From Siri to Smartypants

Abstract:

"From Siri to Smartypants: Building Your Personalized AI Assistant" introduces a transformative project aimed at empowering users with a customized AI assistant to enhance their daily lives. By leveraging cutting-edge technologies, including speech recognition, text-to-speech conversion, and integration with external APIs, the project delivers Smartypants, a versatile assistant capable of providing seamless assistance and boosting productivity.

Smartypants offers a range of functionalities, including retrieving information from Wikipedia, performing Google searches, providing real-time weather updates, playing music on YouTube, setting reminders, telling jokes, and answering basic questions. Users interact with Smartypants through voice commands, allowing for natural and intuitive communication.

The project's core technologies, such as Python libraries for speech recognition and text-to-speech conversion, ensure efficient and accurate processing of user queries. Additionally, integration with external APIs like Wikipedia, Google, and YouTube enables Smartypants to access vast repositories of information and deliver relevant responses promptly.

By developing Smartypants, the project aims to streamline daily tasks and provide users with a reliable assistant to simplify their lives. With its robust functionality and user-friendly interface, Smartypants represents a significant step towards personalized AI assistance tailored to individual needs.

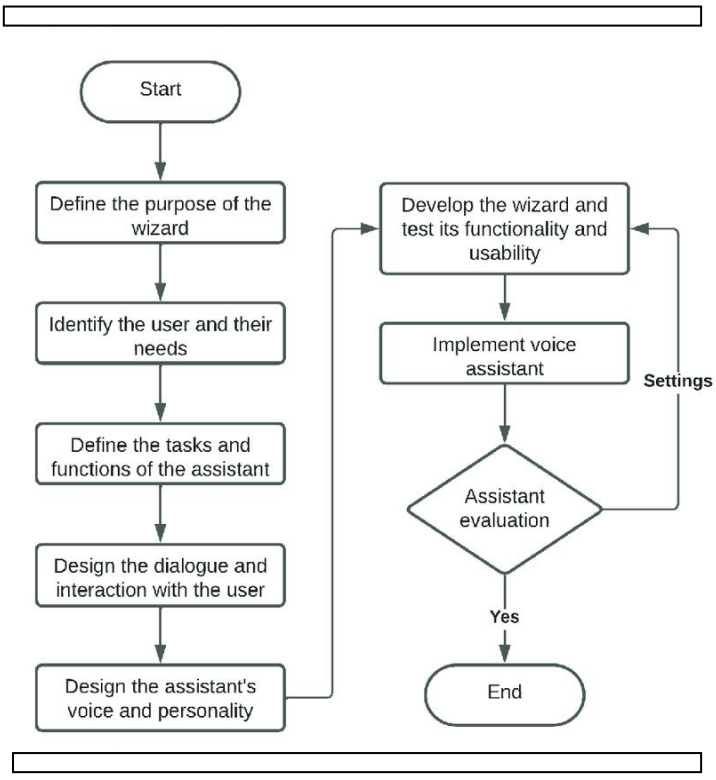
System Requirements:

- Operating System: Windows 10 / macOS Mojave / Ubuntu 18.04 LTS or later
- Processor: Intel Core i5 or equivalent
- RAM: 8 GB
- Storage: 500 MB of available space
- Additional Requirements: Internet connection for API calls, Microphone for voice input, Speaker for audio output.

Tools and Versions:

- Python: 3.8
- Pyttsx3: 2.90
- Wikipedia-API: 0.5.4
- Requests: 2.25.1
- SpeechRecognition: 3.8.1
- Google Search Python: Latest version
- Webbrowser module: Default with Python
- Pytube: 12.0.0
- Dateutil: 2.8.1

Flow Chart:



Source code:

```
import pyttsx3

import datetime

import requests

import speech_recognition as sr

import wikipedia

import webbrowser

import time

import pyjokes

from newsapi import NewsApiClient

from pytube import Search

from dateutil import parser


engine = pyttsx3.init()

def speak(text):

    engine.say(text)

    engine.runAndWait()

def perform_google_search(query):

    speak("Performing Google search...")

    search_url = f"https://www.google.com/search?q={query}"

    webbrowser.open(search_url)

def get_weather(api_key, city):

    url = f"http://api.openweathermap.org/data/2.5/weather?q={city}&appid={api_key}&units=metric"

    try:

        response = requests.get(url)

        response.raise_for_status()
```

```
        data = response.json()
        weather_info = {
            "city": data["name"],
            "temperature": data["main"]["temp"],
            "description": data["weather"][0]["description"],
            "humidity": data["main"]["humidity"],
            "wind_speed": data["wind"]["speed"]
        }
        return weather_info
    except requests.RequestException:
        return None
def recognize_speech():
    recognizer = sr.Recognizer()
    with sr.Microphone() as source:
        speak("Listening...")
        recognizer.adjust_for_ambient_noise(source)
        audio = recognizer.listen(source)
    try:
        query = recognizer.recognize_google(audio)
        print(f"You said: {query}")
        return query.lower()
    except sr.UnknownValueError:
        return "Sorry, I didn't understand that."
    except sr.RequestError:
        return "Sorry, the service is unavailable."
def set_reminder(reminder_time, reminder_message):
```

```
    now = datetime.datetime.now()
    try:
        reminder_time = parser.parse(reminder_time)
        delay = (reminder_time - now).total_seconds()
        if delay > 0:
            speak(f"Reminder set for {reminder_time}.")
            time.sleep(delay)
            speak(f"Reminder: {reminder_message}")
        else:
            speak("The time you entered has already passed.")
    except ValueError:
        speak("The time format is incorrect. Please use a recognizable date and time format.")
def handle_reminder_query(query):
    query = query.replace("set reminder", "").strip()
    parts = query.split(" at ")
    if len(parts) == 2:
        reminder_message = parts[0].strip()
        reminder_time = parts[1].strip()
        set_reminder(reminder_time, reminder_message)
        return f"Reminder set for {reminder_time} to {reminder_message}."
    else:
        speak("Please provide the reminder in the format 'set reminder [your reminder message] at [YYYY-MM-DD
HH:MM:SS]' or a recognizable date and time format.")
        return "Sorry, I couldn't understand the reminder time."
def play_music_on_youtube(song_name):
    speak(f"Searching for {song_name} on YouTube.")
    search = Search(song_name)
```

```
    result = search.results[0]
    webbrowser.open(result.watch_url)
    speak(f"Playing {song_name} on YouTube.")
def handle_music_query(query):
    if 'play music' in query:
        song_name = query.replace('play music', '').strip()
        if song_name:
            play_music_on_youtube(song_name)
            return f"Playing {song_name} on YouTube."
        else:
            return "Sorry, I couldn't understand the song name."
    return "No music command detected."
def tell_joke():
    joke = pyjokes.get_joke()
    speak(joke)
    return joke
def handle_joke_query(query):
    if 'joke' in query:
        return tell_joke()
def get_news(api_key, query):
    newsapi = NewsApiClient(api_key=api_key)
    top_headlines = newsapi.get_top_headlines(q=query, language='en', country='us')
    articles = top_headlines['articles']
    if articles:
        news = [article['title'] for article in articles[:5]]
        news_str = ". ".join(news)
```

```
        speak(news_str)
        return news_str
    else:
        return "Sorry, no news found for that topic."
def handle_news_query(query, api_key):
    query = query.replace("news about", "").strip()
    return get_news(api_key, query)
def handle_query(query, api_keys):
    if 'wikipedia' in query:
        return handle_wikipedia_query(query)
    elif 'google search' in query:
        query = query.replace("google search", "").strip()
        perform_google_search(query)
        return "I have performed the Google search."
    elif 'weather' in query:
        query = query.replace("weather", "").strip()
        return handle_weather_query(query, api_keys['weather'])
    elif 'time' in query:
        return datetime.datetime.now().strftime("The current time is %H:%M")
    elif 'date' in query:
        return datetime.datetime.now().strftime("Today's date is %B %d, %Y")
    elif 'how are you' in query:
        response = "I'm fine, thank you"
        speak(response)
        return response
    elif 'what is your name' in query:
```

```
        response = "My name is Smartypants, your personal AI assistant."
        speak(response)
        return response
    elif 'who am i' in query:
        response = "You are my boss"
        speak(response)
        return response
    elif 'set reminder' in query:
        return handle_reminder_query(query)
    elif 'play music' in query:
        return handle_music_query(query)
    elif 'joke' in query:
        return handle_joke_query(query)
    elif 'news about' in query:
        return handle_news_query(query, api_keys['news'])
    else:
        response = "I'm sorry, I can't help with that right now."
        speak(response)
        return response
def handle_wikipedia_query(query):
    speak("Searching Wikipedia...")
    query = query.replace("wikipedia", "").strip()
    try:
        result = wikipedia.summary(query, sentences=2)
        speak("According to Wikipedia:")
        speak(result)
```

