

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:

```

Lab_4_5.py

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

```

RECENT SESSIONS

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

Build with Agent

AI responses may be inaccurate.

Generate Agent Instructions to onboard AI onto your codebase.

Lab_4_5.py

Describe what to build next

Agent Auto

Indexing completed.

Output:

```

PS C:\2403A51L03\3-2\AI_A_C> & 'c:\Users\isrkh\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrkh\.vscode\extens...ms-python.debugpy-2025.18.0-win32-x64\bundled\debugpy\launcher' '65398' '--' 'C:\2403A51L03\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.
3. 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.
9. I was charged after canceling my plan.

Technical Support:
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:
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:\2403A51L03\3-2\AI_A_C>

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

RECENT SESSIONS

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

Build with Agent

AI responses may be inaccurate.

Generate Agent Instructions to onboard AI onto your codebase.

Lab_4_5.py

Describe what to build next

Agent Auto

Indexing completed.

TASK - C:

Prompt:

Example:

Email: "My payment was deducted twice."

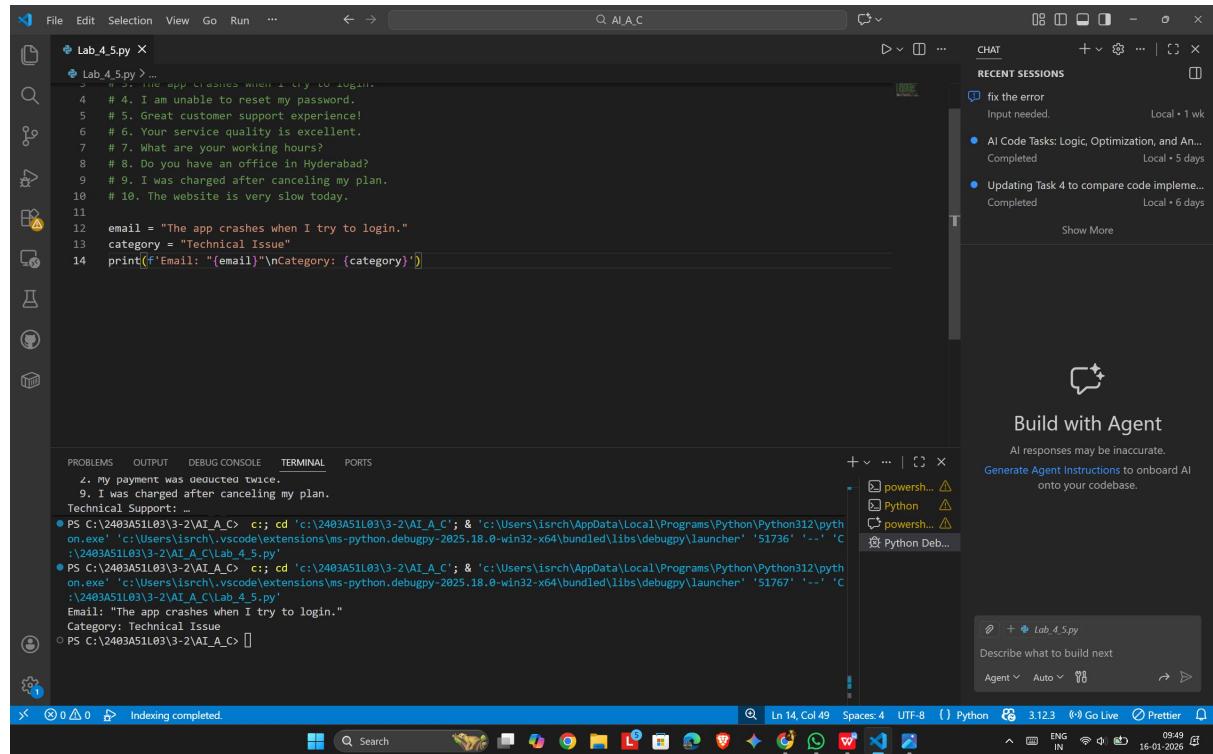
Category: Billing

Now classify this email:

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

And print it

Code:



```

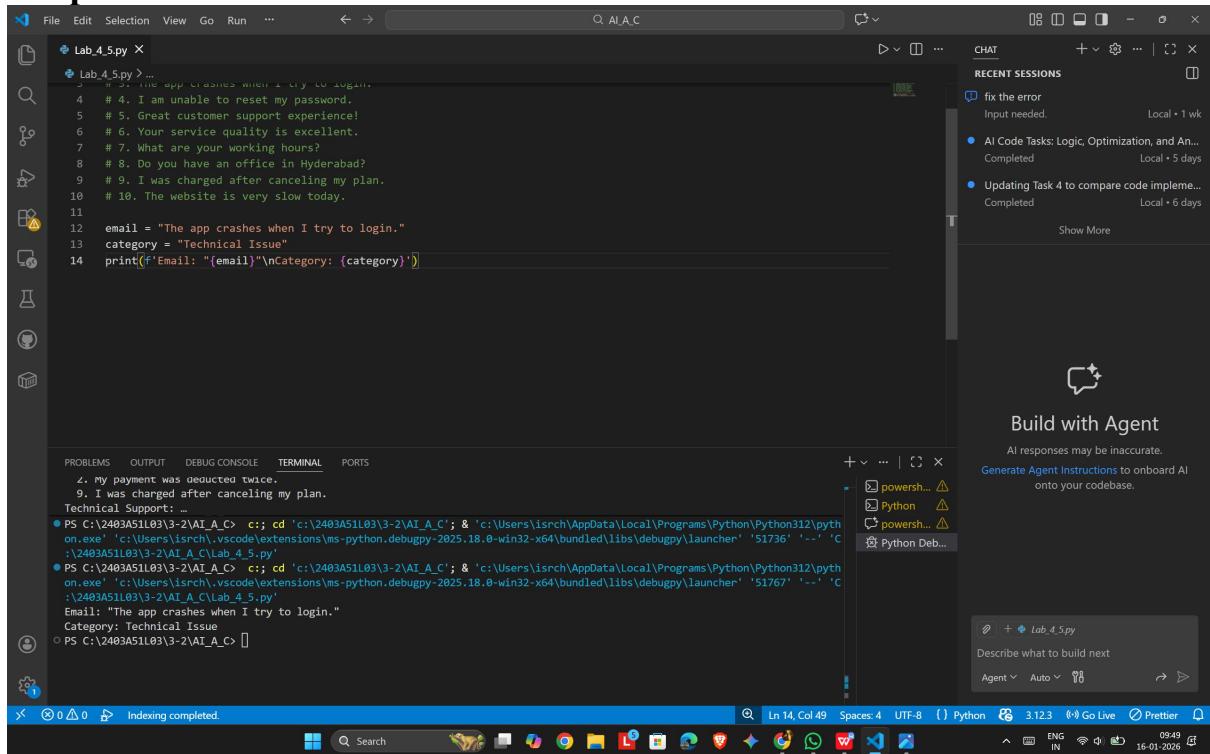
Lab_4_5.py > ...
3. # 3. The app crashes when I try to login.
4. # 4. I am unable to reset my password.
5. # 5. Great customer support experience!
6. # 6. Your service quality is excellent.
7. # 7. What are your working hours?
8. # 8. Do you have an office in Hyderabad?
9. # 9. I was charged after canceling 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 screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows a file named "Lab_4_5.py".
- Terminal:** Displays command-line history related to AI tasks and the current Python script.
- Output:** Shows indexing completion status: "Indexing completed."
- Debug Console:** Not visible in the screenshot.
- RECENT SESSIONS:** Lists recent sessions including "fix the error", "AI Code Tasks: Logic, Optimization, and An...", and "Updating Task 4 to compare code implem...".
- CHAT:** A sidebar with a message bubble icon and the heading "Build with Agent". It includes a note about AI responses being inaccurate and instructions to "Generate Agent Instructions to onboard AI onto your codebase".
- Bottom Status Bar:** Shows file path ("Lab_4_5.py"), line and column ("Ln 14, Col 49"), spaces and tabs ("Spaces: 4 - TAB: 8"), Python version ("Python 3.12.3"), Go Live button, Prettier button, and system status including battery level ("09:49"), signal strength ("ENG IN"), and date ("16-01-2026").

