

School of Computer Science and Artificial Intelligence

Lab Assignment # 4.5

Program	: B. Tech (CSE)
Specialization	: -
Course Title	: AI Assisted Coding
Course Code	: 23CS002PC304
Semester	: II
Academic Session	: 2025-2026
Name of Student	: I.Sathwik Rajeshwara Chary
Enrollment No.	: 2403A51L03
Batch No.	: 51
Date	: 16/01/26

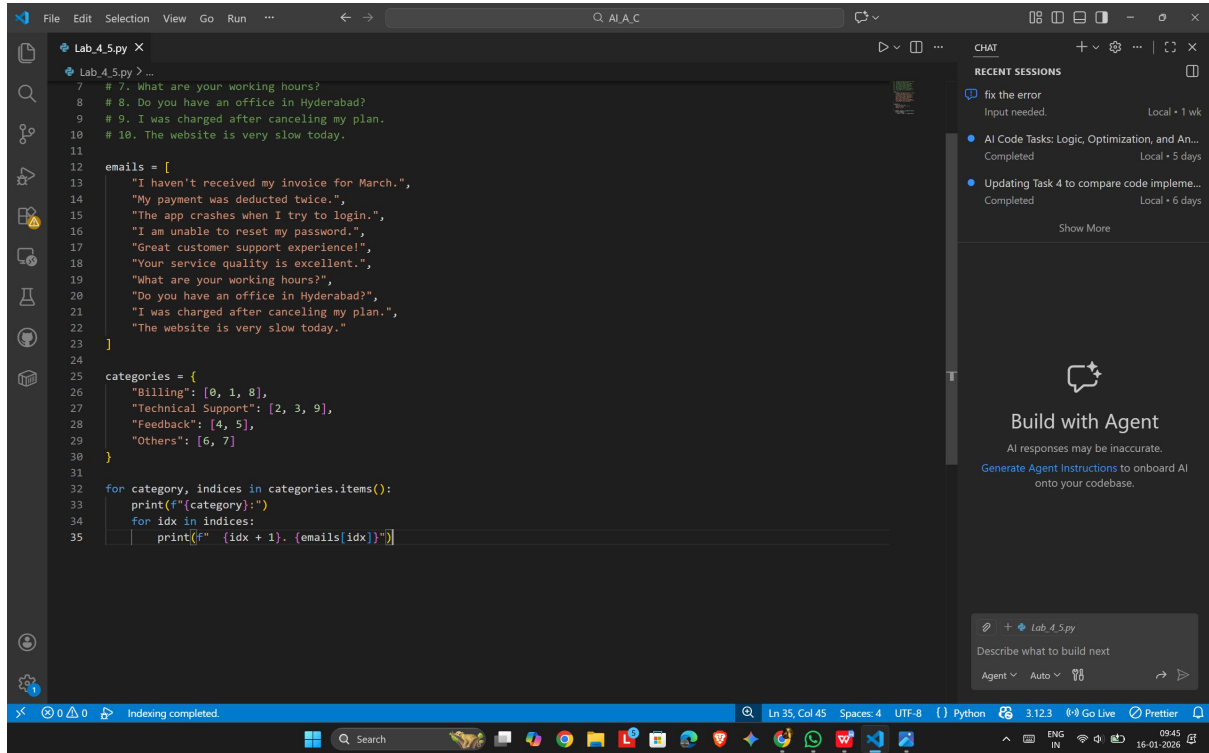
Submission Starts here**Screenshots:****1.****TASK - A:**

- 1. I haven't received my invoice for March.**
- 2. My payment was deducted twice.**
- 3. The app crashes when I try to login.**
- 4. I am unable to reset my password.**
- 5. Great customer support experience!**
- 6. Your service quality is excellent.**
- 7. What are your working hours?**
- 8. Do you have an office in Hyderabad?**
- 9. I was charged after canceling my plan.**
- 10. The website is very slow today.**

TASK - B:**Prompt:**

**Classify the above emails into one of these categories:
Billing, Technical Support, Feedback, Others.**

Code:

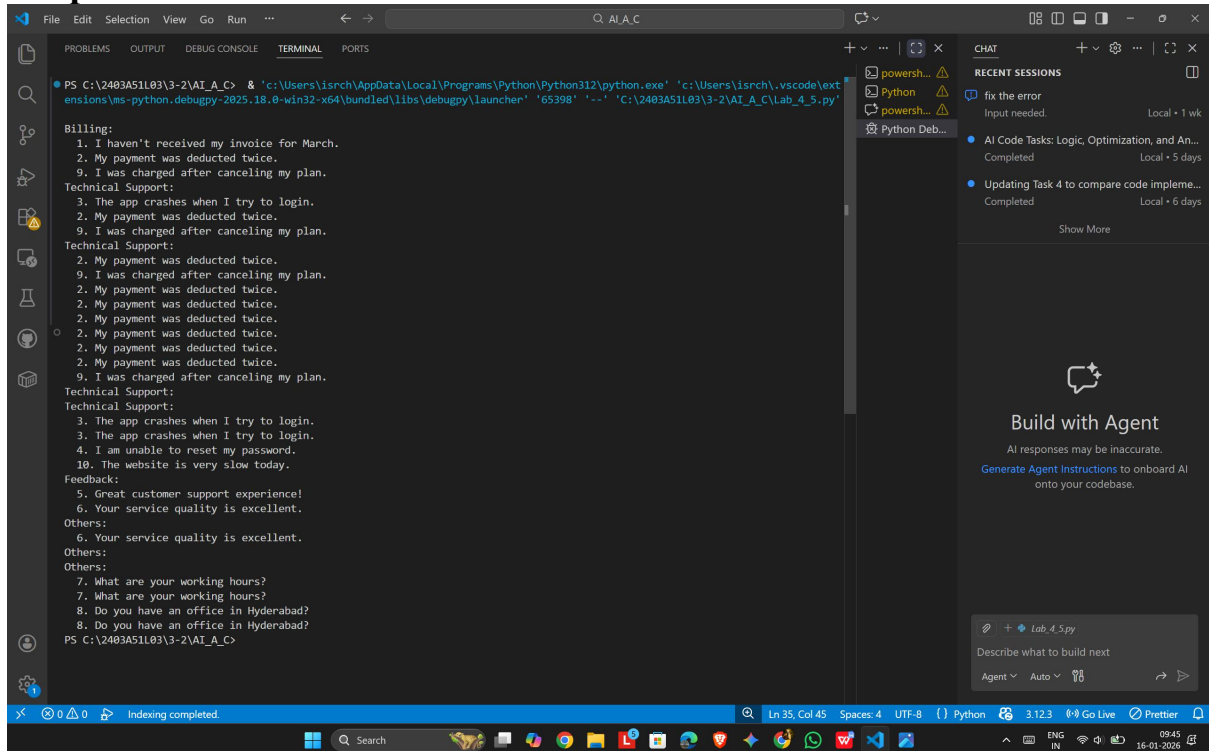


```

7  # 7. What are your working hours?
8  # 8. Do you have an office in Hyderabad?
9  # 9. I was charged after canceling my plan.
10 # 10. The website is very slow today.
11
12 emails = [
13     "I haven't received my invoice for March.",
14     "My payment was deducted twice.",
15     "The app crashes when I try to login.",
16     "I am unable to reset my password.",
17     "Great customer support experience!",
18     "Your service quality is excellent.",
19     "What are your working hours?",
20     "Do you have an office in Hyderabad?",
21     "I was charged after canceling my plan.",
22     "The website is very slow today."
23 ]
24
25 categories = {
26     "Billing": [0, 1, 8],
27     "Technical Support": [2, 3, 9],
28     "Feedback": [4, 5],
29     "Others": [6, 7]
30 }
31
32 for category, indices in categories.items():
33     print(f'{category}:')
34     for idx in indices:
35         print(f'    {idx + 1}. {emails[idx]}')

```

Output:



```

PS C:\2483A51L03\3-2\AI_A_C> & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '65398' '-.' 'C:\2483A51L03\3-2\AI_A_C\Lab_4_5.py'

Billing:
1. I haven't received my invoice for March.
2. My payment was deducted twice.
9. I was charged after canceling my plan.
Technical Support:
3. The app crashes when I try to login.
2. My payment was deducted twice.
9. I was charged after canceling my plan.
Technical Support:
2. My payment was deducted twice.
9. I was charged after canceling my plan.
2. My payment was deducted twice.
2. My payment was deducted twice.
2. My payment was deducted twice.
2. My payment was deducted twice.
2. My payment was deducted twice.
9. I was charged after canceling my plan.
Technical Support:
3. The app crashes when I try to login.
3. The app crashes when I try to login.
4. I am unable to reset my password.
10. The website is very slow today.
Feedback:
5. Great customer support experience!
6. Your service quality is excellent.
Others:
6. Your service quality is excellent.
Others:
7. What are your working hours?
7. What are your working hours?
8. Do you have an office in Hyderabad?
8. Do you have an office in Hyderabad?
PS C:\2483A51L03\3-2\AI_A_C>

```

TASK - C:

Prompt:

Example:

Email: "My payment was deducted twice."

Category: Billing

Now classify this email:

Email: "The app crashes when I try to login."

And print it

Code:

The screenshot shows a Visual Studio Code editor window with a file named 'Lab_4_5.py'. The code in the editor is as follows:

```
1 # 1. The app crashes when I try to login.
2 # 2. I am unable to reset my password.
3 # 3. Great customer support experience!
4 # 4. I am unable to reset my password.
5 # 5. Great customer support experience!
6 # 6. Your service quality is excellent.
7 # 7. What are your working hours?
8 # 8. Do you have an office in Hyderabad?
9 # 9. I was charged after canceling my plan.
10 # 10. The website is very slow today.
11
12 email = "The app crashes when I try to login."
13 category = "Technical Issue"
14 print(f"Email: '{email}'\nCategory: {category}")
```

The bottom panel of the editor shows the 'TERMINAL' output, which displays the execution of the script. The output is:

```
z. My payment was deducted twice.
9. I was charged after canceling my plan.
Technical Support: ...
PS C:\2403A51103\3-2\AI_A_C> cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '51736' '--' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
PS C:\2403A51103\3-2\AI_A_C> cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '51767' '--' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Email: "The app crashes when I try to login."
Category: Technical Issue
PS C:\2403A51103\3-2\AI_A_C>
```

The right sidebar shows the 'CHAT' panel with a 'RECENT SESSIONS' list. The list includes:

- fix the error (Local • 1 wk)
- AI Code Tasks: Logic, Optimization, and An... (Completed, Local • 5 days)
- Updating Task 4 to compare code impleme... (Completed, Local • 6 days)

Below the list is a 'Show More' button. At the bottom of the sidebar is a 'Build with Agent' section with a chat input field and buttons for 'Agent', 'Auto', and 'Go Live'.