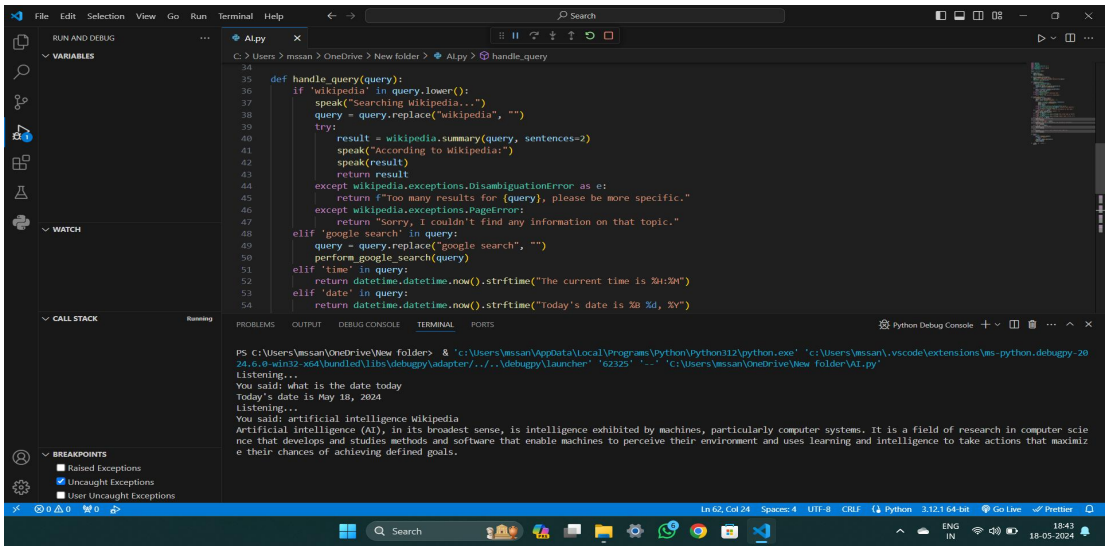
```
            return result
        except wikipedia.exceptions.DisambiguationError as e:
            return f"Too many results for {query}, please be more specific."
        except wikipedia.exceptions.PageError:
            return "Sorry, I couldn't find any information on that topic."
    def handle_weather_query(query, api_key):
        speak("Getting weather information...")
        weather_data = get_weather(api_key, query)
        if weather_data:
            result = (f"Weather in {weather_data['city']}: "
                      f"Temperature: {weather_data['temperature']}°C, "
                      f"Description: {weather_data['description']}, "
                      f"Humidity: {weather_data['humidity']}%, "
                      f"Wind Speed: {weather_data['wind_speed']} m/s.")
            speak(result)
            return result
        else:
            result = "Sorry, I couldn't find the weather information for that location."
            speak(result)
            return result
    def main():
        weather_api_key = "6e6f9659fef62e5c5d1103979100d281"
        news_api_key = "dbe57b028aeb41e285a226a94865f7a7"
        api_keys = {'weather': weather_api_key, 'news': news_api_key}
        while True:
            query = recognize_speech()
```

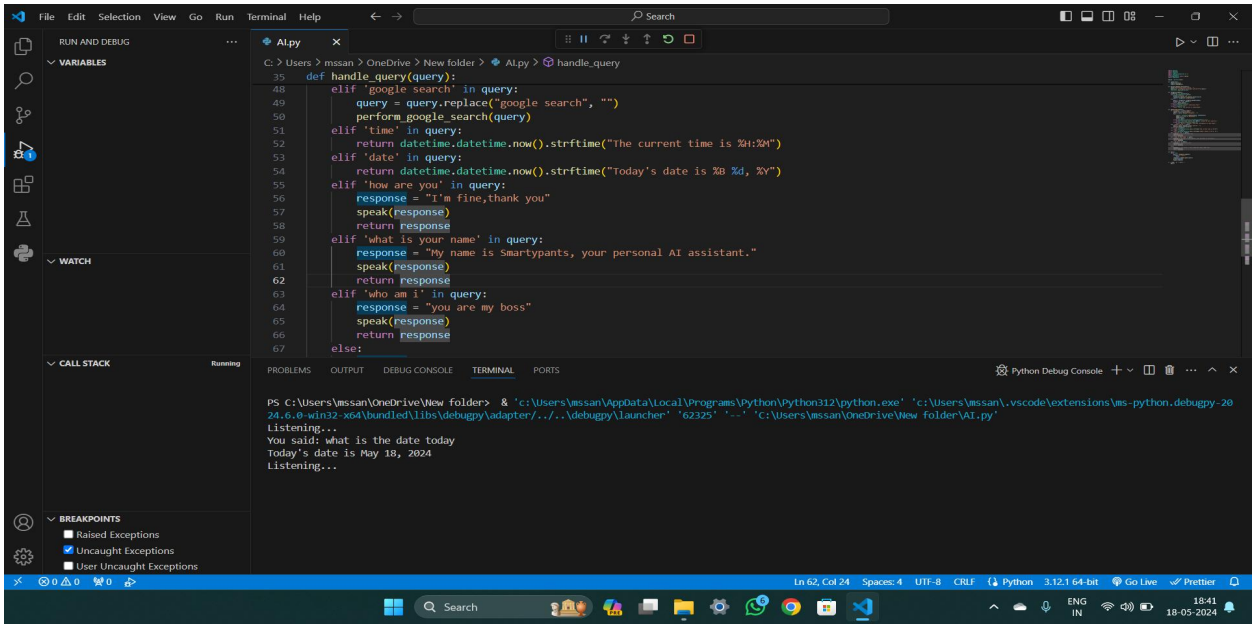
```
        if "stop" in query:
            speak("Goodbye! Have a nice day!")
            break
        response = handle_query(query, api_keys)
        print(response)
        speak(response)
if __name__ == '__main__':
    main()
```