Output:



The screenshot shows the Visual Studio Code interface with the following details:

- Editor:** The main editor window displays the code for `Lab_4_5.py`. The code reads user input from a file and prints an email message with the subject "Email: " and the content of the input file.
- Terminal:** The terminal window shows the execution of the script in a Windows PowerShell environment (PS C:\). It lists several command-line arguments related to AI tasks and debugging.
- AI Assistant:** The right-hand sidebar features the AI Assistant interface. It includes a "RECENT SESSIONS" list with items like "fix the error" (input needed, Local, 1 wk), "AI Code Tasks: Logic, Optimization, and An..." (Completed, Local, 5 days), and "Updating Task 4 to compare code implement..." (Completed, Local, 6 days). A "Build with Agent" section is also present.
- Status Bar:** The bottom status bar shows the current file is `Lab_4_5.py`, the line number is 14, column is 49, spaces are 4, encoding is UTF-8, Python version is 3.12.3, and the date/time is 16-01-2026.

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:\2403A51L03\3-2\AI_A_C> c:; cd 'c:\2403A51L03\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\vscodeextensions\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>

Indexing completed.

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:\2403A51L03\3-2\AI_A_C> c:; cd 'c:\2403A51L03\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\vscodeextensions\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>

Indexing completed.

2.

TASK - A:

Categories

Flight Booking
 Hotel Booking
 Cancellation
 General Travel Info

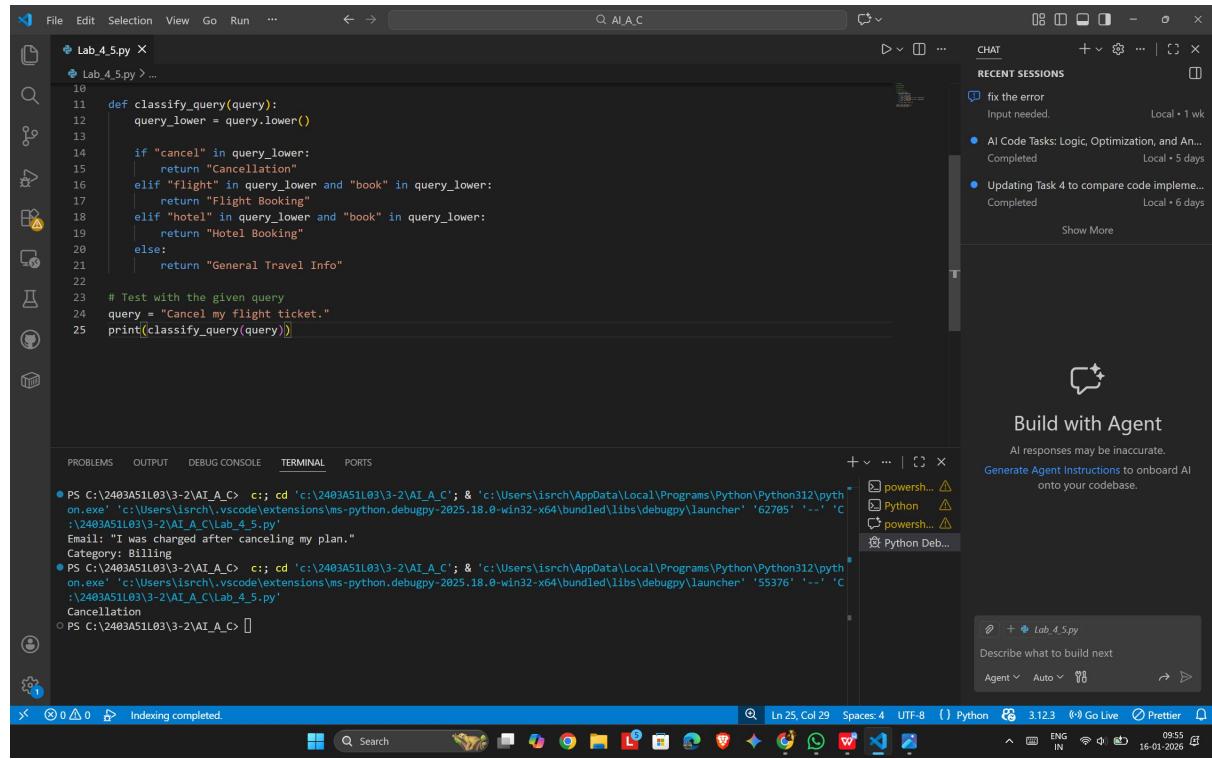
TASK - B:

Prompt:

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

Query: "Cancel my flight ticket."