Output:

The screenshot shows a VS Code editor with a file named `Lab_4_5.py`. The script contains a list of customer feedback items and a function to print them in a structured format. The terminal output shows the script being executed, and the feedback items are printed as JSON objects.

```

1 # 3. The app crashes when I try to login.
2 # 4. I am unable to reset my password.
3 # 5. Great customer support experience!
4 # 6. Your service quality is excellent.
5 # 7. What are your working hours?
6 # 8. Do you have an office in Hyderabad?
7 # 9. I was charged after canceling my plan.
8 # 10. The website is very slow today.
9
10 email = "The app crashes when I try to login."
11 category = "Technical Issue"
12 print(f'Email: "{email}"\nCategory: {category}')

```

The terminal output shows the following commands and results:

```

PS C:\2403A51103\3-2\AI_A_C> cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '51736' '--' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
PS C:\2403A51103\3-2\AI_A_C> cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '51767' '--' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Email: "The app crashes when I try to login."
Category: Technical Issue
PS C:\2403A51103\3-2\AI_A_C>

```

TASK - D:

Prompt:

Example 1:

Email: "My payment was deducted twice."

Category: Billing

Example 2:

Email: "I am unable to reset my password."

Category: Technical Support

Example 3:

Email: "Great customer support experience!"

Category: Feedback

Now classify:

Email: "I was charged after canceling my plan."

Code:

```

11
12 # Classification function
13 def classify_email(email):
14     email_lower = email.lower()
15
16     if any(word in email_lower for word in ["payment", "charged", "deducted", "billing", "invoice", "refund"]):
17         return "Billing"
18     elif any(word in email_lower for word in ["login", "password", "crash", "error", "bug", "technical"]):
19         return "Technical Support"
20     elif any(word in email_lower for word in ["great", "excellent", "good", "appreciate", "thank"]):
21         return "Feedback"
22     elif any(word in email_lower for word in ["hours", "office", "location", "address", "contact"]):
23         return "General Inquiry"
24     else:
25         return "Other"
26
27 # Test with the provided example
28 email = "I was charged after canceling my plan."
29 print(f"Email: \"{email}\"")
30 print(f"Category: {classify_email(email)}")

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

PS C:\2403A51103\3-2\AI_A_C> c:; cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '62705' '-.' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Email: "I was charged after canceling my plan."
Category: Billing
PS C:\2403A51103\3-2\AI_A_C>

```

Build with Agent

AI responses may be inaccurate.
Generate Agent Instructions to onboard AI onto your codebase.

Describe what to build next

Agent Auto

Output:

```

11
12 # Classification function
13 def classify_email(email):
14     email_lower = email.lower()
15
16     if any(word in email_lower for word in ["payment", "charged", "deducted", "billing", "invoice", "refund"]):
17         return "Billing"
18     elif any(word in email_lower for word in ["login", "password", "crash", "error", "bug", "technical"]):
19         return "Technical Support"
20     elif any(word in email_lower for word in ["great", "excellent", "good", "appreciate", "thank"]):
21         return "Feedback"
22     elif any(word in email_lower for word in ["hours", "office", "location", "address", "contact"]):
23         return "General Inquiry"
24     else:
25         return "Other"
26
27 # Test with the provided example
28 email = "I was charged after canceling my plan."
29 print(f"Email: \"{email}\"")
30 print(f"Category: {classify_email(email)}")

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

PS C:\2403A51103\3-2\AI_A_C> c:; cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '62705' '-.' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Email: "I was charged after canceling my plan."
Category: Billing
PS C:\2403A51103\3-2\AI_A_C>

```

Build with Agent

AI responses may be inaccurate.
Generate Agent Instructions to onboard AI onto your codebase.

Describe what to build next

Agent Auto

2.

TASK - A:

Categories

Flight Booking

Hotel Booking

Cancellation

General Travel Info

TASK - B:

Prompt:

Classify the travel query into Flight Booking, Hotel Booking, Cancellation, or General Travel Info.

Query: "Cancel my flight ticket."

Code:

```
def classify_query(query):
    query_lower = query.lower()

    if "cancel" in query_lower:
        return "Cancellation"
    elif "flight" in query_lower and "book" in query_lower:
        return "Flight Booking"
    elif "hotel" in query_lower and "book" in query_lower:
        return "Hotel Booking"
    else:
        return "General Travel Info"

# Test with the given query
query = "Cancel my flight ticket."
print(classify_query(query))
```

Terminal Output:

```
PS C:\2403A51103\3-2\AI_A_C> c:: cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '62705' '-.' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Email: "I was charged after canceling my plan."
Category: Billing

PS C:\2403A51103\3-2\AI_A_C> c:: cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '55376' '-.' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Cancellation

PS C:\2403A51103\3-2\AI_A_C>
```

Output:

The screenshot shows a VS Code editor with a Python file named `Lab_4_5.py`. The script defines a function `classify_query` that takes a query string and returns a category based on keywords. The terminal shows the execution of the script, which outputs "Cancellation" for the query "Cancel my flight ticket."

```

10
11 def classify_query(query):
12     query_lower = query.lower()
13
14     if "cancel" in query_lower:
15         return "Cancellation"
16     elif "flight" in query_lower and "book" in query_lower:
17         return "Flight Booking"
18     elif "hotel" in query_lower and "book" in query_lower:
19         return "Hotel Booking"
20     else:
21         return "General Travel Info"
22
23 # Test with the given query
24 query = "Cancel my flight ticket."
25 print(classify_query(query))

```

Terminal Output:

```

PS C:\2403A51L03\3-2\AI_A_C> cd 'c:\2403A51L03\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '62705' '--' 'C:\2403A51L03\3-2\AI_A_C\Lab_4_5.py'
Email: "I was charged after canceling my plan."
Category: Billing
PS C:\2403A51L03\3-2\AI_A_C> cd 'c:\2403A51L03\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '55376' '--' 'C:\2403A51L03\3-2\AI_A_C\Lab_4_5.py'
Cancellation
PS C:\2403A51L03\3-2\AI_A_C>

```

TASK - C:

Prompt:

Example:

Query: "Book a flight from Delhi to Mumbai"

Category: Flight Booking

Now classify:

Query: "Need a hotel in Goa."

