AI Assisted Coding
Year/Sem	II/I	**Regulation**	R24
Date and Day of Assignment	06-08-2025	**Time(s)**	
Duration	2 Hours	**Applicable to Batches**	24CSBTB01 To 24CSBTB39
AssignmentNumber:2.5(Present assignment number)/**24**(Total number of assignments)			

Q.No.	Question	*ExpectedTime to complete*
1	**Lab 2: Additional AI Coding Tools and API Configurations** **Objective:** To build a simple AI weather chatbot that fetches real-time weather data using the OpenWeatherMap API and responds to natural language queries using basic NLP techniques. Suppose that you are working as a junior developer in an AI services startup. Your team is building a smart assistant for travelers to get weather updates by simply asking questions like "What's the weather in Mumbai?" or "Is it raining in New York?" Tasks to be completed are as below 1. API Configuration: ● Register for a free OpenWeatherMap API key. ● Read the documentation and test API calls using tools like Postman or Python requests. ● CODE :-	01.08.2025 EOD

```python
import requests

# Replace this with your actual API key
api_key = "your_api_key_here"

# Get city name from user
city = input("Enter city name: ")

# Base URL
base_url = "http://api.openweathermap.org/data/2.5/weather"

# Parameters for the API call
params = {
    "q": city,
    "appid": api_key,
    "units": "metric"  # Use "imperial" for Fahrenheit
}

# Make the API call
response = requests.get(base_url, params=params)

# Check response status
if response.status_code == 200:
    data = response.json()
    temperature = data["main"]["temp"]
    description = data["weather"][0]["description"]
    humidity = data["main"]["humidity"]

    print(f"\nWeather in {city}:")
    print(f"Temperature: {temperature}°C")
    print(f"Description: {description}")
    print(f"Humidity: {humidity}%")
```

- OUTPUT :-

```
PS C:\AI CODEING>  & 'c:\Users\kbhuv\AppData\Local\Prog
 '54836' '--' 'c:\AI CODEING\lab ass 1.1.py'
Enter city name: ongole
```

2. NLP Integration:

- Use spaCy or NLTK to extract city names from user input.
- Convert natural language queries to appropriate API parameters.
- CODE:-

```python
import requests

# Replace with your actual API key from OpenWeatherMap
api_key = "your_api_key_here"
base_url = "http://api.openweathermap.org/data/2.5/weather"

# Function to extract city name from simple input
def extract_city(text):
    # Looks for the word 'in' and assumes the next word is a city
    if " in " in text.lower():
        parts = text.lower().split(" in ")
        city = parts[-1].strip().title()  # Title case for city name
        return city
    return None

# Function to get weather data from OpenWeatherMap
def get_weather(city):
    params = {
        "q": city,
        "appid": api_key,
        "units": "metric"
    }

    response = requests.get(base_url, params=params)
```

-
```python
    if response.status_code == 200:
    if response.status_code == 200:
        data = response.json()
        temp = data["main"]["temp"]
        desc = data["weather"][0]["description"]
        humidity = data["main"]["humidity"]

        return (
            f"🌍 Weather in {city}:\n"
            f"🌡️ Temperature: {temp}°C\n"
            f"🌧️ Condition: {desc}\n"
            f"💧 Humidity: {humidity}%"
        )
    elif response.status_code == 404:
        return "❌ City not found. Please try again."
    else:
        return "❌ Error fetching weather data."

# === Chatbot Loop ===
print("🤖 Hi! I'm WeatherBot. Ask me about the weather in any city.")
print("Type 'exit' to quit.\n")

while True:
    user_input = input("You: ")
```

-
```python
  if user_input.lower() == "exit":
      print("Bot: Goodbye! Stay safe. 🌞")
      break

  elif "weather" in user_input.lower():
      city = extract_city(user_input)

      if city:
          print("Bot: Checking the weather...")
          result = get_weather(city)
          print(f"Bot: {result}")
      else:
          print("Bot: Please include the city like: 'What's the weather in Mumbai?'")
  else:
      print("Bot: I can help you with weather info. Try asking something like: 'What's the weather in Paris?'")
```

-
- OUTPUT :-
-
```
d\libs\debugpy\launcher' '53546' '--' 'c:\AI CODEING\lab ass 1.1.py'
🤖 Hi! I'm WeatherBot. Ask me about the weather in any city.
Type 'exit' to quit.

You: hi how r u
Bot: I can help you with weather info. Try asking something like: 'What's the weather in Paris?'
You: are you marred
Bot: I can help you with weather info. Try asking something like: 'What's the weather in Paris?'
You: 'What's the weather in Paris?'
Bot: Checking the weather...
Bot: ❌ Error fetching weather data.
You:
```

3. Chatbot Design:

- Develop a basic chatbot using Python with user input and response cycles.
- Integrate it with OpenWeatherMap API to display temperature, condition, and humidity.
- CODE :-

```python
import requests

# === Your OpenWeatherMap API key ===
api_key = "your_api_key_here"
base_url = "http://api.openweathermap.org/data/2.5/weather"

def get_weather(city):
    params = {
        "q": city,
        "appid": api_key,
        "units": "metric"
    }

    response = requests.get(base_url, params=params)

    if response.status_code == 200:
        data = response.json()
        temperature = data["main"]["temp"]
        description = data["weather"][0]["description"]
        humidity = data["main"]["humidity"]

        return (
            f"Weather in {city}:\n"
            f"🌡️ Temperature: {temperature}°C\n"
            f"🌧️ Condition: {description}\n"
            f"💧 Humidity: {humidity}%"
        )
    elif response.status_code == 404:
        return "❌ City not found. Please try again."
    else:
        return "❌ Unable to fetch weather data at the moment."

# === Chatbot Loop ===
print("🤖 Hello! I'm WeatherBot. Ask me about the weather in any city.")
print("Type 'exit' to end the conversation.\n")
```

-

```python
while True:
    user_input = input("You: ")

    if user_input.lower() == "exit":
        print("Bot: Goodbye! Stay safe. 🌞")
        break

    elif "weather" in user_input.lower():
        # Simple city extraction (assumes last word is city)
        words = user_input.split()
        city = words[-1]

        print("Bot: Let me check the weather for you...")
        result = get_weather(city)
        print(f"Bot: {result}")
    else:
        print("Bot: I can help you with weather info. Ask me like: 'What's the weather in Delhi?'")
```

- OUTPUT :-

```
PS C:\AI CODEING> & 'c:\Users\kbhuv\AppData\Local\Programs\Python\Python313\python.e
debugpy\launcher' '53478' '--' 'c:\AI CODEING\lab ass 1.1.py'
🤖 Hello! I'm WeatherBot. Ask me about the weather in any city.
Type 'exit' to end the conversation.

You: hi I love u
Bot: I can help you with weather info. Ask me like: 'What's the weather in Delhi?'
You:
```

Requirements:

- VS Code with Github Copilot or Cursor API and/or Google Colab with Gemini

	Deliverables: ● A working Python script or notebook. ● A short video demo or screenshot of chatbot interaction. ● A markdown file with steps followed and challenges faced. .	