Code:



```

File Edit Selection View Go Run ...
Lab_4_5.py > ...
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))

```

RECENT SESSIONS

- fix the error Local • 1 wk
- Input needed.
- AI Code Tasks: Logic, Optimization, and An... Completed Local • 5 days
- Updating Task 4 to compare code implemen... Completed Local • 6 days
- Show More

CHAT

Build with Agent

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

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

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>

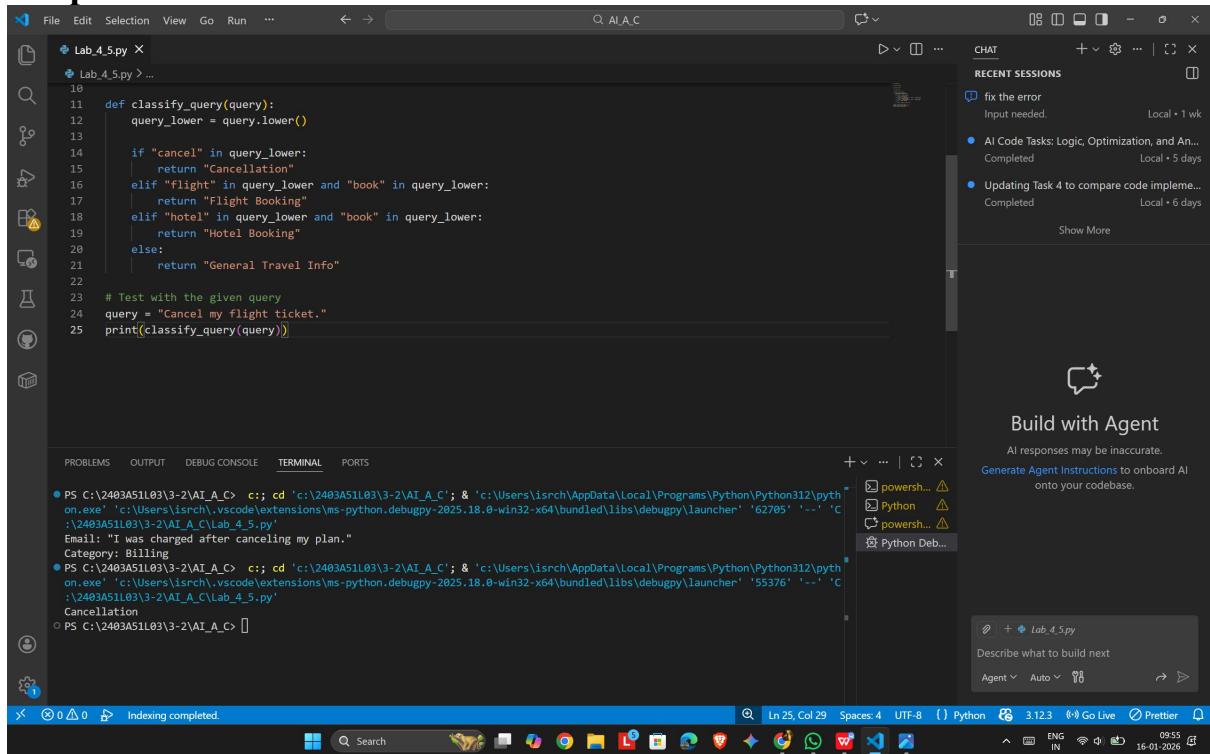
Indexing completed

Search

Ln 25, Col 29 Spaces: 4 UTF-8 Python 3.12.3 Go Live Prettier

0955 16-01-2026

Output:



```

File Edit Selection View Go Run ... < > AI/AI C RECENT SESSIONS CHAT + ... RECENT SESSIONS
Lab_4_5.py > ...
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))

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

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>

```

Indexing completed.

CHAT + ... RECENT SESSIONS CHAT + ... RECENT SESSIONS

RECENT SESSIONS

fix the error Input needed. Local • 1 wk

AI Code Tasks: Logic, Optimization, and An... Completed Local • 5 days

Updating Task 4 to compare code implem... Completed Local • 6 days

Show More

Build with Agent

All responses may be inaccurate.

Generate Agent Instructions to onboard AI onto your codebase.

+ Lab_4_5.py

Describe what to build next

Agent Auto 09:55 16-01-2026

TASK - C:

Prompt:

Example:

Query: "Book a flight from Delhi to Mumbai"

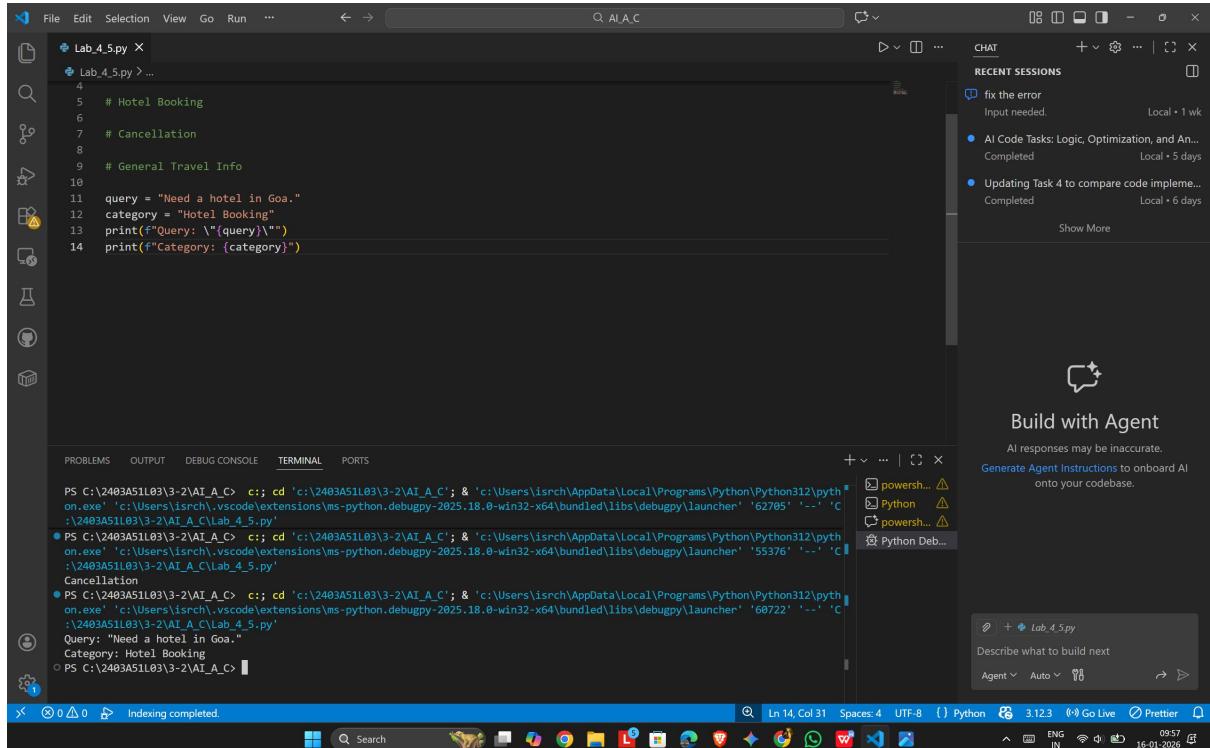
Category: Flight Booking

Now classify:

Query: "Need a hotel in Goa."