And print it

Code:

```

4
5 # Hotel Booking
6
7 # Cancellation
8
9 # General Travel Info
10
11 query = "Need a hotel in Goa."
12 category = "Hotel Booking"
13 print(f"Query: \"{query}\"")
14 print(f"Category: {category}")

```

```

PS C:\2403A51103\3-2\AI_A_C> cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '62705' '--' 'c:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
PS C:\2403A51103\3-2\AI_A_C> cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '55376' '--' 'c:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Cancellation
PS C:\2403A51103\3-2\AI_A_C> cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '60722' '--' 'c:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Query: "Need a hotel in Goa."
Category: Hotel Booking
PS C:\2403A51103\3-2\AI_A_C>

```

Output:

```

4
5 # Hotel Booking
6
7 # Cancellation
8
9 # General Travel Info
10
11 query = "Need a hotel in Goa."
12 category = "Hotel Booking"
13 print(f"Query: \"{query}\"")
14 print(f"Category: {category}")

```

```

PS C:\2403A51103\3-2\AI_A_C> cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '62705' '--' 'c:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
PS C:\2403A51103\3-2\AI_A_C> cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '55376' '--' 'c:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Cancellation
PS C:\2403A51103\3-2\AI_A_C> cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '60722' '--' 'c:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Query: "Need a hotel in Goa."
Category: Hotel Booking
PS C:\2403A51103\3-2\AI_A_C>

```

TASK - D:

Prompt:

Query: "Book a flight from Delhi to Mumbai" → Flight Booking

Query: "Need a hotel in Goa" → Hotel Booking

Query: "Cancel my booking" → Cancellation

Now classify:

Query: "Best time to visit Kashmir?"

And print

Code:

```

1 # hotel booking
2
3 # Cancellation
4
5 # General Travel Info
6
7 def classify_query(query):
8     if "flight" in query.lower() or "book" in query.lower() and "flight" in query.lower():
9         return "Flight Booking"
10    elif "hotel" in query.lower() or "accommodation" in query.lower():
11        return "Hotel Booking"
12    elif "cancel" in query.lower():
13        return "Cancellation"
14    else:
15        return "General Travel Info"
16
17 query = "Best time to visit Kashmir?"
18 print(classify_query(query))

```

```

PS C:\2403A51103\3-2\AI_A_C> c:\cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '62705' '--' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
PS C:\2403A51103\3-2\AI_A_C> c:\cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '55376' '--' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Cancellation
PS C:\2403A51103\3-2\AI_A_C> c:\cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '68722' '--' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Query: "Need a hotel in Goa."
Category: Hotel Booking
PS C:\2403A51103\3-2\AI_A_C>

```

Output:

```

1 # hotel booking
2
3 # Cancellation
4
5 # General Travel Info
6
7 def classify_query(query):
8     if "flight" in query.lower() or "book" in query.lower() and "flight" in query.lower():
9         return "Flight Booking"
10    elif "hotel" in query.lower() or "accommodation" in query.lower():
11        return "Hotel Booking"
12    elif "cancel" in query.lower():
13        return "Cancellation"
14    else:
15        return "General Travel Info"
16
17 query = "Best time to visit Kashmir?"
18 print(classify_query(query))

```

```

PS C:\2403A51103\3-2\AI_A_C> c:\cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '62705' '--' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
PS C:\2403A51103\3-2\AI_A_C> c:\cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '55376' '--' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Cancellation
PS C:\2403A51103\3-2\AI_A_C> c:\cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '68722' '--' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Query: "Need a hotel in Goa."
Category: Hotel Booking
PS C:\2403A51103\3-2\AI_A_C>

```

3.

TASK - A:

1. Why am I getting an indentation error in Python?
2. My loop executes but gives wrong output.
3. How can I make this code run faster?
4. What is recursion in programming?
5. Why does my program crash without any error?

TASK - B:

Prompt:

Classify the following programming question into one of these categories: Syntax Error, Logic Error, Optimization, Conceptual Question.

Question: "Why am I getting an indentation error in Python?"

Code:

```

1 # 1. Why am I getting an indentation error in Python?
2 # 2. My loop executes but gives wrong output.
3 # 3. How can I make this code run faster?
4 # 4. What is recursion in programming?
5 # 5. Why does my program crash without any error?
6
7 | Classification of the question
8 question = "Why am I getting an indentation error in Python?"
9 category = "Syntax Error"
10
11 # Output the classification
12 print(f"Question: {question}")
13 print(f"Category: {category}")

```

```

on.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '55376' '-' 'C
:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Cancellation
PS C:\2403A51103\3-2\AI_A_C> cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\pyth
on.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '60722' '-' 'C
:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Query: "Need a hotel in Goa."
Category: Hotel Booking
PS C:\2403A51103\3-2\AI_A_C> ^C
PS C:\2403A51103\3-2\AI_A_C> cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\pyth
on.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '55567' '-' 'C
:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Question: Why am I getting an indentation error in Python?
Category: Syntax Error
PS C:\2403A51103\3-2\AI_A_C>

```

CHAT

← CLASSIFYING A PYTHON INDENTATION...
 NEED A HOTEL IN GOA. CATEGORY: HOTEL BOOKING.
 Syntax Error, Logic Error, Optimization, Conceptual Question.
 Question: "Why am I getting an indentation error in Python?"

The question "Why am I getting an indentation error in Python?" falls into the category of **Syntax Error**. Indentation errors in Python are related to the incorrect use of whitespace, which is a syntax issue.