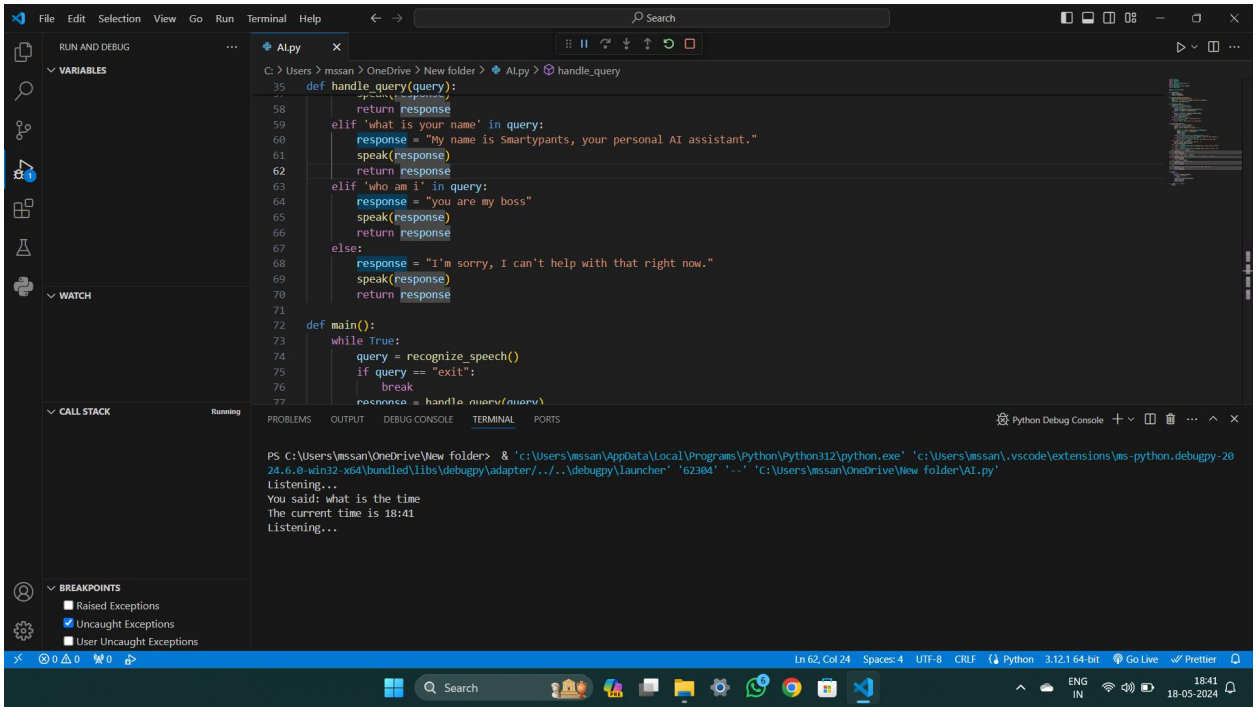
Project Hurdles:

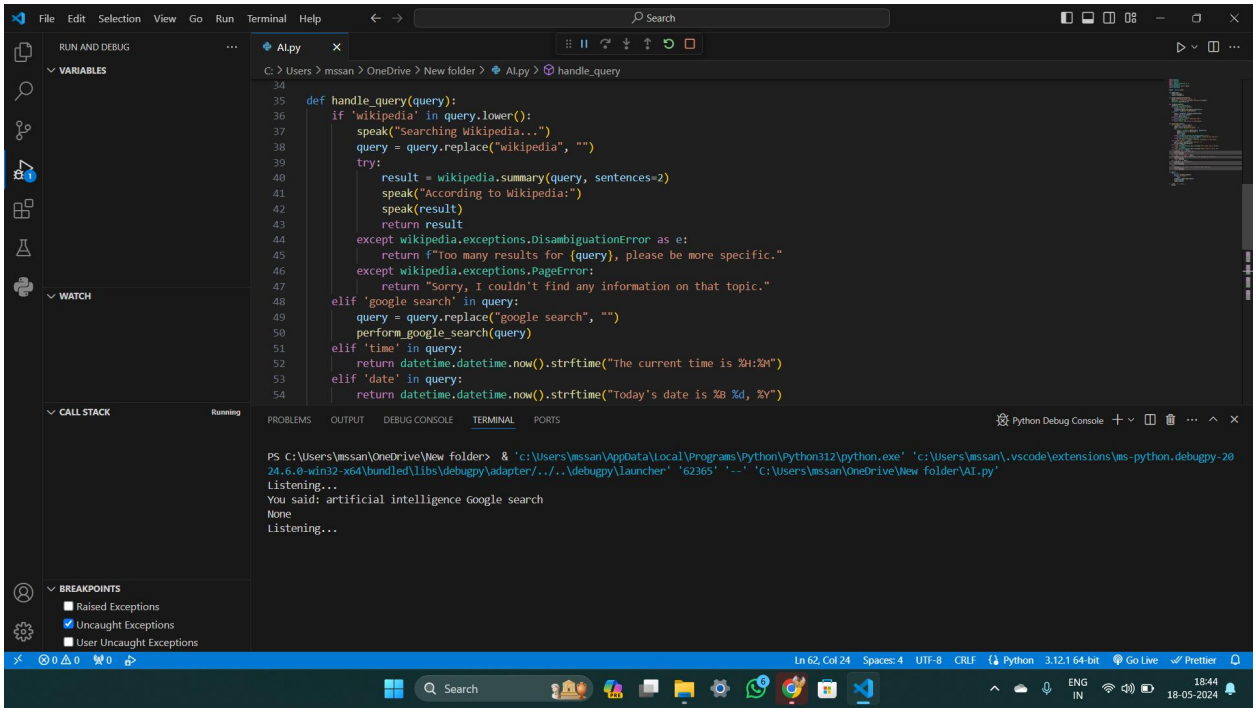
- Speech recognition accuracy: Ensure clear voice input and reduce background noise.
- Handling Wikipedia disambiguation errors: Prompt users to be more specific.
- Managing API request limits and errors: Implement rate limiting and error handling.
- Ensuring reliable internet connectivity: Make sure the system is connected to the internet.
- Improving user interaction and response clarity: Refine responses for better user experience.
- Addressing microphone and speaker compatibility issues: Ensure proper hardware setup and driver installation.
- Optimizing response time for user queries: Enhance processing speed and minimize delays.

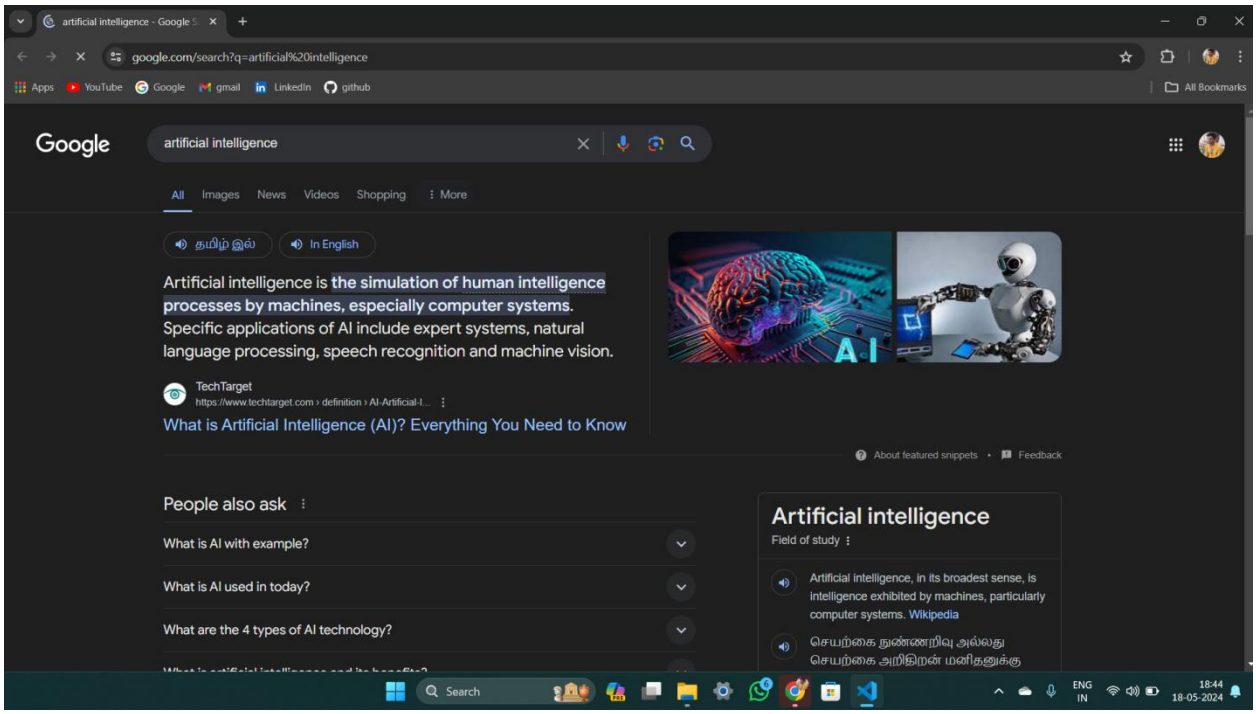
OUTPUT:

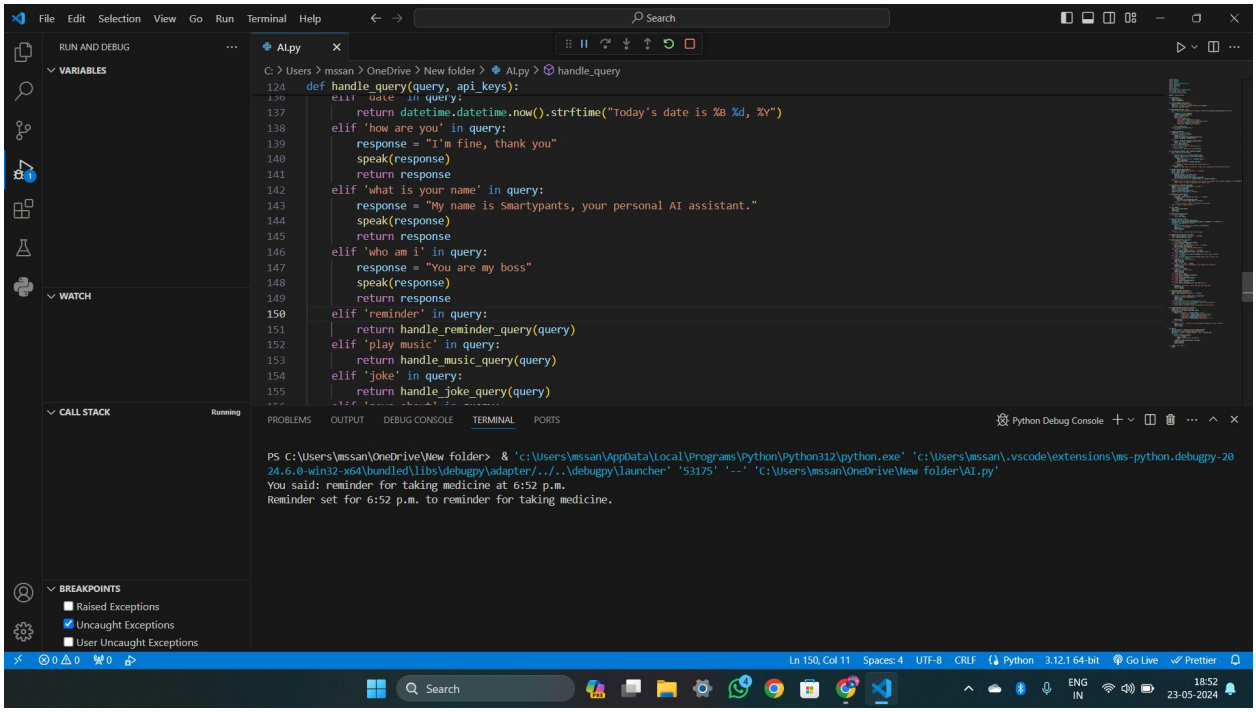


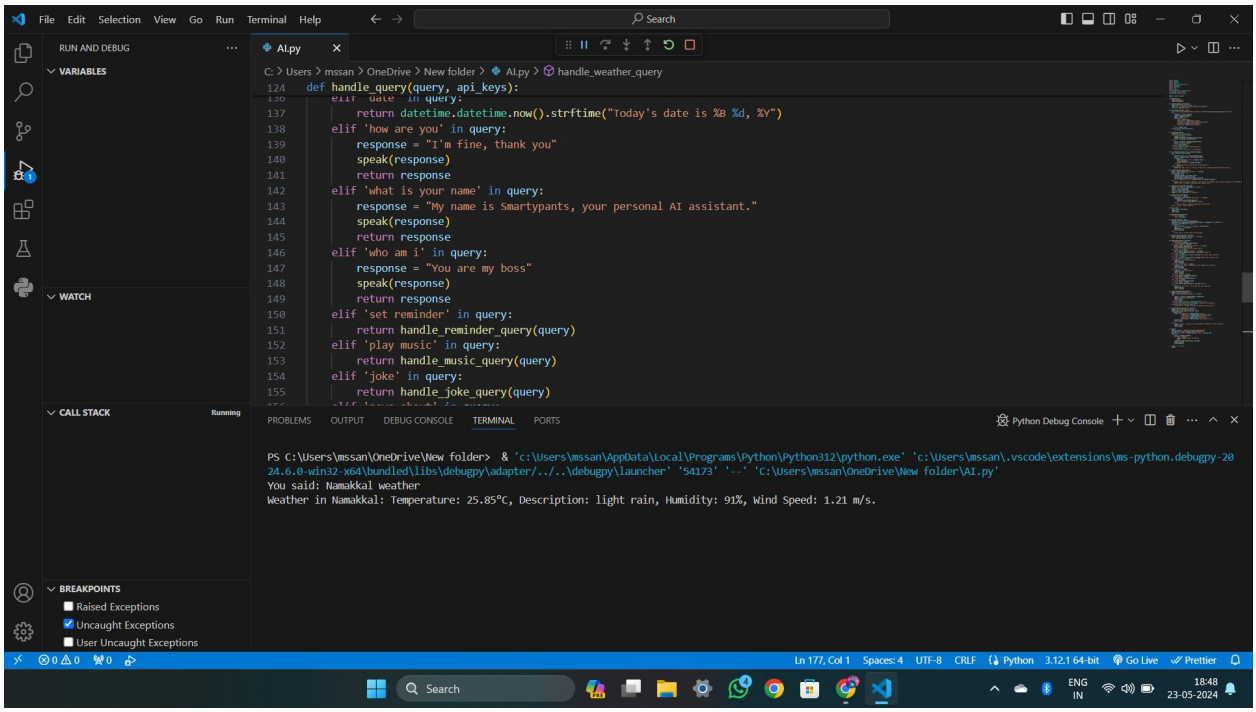

File Edit Selection View Go Run Terminal Help
Search
RUN AND DEBUG
VARIABLES
WATCH
CALL STACK
Running
BREAKPOINTS
Raised Exceptions
Uncaught Exceptions
User Uncaught Exceptions
AI.py
C: > Users > mssan > OneDrive > New folder > AI.py > handle_query
def handle_query(query):
elif 'google search' in query:
query = query.replace("google search", "")
perform_google_search(query)
elif 'time' in query:
return datetime.datetime.now().strftime("The current time is %H:%M")
elif 'date' in query:
return datetime.datetime.now().strftime("Today's date is %B %d, %Y")
elif 'how are you' in query:
response = "I'm fine,thank you"
speak(response)
return response
elif 'what is your name' in query:
response = "My name is Smartypants, your personal AI assistant."
speak(response)
return response
elif 'who am i' in query:
response = "you are my boss"
speak(response)
return response
else:
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python Debug Console
PS C:\Users\mssan\OneDrive\New folder> & 'c:\Users\mssan\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\mssan\.vscode\extensions\ms-python.debugpy-2024.6.0-win32-x64\bundled\libs\debugpy\adapter/../..\debugpy\launcher' '62325' '--' 'C:\Users\mssan\OneDrive\New folder\AI.py'
Listening...
You said: what is the date today
Today's date is May 18, 2024
Listening...
Ln 62, Col 24 Spaces: 4 UTF-8 CRLF Python 3.12.1 64-bit Go Live Prettier
ENG IN
18:41
18-05-2024