And print it

Code:



```

File Edit Selection View Go Run ... ⏪ ⏩ 🔍 AI/AI
RECENT SESSIONS
fix the error Input needed. Local · 1 wk
AI Code Tasks: Logic, Optimization, and An... Completed Local · 5 days
Updating Task 4 to compare code implem... Completed Local · 6 days
Show More

Lab_4_5.py ×
Lab_4_5.py > ...
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}")

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS + ⌂ X

```

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\vscodeextensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '62705' '---' 'C:\2403A51L03\3-2\AI_A_C\Lab_4_5.py'
● 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\vscodeextensions\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> cd 'c:\2403A51L03\3-2\AI_A_C'; & 'c:\Users\isrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\isrch\vscodeextensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '60722' '---' 'C:\2403A51L03\3-2\AI_A_C\Lab_4_5.py'
Query: "Need a hotel in Goa."
Category: Hotel Booking
○ PS C:\2403A51L03\3-2\AI_A_C>

```

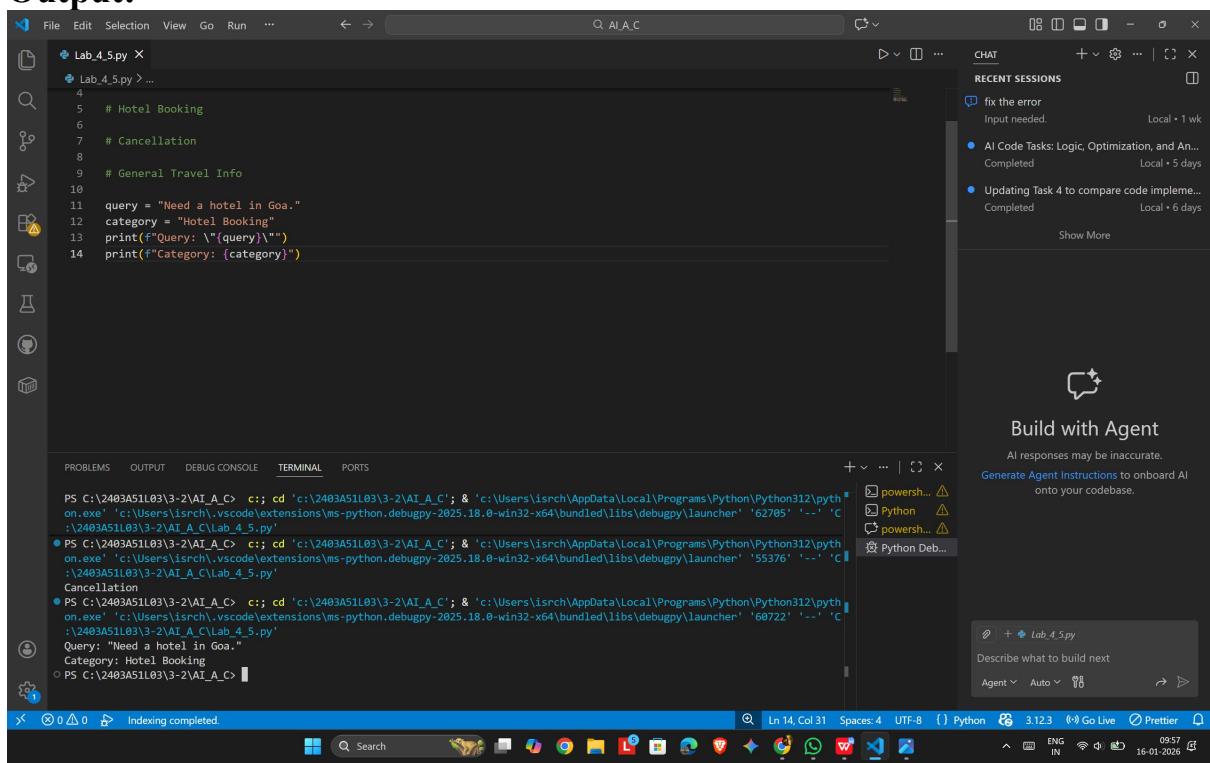
+ Lab_4_5.py

Describe what to build next

Agent Auto 09:57 16-01-2026

Indexing completed.

Output:



```

File Edit Selection View Go Run ... ⏪ ⏩ 🔍 AI/AI
RECENT SESSIONS
fix the error Input needed. Local · 1 wk
AI Code Tasks: Logic, Optimization, and An... Completed Local · 5 days
Updating Task 4 to compare code implem... Completed Local · 6 days
Show More

Lab_4_5.py ×
Lab_4_5.py > ...
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}")

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS + ⌂ X

```

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\vscodeextensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '62705' '---' 'C:\2403A51L03\3-2\AI_A_C\Lab_4_5.py'
● PS C:\2403A51L03\3-2\AI_A_C> cd 'c:\2403A51L03\3-2\AI_A_C'; & 'c:\Users\isrch\vscodeextensions\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> cd 'c:\2403A51L03\3-2\AI_A_C'; & 'c:\Users\isrch\vscodeextensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '60722' '---' 'C:\2403A51L03\3-2\AI_A_C\Lab_4_5.py'
Query: "Need a hotel in Goa."
Category: Hotel Booking
○ PS C:\2403A51L03\3-2\AI_A_C>

```

+ Lab_4_5.py

Describe what to build next

Agent Auto 09:57 16-01-2026

Indexing completed.

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:

The screenshot shows a Microsoft Visual Studio Code (VS Code) window with the following details:

- File Explorer:** Shows a file named "Lab_4.py" with the content provided below.
- Code Editor:** The code defines a function `classify_query` that takes a query string and returns a category based on keywords like "flight", "book", "hotel", "accommodation", "cancel", or "General Travel Info".
- Terminal:** The terminal shows the execution of the script and its output:
 - Line 1: `PS C:\> python Lab_4.py`
 - Line 2: `Best time to visit Kashmir?"`
 - Line 3: `Classification: General Travel Info`
 - Line 4: `PS C:\>`
- Output:** Shows the output of the script execution.
- Debug Console:** Shows the output of the script execution.
- Terminal:** Shows the output of the script execution.
- Ports:** Shows the output of the script execution.
- Problems:** Shows the output of the script execution.
- CHAT:** A sidebar with a message "fix the error" and a link to "Input needed".
- RECENT SESSIONS:** A sidebar with items: "AI Code Tasks: Logic, Optimization, and An...", "Updating Task 4 to compare code implemen...", and "Completed Local 5 days ago".
- Build with Agent:** A sidebar with a message "AI responses may be inaccurate. Generate Agent Instructions to onboard AI onto your codebase." and a speech bubble icon.
- Taskbar:** Shows the status bar with "Indexing completed.", the current file "Lab_4.py", the line number "Ln 22, Col 29", the space count "Spaces: 4", the file type "UTF-8", the language "Python", the file size "3.123", the file extension ".py", the file status "IN", and the date "16-01-2026".