Explore and understand your code

Ask Auto

Output:

```

1 # 1. Why am I getting an indentation error in Python?
2 # 2. My loop executes but gives wrong output.
3 # 3. How can I make this code run faster?
4 # 4. What is recursion in programming?
5 # 5. Why does my program crash without any error?
6
7 # Classification of the question
8 question = "Why am I getting an indentation error in Python?"
9 category = "Syntax Error"
10
11 # Output the classification
12 print(f"Question: {question}")
13 print(f"Category: {category}")

```

```

on.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '55376' '--' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Cancellation
PS C:\2403A51103\3-2\AI_A_C> c:; cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '68722' '--' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Query: "Need a hotel in Goa."
Category: Hotel Booking
PS C:\2403A51103\3-2\AI_A_C> c:; cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\debugpy\launcher' '55567' '--' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Question: Why am I getting an indentation error in Python?
Category: Syntax Error
PS C:\2403A51103\3-2\AI_A_C>

```

CLASSIFYING A PYTHON INDENTATION...
question into one of these categories:
Syntax Error, Logic Error, Optimization, Conceptual Question.
Question: "Why am I getting an indentation error in Python?"

The question "Why am I getting an indentation error in Python?" falls into the category of **Syntax Error**. Indentation errors in Python are related to the incorrect use of whitespace, which is a syntax issue.

TASK - C:

Prompt:

Example:

Question: "What is recursion in programming?"

Category: Conceptual Question

Now classify:

Question: "My loop executes but gives wrong output."

Code:

```

1 # 1. Why am I getting an indentation error in Python?
2 # 2. My loop executes but gives wrong output.
3 # 3. How can I make this code run faster?
4 # 4. What is recursion in programming?
5 def classify_question(question):
6     if "recursion" in question:
7         return "Conceptual Question"
8     elif "loop" in question and "wrong output" in question:
9         return "Logic Error"
10    elif "crash" in question:
11        return "Runtime Error"
12    else:
13        return "General Question"
14
15 # Example usage
16 question = "My loop executes but gives wrong output."
17 category = classify_question(question)
18 print(f"Question: \"{question}\"\\nCategory: {category}")
19
20

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

.:2403A51103\3-2\AI_A_C\Lab_4_5.py'
Query: "Need a hotel in Goa."
Category: Hotel Booking
PS C:\2403A51103\3-2\AI_A_C> ^C
PS C:\2403A51103\3-2\AI_A_C> c; cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '55567' '--' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Question: Why am I getting an indentation error in Python?
Category: Syntax Error
PS C:\2403A51103\3-2\AI_A_C> c; cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '55605' '--' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Question: "My loop executes but gives wrong output."
Category: Logic Error
PS C:\2403A51103\3-2\AI_A_C>

```

Output:

```

1 # 1. Why am I getting an indentation error in Python?
2 # 2. My loop executes but gives wrong output.
3 # 3. How can I make this code run faster?
4 # 4. What is recursion in programming?
5 def classify_question(question):
6     if "recursion" in question:
7         return "Conceptual Question"
8     elif "loop" in question and "wrong output" in question:
9         return "Logic Error"
10    elif "crash" in question:
11        return "Runtime Error"
12    else:
13        return "General Question"
14
15 # Example usage
16 question = "My loop executes but gives wrong output."
17 category = classify_question(question)
18 print(f"Question: \"{question}\"\\nCategory: {category}")
19
20

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

.:2403A51103\3-2\AI_A_C\Lab_4_5.py'
Query: "Need a hotel in Goa."
Category: Hotel Booking
PS C:\2403A51103\3-2\AI_A_C> ^C
PS C:\2403A51103\3-2\AI_A_C> c; cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '55567' '--' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Question: Why am I getting an indentation error in Python?
Category: Syntax Error
PS C:\2403A51103\3-2\AI_A_C> c; cd 'c:\2403A51103\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '55605' '--' 'C:\2403A51103\3-2\AI_A_C\Lab_4_5.py'
Question: "My loop executes but gives wrong output."
Category: Logic Error
PS C:\2403A51103\3-2\AI_A_C>

```

TASK - D:

Prompt:

Examples:

Question: "Why am I getting an indentation error in Python?"

Category: Syntax Error

Question: "My loop executes but gives wrong output."

Category: Logic Error

Question: "How can I make this code run faster?"

Category: Optimization

Now classify:

Question: "What is recursion in programming?"

Code:

The screenshot shows a Google Colab notebook titled 'Untitled72.ipynb'. The code defines a function `classify_question` that takes a question string and returns a category based on keywords. The keywords and their corresponding categories are: 'indentation error', 'syntax error', 'unexpected eof' (Syntax Error); 'wrong output', 'incorrect result', 'bug' (Logic Error); 'faster', 'speed', 'optimize', 'performance' (Optimization); 'what is', 'explain', 'concept', 'recursion', 'object-oriented', 'data' (Programming Concept); and any other keywords (Unclassified). The code then classifies the question 'What is recursion in programming?' and prints the result.

```
def classify_question(question):
    question = question.lower()
    if any(keyword in question for keyword in ['indentation error', 'syntax error', 'unexpected eof']):
        return 'Syntax Error'
    elif any(keyword in question for keyword in ['wrong output', 'incorrect result', 'bug']):
        return 'Logic Error'
    elif any(keyword in question for keyword in ['faster', 'speed', 'optimize', 'performance']):
        return 'Optimization'
    elif any(keyword in question for keyword in ['what is', 'explain', 'concept', 'recursion', 'object-oriented', 'data']):
        return 'Programming Concept'
    else:
        return 'Unclassified'

# Classify the specific question from the prompt
question_to_classify = "What is recursion in programming?"
classified_category = classify_question(question_to_classify)

print("--- Your Question ---")
print(f"Question: \"{question_to_classify}\"")
print(f"Category: {classified_category}")

--- Your Question ---
Question: "What is recursion in programming?"
Category: Programming Concept
```

The Gemini chat interface on the right shows the user's prompt: "I want you to give code for it". The Gemini response is: "You're right! Here's some Python code that attempts to classify the questions based on keywords, and then applies it to your example. The code has successfully classified the question 'What is recursion in programming?' as a 'Programming Concept'. It also demonstrated how it classifies other types of questions based on keywords, such as 'Syntax Error', 'Logic Error', and 'Optimization'."