Output:

The screenshot shows a Microsoft Visual Studio Code interface with the following details:

- File Explorer:** Shows a file named "Lab_4.py" with content related to travel booking.
- Terminal:** Displays a Python session running on Windows 10. The session shows imports from "ms-python.debugpy" and "ms-python.vscode-languagefeatures", and runs the "classify_query" function with the input "Best time to visit Kashmir?" resulting in the output "Cancellation".
- Output:** Shows the message "Indexing completed."
- Search:** A search bar at the top with the text "Q AI_A_C".
- Chat:** A sidebar titled "CHAT" with a message "fix the error Input needed." and two recent sessions: "AI Code Tasks: Logic, Optimization, and An..." and "Updating Task 4 to compare code impleme...".
- Problems:** A sidebar showing no problems.
- Terminal:** A sidebar showing the current working directory as "PS C:\...\Lab_4.py" and the command "python -m vscode.debug".
- Ports:** A sidebar showing port information: "powerhell" (localhost:55376), "Python" (localhost:56722), "powerhell" (localhost:55376), and "Python Deb..." (localhost:60722).
- Build with Agent:** A sidebar with the message "AI responses may be inaccurate. Generate Agent Instructions to onboard AI onto your codebase." and a "Next" button.

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:

The screenshot shows a VS Code interface with the following details:

- File:** Lab_4_5.py
- Content:**

```

Lab_4_5.py > ...
1 # 1. Why am I getting an indentation error in Python?
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}")

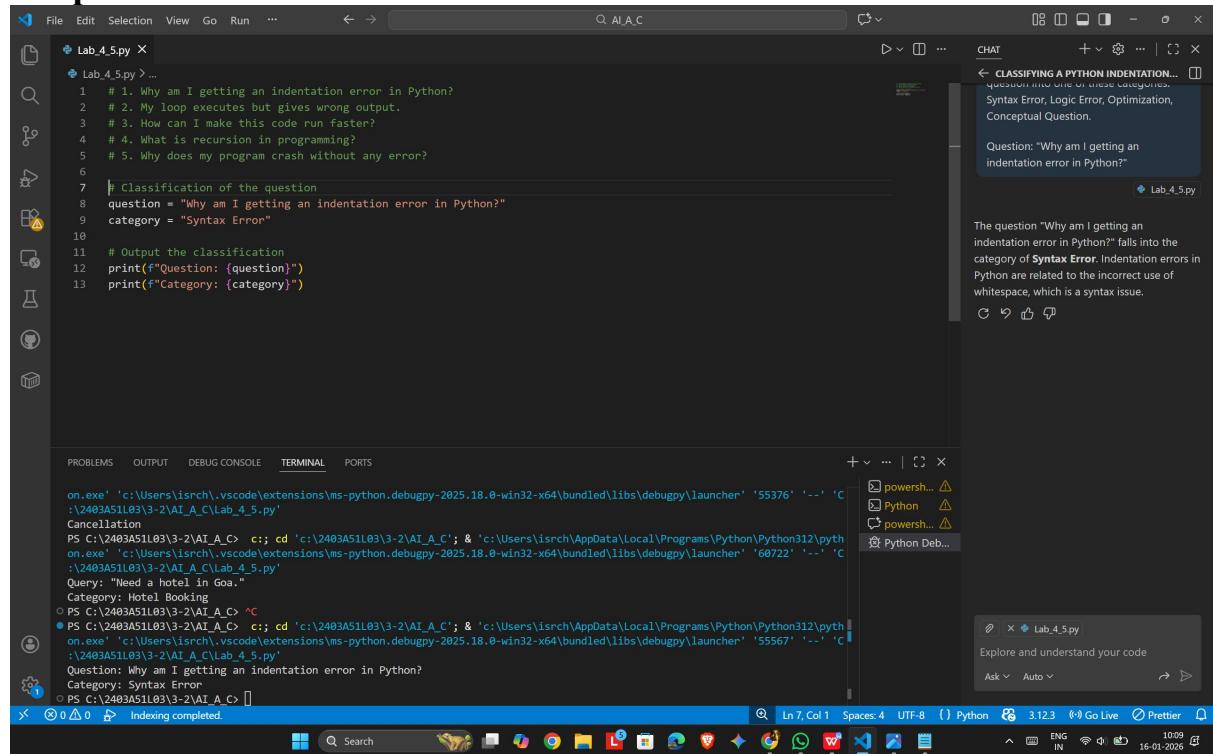
```
- Terminal:**

```

on.exe' 'c:\Users\isrchi.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> :&; cd 'c:\2403A51L03\3-2\AI_A_C'; & 'c:\Users\isrchi\AppData\Local\Programs\Python\Python312\pyt
on.exe' 'c:\Users\isrchi\vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '69722' '...
:C:\2403A51L03\3-2\AI_A_C\Lab_4_5.py'
Query: "Need a hotel in Goa."
Category: Hotel Booking
PS C:\2403A51L03\3-2\AI_A_C> ^
● PS C:\2403A51L03\3-2\AI_A_C> :&; cd 'c:\2403A51L03\3-2\AI_A_C'; & 'c:\Users\isrchi\AppData\Local\Programs\Python\Python312\pyt
on.exe' 'c:\Users\isrchi\vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '55567' '...
:C:\2403A51L03\3-2\AI_A_C\Lab_4_5.py'
Question: Why am I getting an indentation error in Python?
Category: Syntax Error
○ PS C:\2403A51L03\3-2\AI_A_C> []

```
- Right Panel:**
 - CHAT: 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?"
 - Details: 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.
- Bottom Status Bar:**
 - Ln 7, Col 1 Spaces: 4 UTF-8 Python 3.12.3 Go Live Prettier
 - System icons: ENG IN 10:09 16-01-2026

Output:



The screenshot shows a VS Code interface with the following details:

- File Explorer:** Shows a file named "Lab_4_5.py" with content related to AI classification.
- Editor:** Displays a Python script with code snippets and comments about AI classification.
- Terminal:** Shows command-line history related to the file.
- Output:** Shows logs from the Python extension, indicating indexing completion.
- CHAT:** A sidebar titled "CLASSIFYING A PYTHON INDENTATION..." with a message about syntax errors.
- Problems:** Shows no problems.
- Search:** Shows search results for "powershell" and "Python".
- Activity Bar:** Includes icons for File, Edit, Selection, View, Go, Run, etc.

TASK - C:

Prompt:

Example:

Question: "What is recursion in programming?"

Category: Conceptual Question

Now classify:

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

Code:

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. On the left, there's a code editor with a file named 'Lab_4_5.py'. The code is a script that classifies different types of errors based on user input. Below the code editor is a terminal window displaying the execution of the script and its output.

```
# 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?  
  
def classify_question(question):  
    if "recursion" in question:  
        return "Conceptual Question"  
    elif "loop" in question and "wrong output" in question:  
        return "Syntax Error"  
    elif "crash" in question:  
        return "Runtime Error"  
    else:  
        return "General Question"  
  
# Example usage  
question = "My loop executes but gives wrong output."  
category = classify_question(question)  
print(f"Question: \"{question}\"\nCategory: {category}")
```

The terminal output shows the script being run and its results:

```
PS C:\> python Lab_4_5.py  
Query: "Need a hotel in Goa."  
Category: Hotel Booking  
PS C:\> python Lab_4_5.py  
Query: "Why am I getting an indentation error in Python?"  
Category: Syntax Error  
PS C:\> python Lab_4_5.py  
Query: "My loop executes but gives wrong output."  
Category: Logic Error  
PS C:\> python Lab_4_5.py  
Query: "How can I make this code run faster?"  
Category: General Question  
PS C:\> python Lab_4_5.py  
Query: "What is recursion in programming?"  
Category: Conceptual Question
```

A floating sidebar on the right is titled "CLASSIFYING A PYTHON INDENTATION..." and contains a single question: "Question: Why am I getting an indentation error in Python?". It also includes a link to "Lab_4_5.py".

Output:

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The main area displays a Python file named `Lab_4_5.py` with the following content:

```
# 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?  
  
def classify_question(question):  
    if "recursion" in question:  
        return "Conceptual Question"  
    elif "loop" in question and "wrong output" in question:  
        return "Logic Error"  
    elif "crash" in question:  
        return "Runtime Error"  
    else:  
        return "General Question"  
  
# Example usage  
question = "My loop executes but gives wrong output."  
category = classify_question(question)  
print(f"Question: \"{question}\"\nCategory: {category}")
```

Below the code editor are tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS. The TERMINAL tab is active, showing command-line history related to the file:

- `:2403A51l03\3-2\AI_A_C\lab_4_5.py`
- Query: "Need a hotel in Goa."
- Category: Hotel Booking
- `PS C:\2403A51l03\3-2\AI_A_C> ^C`
- `PS C:\2403A51l03\3-2\AI_A_C> cd 'c:\2403A51l03\3-2\AI_A_C' & 'c:\Users\lsrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\lsrch\.vscode\extensions\ms-python.python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '55567' ... 'c:\2403A51l03\3-2\AI_A_C\lab_4_5.py'`
- Question: "Why am I getting an indentation error in Python?"
- Category: Syntax Error
- `PS C:\2403A51l03\3-2\AI_A_C> cd 'c:\2403A51l03\3-2\AI_A_C' & 'c:\Users\lsrch\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\lsrch\.vscode\extensions\ms-python.python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '55685' ... 'c:\2403A51l03\3-2\AI_A_C\lab_4_5.py'`
- Question: "My loop executes but gives wrong output."
- Category: Logic Error
- `PS C:\2403A51l03\3-2\AI_A_C> []`

The status bar at the bottom indicates "Indexing completed." On the right side, there is a sidebar titled "CLASSIFYING A PYTHON INDENTATION..." which includes a "CHAT" section and a "Question" box containing the text "Why am I getting an indentation error in Python?". Below this, a note states: "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." There are also icons for GitHub, LinkedIn, and other social media.

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:

```

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']):
        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?

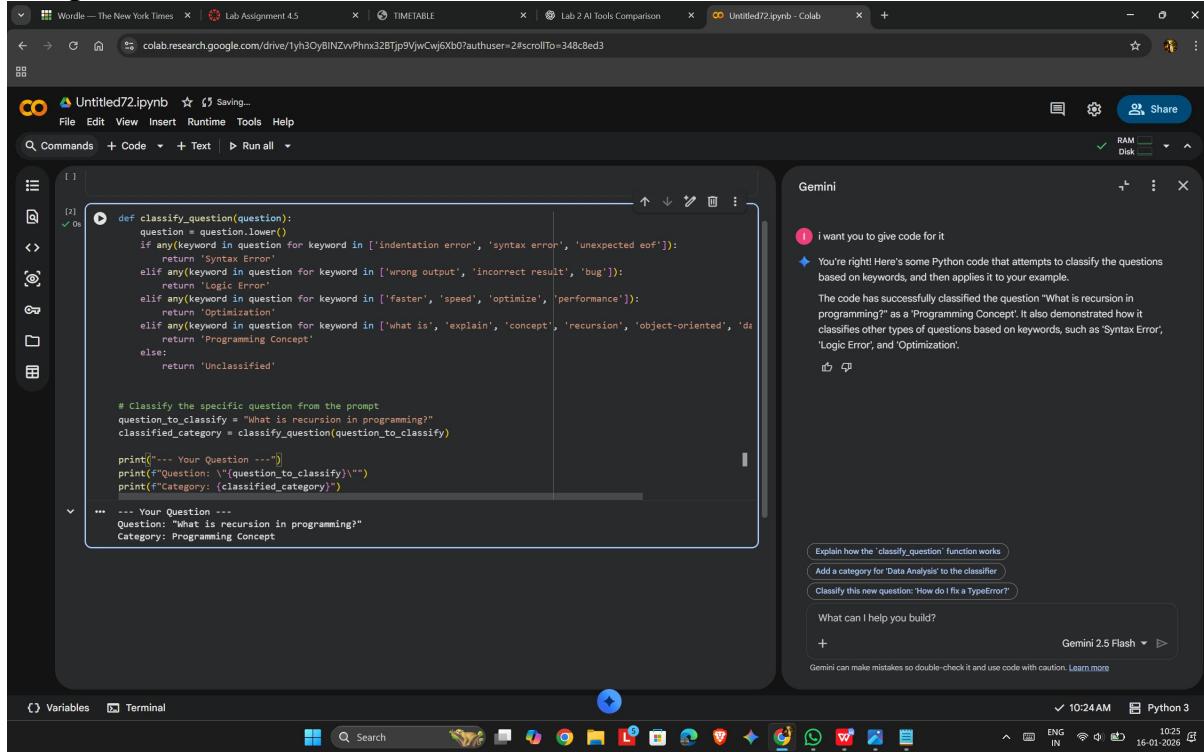
What can I help you build?

+
Gemini 2.5 Flash >

Gemini can make mistakes so double-check it and use code with caution. Learn more

10:24 AM Python 3

Output:



The screenshot shows a Jupyter Notebook cell containing Python code for classifying questions based on keywords. The code defines a function `classify_question` that checks for various error types and optimization terms. It then prints a specific question and its classification.

```

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']):
        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

```

To the right, the Gemini AI interface is shown, responding to a prompt about the code. It has a sidebar with options like "Explain how the 'classify_question' function works", "Add a category for 'Data Analysis' to the classifier", and "Classify this new question: How do I fix a TypeError?".

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 Google Colab notebook titled "Untitled72.ipynb". The code defines a function `classify_social_media_post` that takes a post as input and returns its category based on specific keywords. A sample post is classified as "Complaint". The Gemini sidebar provides context about the code and its output.