Output:

The screenshot shows a Google Colab notebook with a Python function `classify_question` and a Gemini chat interface. The function classifies questions based on keywords into categories like 'Syntax Error', 'Logic Error', 'Optimization', 'Performance', 'Programming Concept', or 'Unclassified'. The Gemini chat shows a user asking for code to classify questions, and the AI providing the code and explaining its logic.

```
def classify_question(question):
    question = question.lower()
    if any(keyword in question for keyword in ['indentation error', 'syntax error', 'unexpected eof']):
        return 'Syntax Error'
    elif any(keyword in question for keyword in ['wrong output', 'incorrect result', 'bug']):
        return 'Logic Error'
    elif any(keyword in question for keyword in ['faster', 'speed', 'optimize', 'performance']):
        return 'Optimization'
    elif any(keyword in question for keyword in ['what is', 'explain', 'concept', 'recursion', 'object-oriented', 'de']):
        return 'Programming Concept'
    else:
        return 'Unclassified'

# Classify the specific question from the prompt
question_to_classify = "What is recursion in programming?"
classified_category = classify_question(question_to_classify)

print("--- Your Question ---")
print(f"Question: '{question_to_classify}'")
print(f"Category: {classified_category}")

--- Your Question ---
Question: "What is recursion in programming?"
Category: Programming Concept
```

Gemini

i want you to give code for it

You're right! Here's some Python code that attempts to classify the questions based on keywords, and then applies it to your example.

The code has successfully classified the question "What is recursion in programming?" as a 'Programming Concept'. It also demonstrated how it classifies other types of questions based on keywords, such as 'Syntax Error', 'Logic Error', and 'Optimization'.

Explain how the 'classify_question' function works

Add a category for 'Data Analysis' to the classifier

Classify this new question: 'How do I fix a TypeError?'

What can I help you build?

Gemini 2.5 Flash

Gemini can make mistakes so double-check it and use code with caution. [Learn more](#)

4.

TASK - A:

1. Check out our new product launch!
2. The app keeps crashing, very annoying.
3. Amazing service, thank you!
4. When will the next update be released?
5. Your customer care never responds.

TASK - B:

Prompt:

Classify the following social media post into one of these categories: Promotion, Complaint, Appreciation, Inquiry.

Post: "The app keeps crashing, very annoying."

Code:

The screenshot shows a Jupyter Notebook titled 'Untitled72.ipynb' in a web browser. The code defines a function `classify_social_media_post(post)` that classifies social media posts based on keywords. The keywords are categorized into 'Promotion', 'Complaint', 'Appreciation', 'Inquiry', and 'Unclassified'. The code then classifies a specific post: "The app keeps crashing, very annoying." and prints the result: "Category: Complaint".

```
[1] def classify_social_media_post(post):
    post = post.lower()
    if any(keyword in post for keyword in ['new product', 'launch', 'deal', 'offer', 'sale', 'check out']):
        return 'Promotion'
    elif any(keyword in post for keyword in ['crashing', 'bug', 'error', 'annoying', 'never responds', 'broken', 'iss']):
        return 'Complaint'
    elif any(keyword in post for keyword in ['amazing', 'thank you', 'great', 'love it', 'awesome', 'best service']):
        return 'Appreciation'
    elif any(keyword in post for keyword in ['when', 'how', 'what', 'update', 'question', 'can you tell me']):
        return 'Inquiry'
    else:
        return 'Unclassified'

# Classify the specific post from the prompt
post_to_classify = "The app keeps crashing, very annoying."
classified_category = classify_social_media_post(post_to_classify)

print("--- Your Post ---")
print(f"Post: '{post_to_classify}'")
print(f"Category: {classified_category}")

--- Your Post ---
Post: "The app keeps crashing, very annoying."
Category: Complaint
```

The Gemini chat window on the right shows the following response:

i want it in code

Certainly! Here's the Python code to classify social media posts based on keywords.

The code has successfully classified the social media post "The app keeps crashing, very annoying." as a 'Complaint'. The output also shows how different posts are categorized, such as 'Promotion', 'Appreciation', and 'Inquiry', based on the keywords present in them.

Output:

This screenshot is identical to the one above, showing the same Jupyter Notebook code and Gemini AI chat window. The code classifies the post "The app keeps crashing, very annoying." as a 'Complaint'.

TASK - C:

Prompt:

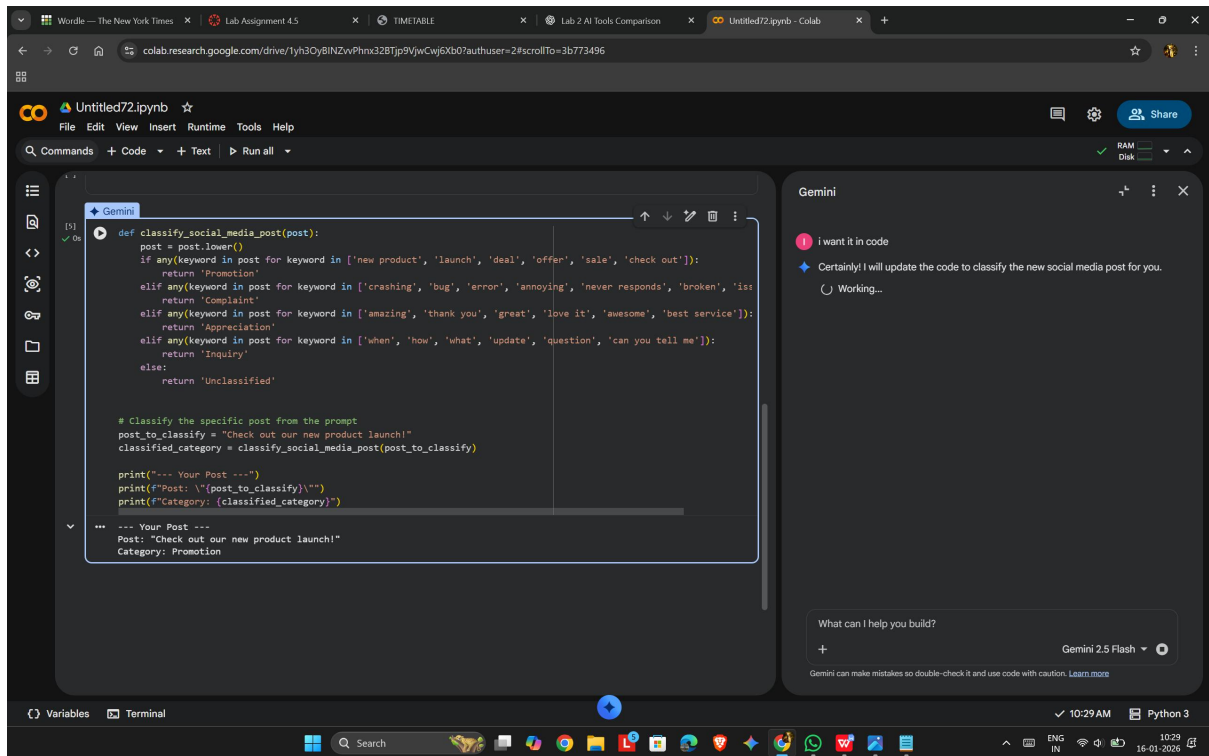
Example:

Post: "Amazing service, thank you!"

Category: Appreciation

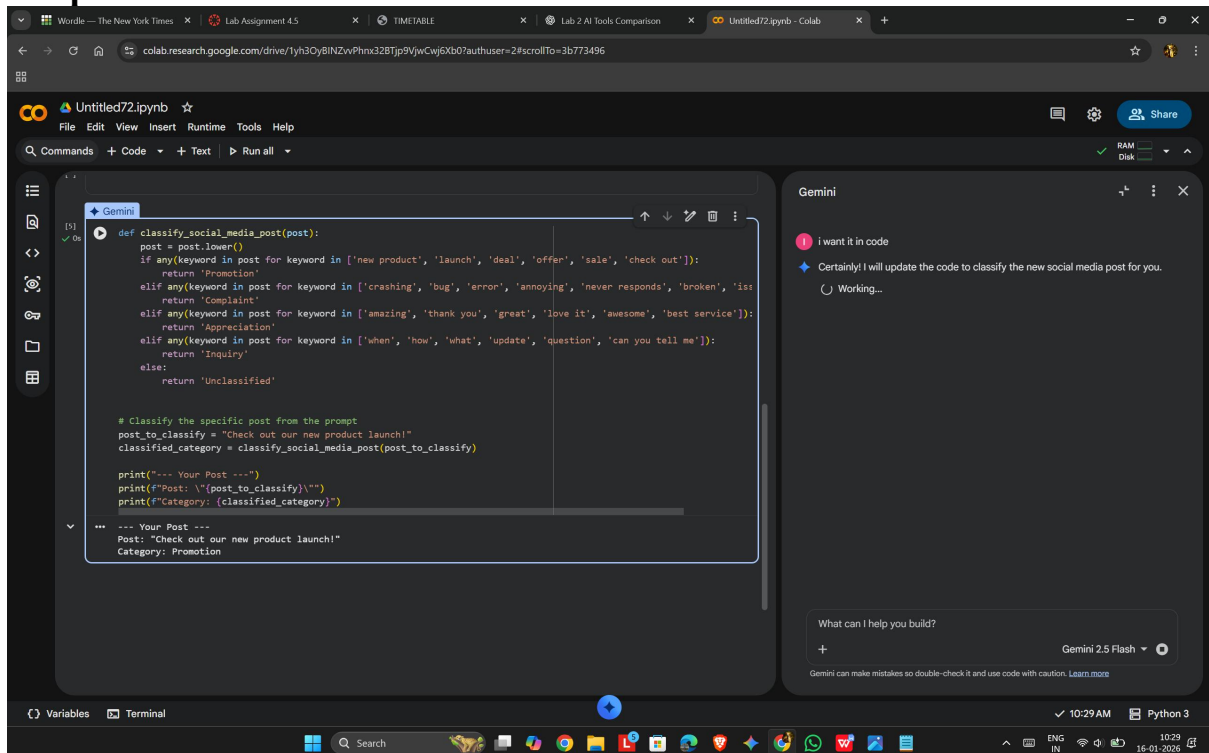
Now classify:
Post: "Check out our new product launch!"