```

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', 'is']):
        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

```

Output:

The screenshot shows the same Google Colab notebook and code as the previous one. The Gemini sidebar reiterates the successful classification of the post as "Complaint".

```

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', 'is']):
        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

```

TASK - C:

Prompt:

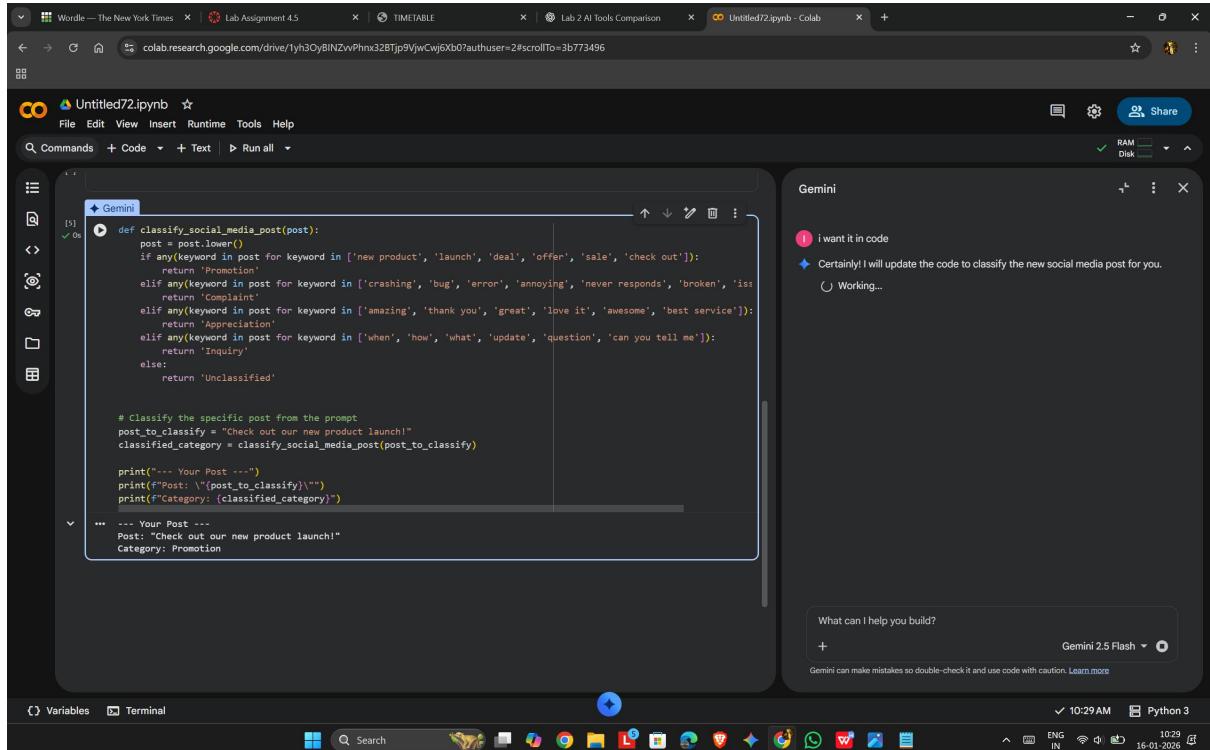
Example:

Post: "Amazing service, thank you!"

Category: Appreciation

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

Code:



```

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', 'is']):
        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'

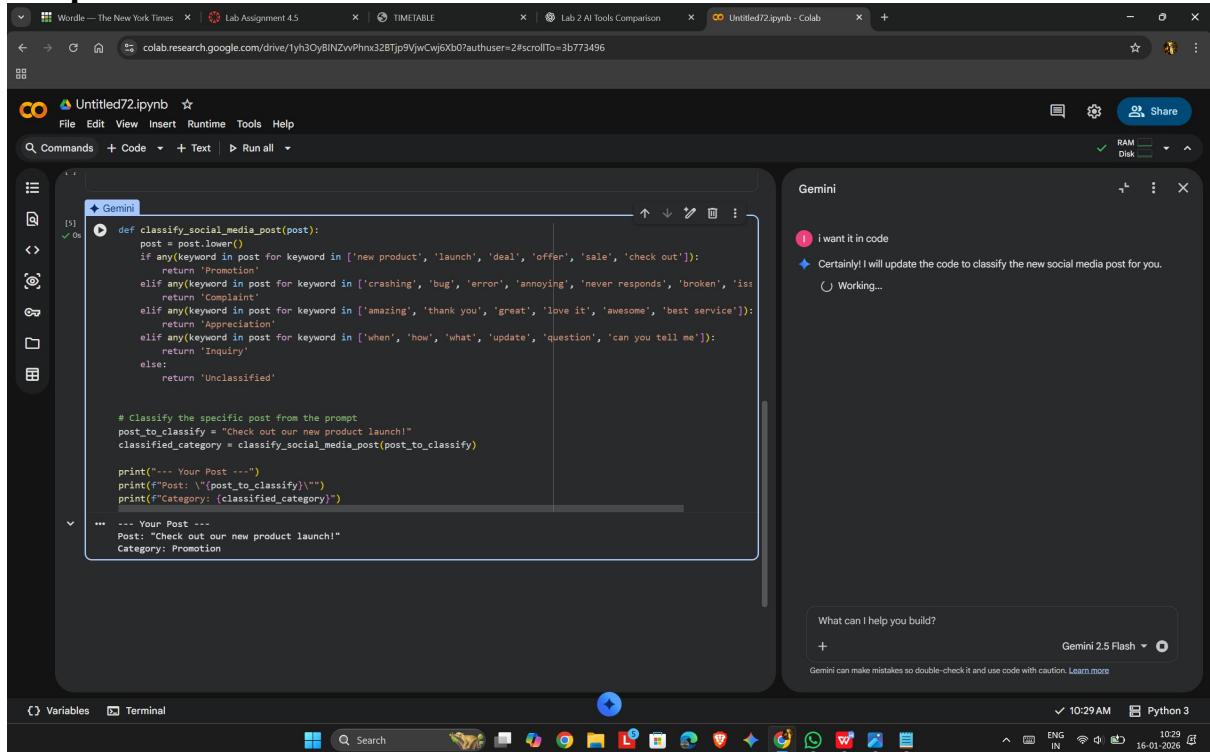
# Classify the specific post from the prompt
post_to_classify = "Check out our new product launch!"
classified_category = classify_social_media_post(post_to_classify)