Code:



```
[5] def classify_social_media_post(post):  
    post = post.lower()  
    if any(keyword in post for keyword in ['new product', 'launch', 'deal', 'offer', 'sale', 'check out']):  
        return 'Promotion'  
    elif any(keyword in post for keyword in ['crashing', 'bug', 'error', 'annoying', 'never responds', 'broken', 'iss  
        return 'Complaint'  
    elif any(keyword in post for keyword in ['amazing', 'thank you', 'great', 'love it', 'awesome', 'best service']):  
        return 'Appreciation'  
    elif any(keyword in post for keyword in ['when', 'how', 'what', 'update', 'question', 'can you tell me']):  
        return 'Inquiry'  
    else:  
        return 'Unclassified'  
  
# Classify the specific post from the prompt  
post_to_classify = "Check out our new product launch!"  
classified_category = classify_social_media_post(post_to_classify)  
  
print("--- Your Post ---")  
print(f"Post: \"{post_to_classify}\"")  
print(f"Category: {classified_category}")  
  
--- Your Post ---  
Post: "Check out our new product launch!"  
Category: Promotion
```

Output:



```
[5] def classify_social_media_post(post):  
    post = post.lower()  
    if any(keyword in post for keyword in ['new product', 'launch', 'deal', 'offer', 'sale', 'check out']):  
        return 'Promotion'  
    elif any(keyword in post for keyword in ['crashing', 'bug', 'error', 'annoying', 'never responds', 'broken', 'iss  
        return 'Complaint'  
    elif any(keyword in post for keyword in ['amazing', 'thank you', 'great', 'love it', 'awesome', 'best service']):  
        return 'Appreciation'  
    elif any(keyword in post for keyword in ['when', 'how', 'what', 'update', 'question', 'can you tell me']):  
        return 'Inquiry'  
    else:  
        return 'Unclassified'  
  
# Classify the specific post from the prompt  
post_to_classify = "Check out our new product launch!"  
classified_category = classify_social_media_post(post_to_classify)  
  
print("--- Your Post ---")  
print(f"Post: \"{post_to_classify}\"")  
print(f"Category: {classified_category}")  
  
--- Your Post ---  
Post: "Check out our new product launch!"  
Category: Promotion
```

TASK - D:

Prompt:

Post: "Check out our new product launch!" → Promotion

Post: "The app keeps crashing, very annoying" → Complaint

Post: "Amazing service, thank you!" → Appreciation

Now classify:

Post: "When will the next update be released?"

Code:

The screenshot shows a Jupyter Notebook environment with a dark theme. The main code area contains a Python function `classify_social_media_post(post)` that classifies posts based on keywords. The function returns 'Promotion' for launch-related keywords, 'Complaint' for negative feedback keywords, 'Appreciation' for positive feedback keywords, 'Inquiry' for questions, and 'Unclassified' otherwise. Below the function, a specific post is classified using the function.

```
# 4. When will the next update be released?
# 5. Your customer care never responds.

def classify_social_media_post(post):
    post = post.lower()
    if any(keyword in post for keyword in ['new product', 'launch', 'deal', 'offer', 'sale', 'check out']):
        return 'Promotion'
    elif any(keyword in post for keyword in ['crashing', 'bug', 'error', 'annoying', 'never responds', 'broken', 'iss']):
        return 'Complaint'
    elif any(keyword in post for keyword in ['amazing', 'thank you', 'great', 'love it', 'awesome', 'best service']):
        return 'Appreciation'
    elif any(keyword in post for keyword in ['when', 'how', 'what', 'update', 'question', 'can you tell me']):
        return 'Inquiry'
    else:
        return 'Unclassified'

# Classify the specific post from the prompt
post_to_classify = "When will the next update be released?"
classified_category = classify_social_media_post(post_to_classify)

print("--- Your Post ---")
print(f"Post: '{post_to_classify}'")
print(f"Category: {classified_category}")

--- Your Post ---
Post: "When will the next update be released?"
Category: Inquiry
```

The Gemini chat window on the right shows a conversation where the user provides the classification rules and asks for the classification of a new post. The Gemini model responds by confirming it will update the code and is currently working on it.

Post: "Check out our new product launch!" → Promotion
Post: "The app keeps crashing, very annoying" → Complaint
Post: "Amazing service, thank you!" → Appreciation
Now classify: Post: "When will the next update be released?"

Certainly! I will update the code to classify the new social media post for you.
Working...

What can I help you build?
+ Gemini 2.5 Flash

Output:

The screenshot displays a Google Colab notebook titled 'Untitled72.ipynb'. The notebook contains a Python script designed to classify social media posts based on specific keywords. The script defines a function `classify_social_media_post(post)` that checks for keywords related to product launches, complaints, appreciation, inquiries, and updates. It then applies this function to a sample post: 'When will the next update be released?'. The output of the script is shown in the cell's output area.

```
# 4. When will the next update be released?
# 5. Your customer care never responds.

def classify_social_media_post(post):
    post = post.lower()
    if any(keyword in post for keyword in ['new product', 'launch', 'deal', 'offer', 'sale', 'check out']):
        return 'Promotion'
    elif any(keyword in post for keyword in ['crashing', 'bug', 'error', 'annoying', 'never responds', 'broken', 'iss']):
        return 'Complaint'
    elif any(keyword in post for keyword in ['amazing', 'thank you', 'great', 'love it', 'awesome', 'best service']):
        return 'Appreciation'
    elif any(keyword in post for keyword in ['when', 'how', 'what', 'update', 'question', 'can you tell me']):
        return 'Inquiry'
    else:
        return 'Unclassified'

# Classify the specific post from the prompt
post_to_classify = "When will the next update be released?"
classified_category = classify_social_media_post(post_to_classify)

print("--- Your Post ---")
print(f"Post: '{post_to_classify}'")
print(f"Category: {classified_category}")

--- Your Post ---
Post: "When will the next update be released?"
Category: Inquiry
```

On the right side of the notebook, the Gemini AI interface is visible. It shows a conversation history with a user asking for help with a post classification task and Gemini responding with a plan to update the code. The current input field contains the text: 'What can I help you build?'.