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:



```

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', 'is']):
        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

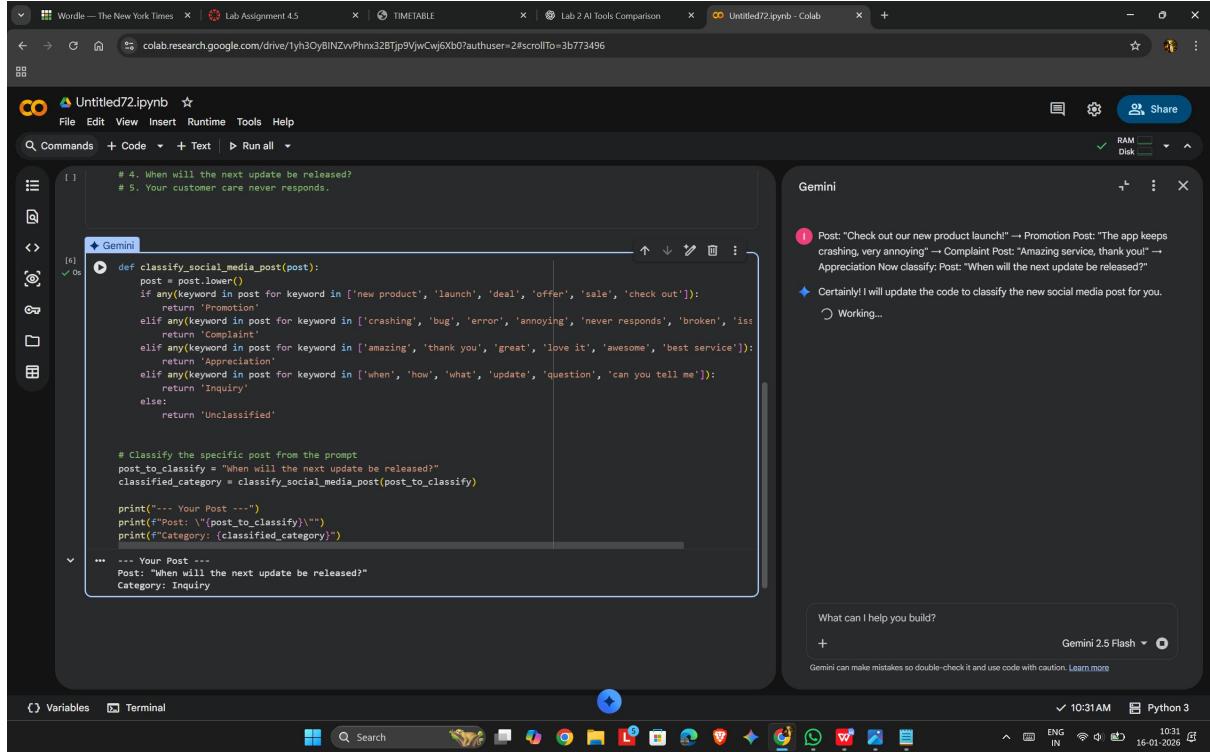
Post: "The app keeps crashing, very annoying" → Complaint

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

Now classify:

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

Code:



```
# 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
```

Gemini

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?"

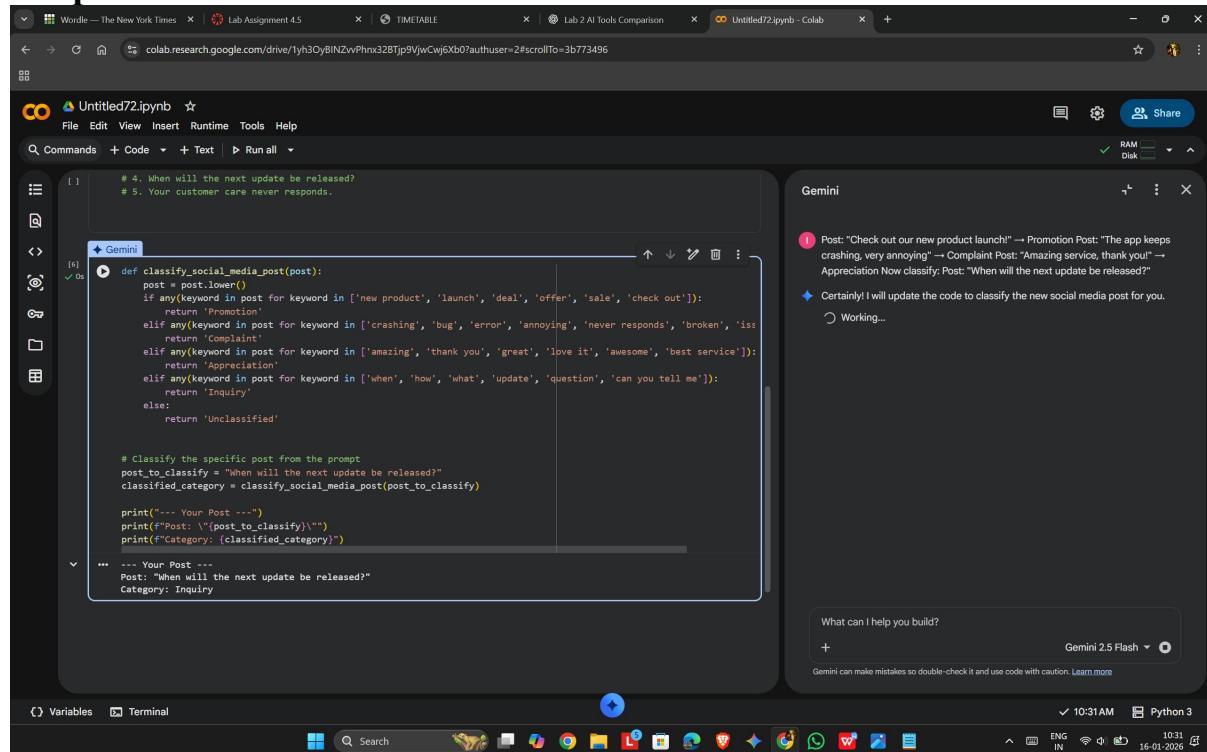
Certainly! I will update the code to classify the new social media post for you.

Working...

What can I help you build?

Gemini 2.5 Flash

10:31 AM Python 3 ENG IN 16-01-2026

Output:


The screenshot shows a Jupyter Notebook interface in Google Colab. On the left, a code cell contains Python code for classifying social media posts based on keywords. The code defines a function `classify_social_media_post` that checks for keywords related to promotion, complaint, appreciation, inquiry, or uncategorized posts. A prompt is provided to classify a post about the next update being released.

```

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', 'is']):
        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, a Gemini AI interface is shown. It has a history of interactions where the user asks about classifying a post and the AI responds with the classification. Below the history, there's a text input field asking "What can I help you build?" and a note that Gemini can make mistakes so double-check it and use code with caution.