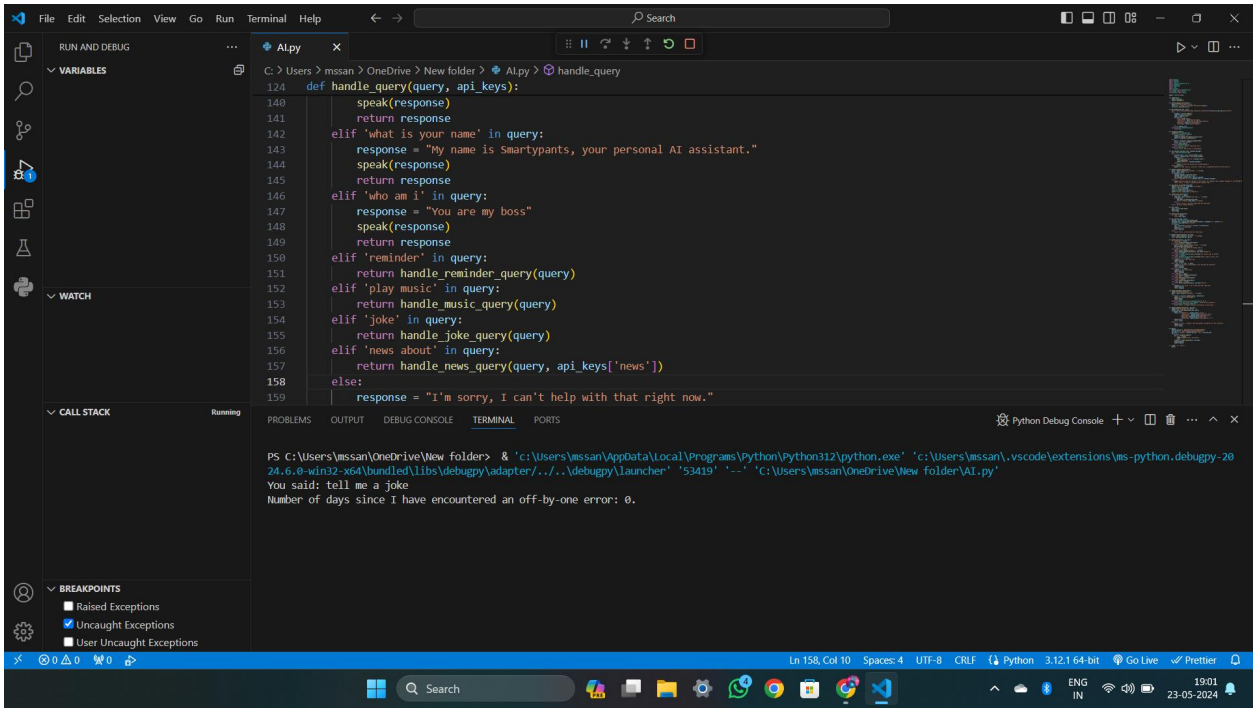

File Edit Selection View Go Run Terminal Help
Search
RUN AND DEBUG
VARIABLES
WATCH
CALL STACK
Running
BREAKPOINTS
Raised Exceptions
Uncaught Exceptions
User Uncaught Exceptions
AI.py
C: > Users > mssan > OneDrive > New folder > AI.py > handle_query
def handle_query(query):
return response
elif 'what is your name' in query:
response = "My name is Smartypants, your personal AI assistant."
speak(response)
return response
elif 'who am i' in query:
response = "you are my boss"
speak(response)
return response
else:
response = "I'm sorry, I can't help with that right now."
speak(response)
return response
def main():
while True:
query = recognize_speech()
if query == "exit":
break
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python Debug Console
PS C:\Users\mssan\OneDrive\New folder> & 'c:\Users\mssan\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\mssan\.vscode\extensions\ms-python.debugpy-2024.6.0-win32-x64\bundled\libs\debugpy\adapter/../..\debugpy\launcher' '62304' '--' 'C:\Users\mssan\OneDrive\New folder\AI.py'
Listening...
You said: what is the time
The current time is 18:41
Listening...
Ln 62, Col 24 Spaces: 4 UTF-8 CRLF Python 3.12.1 64-bit Go Live Prettier
ENG IN
18:41
18-05-2024

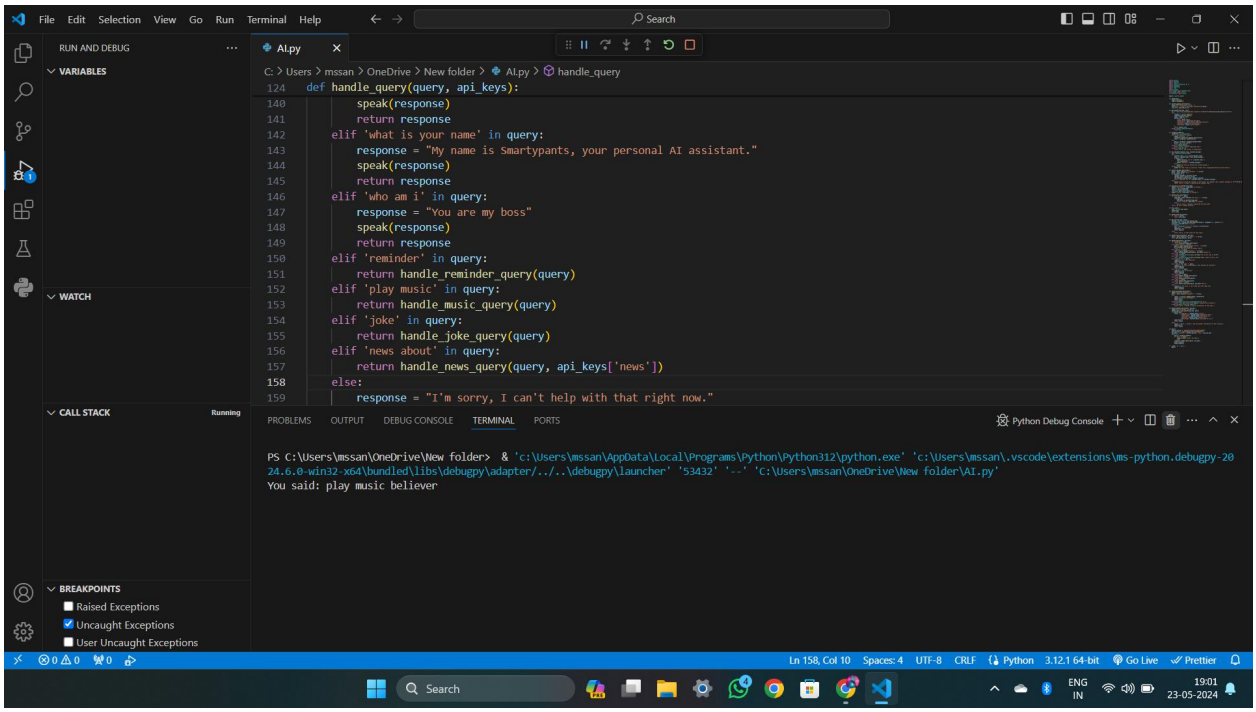

File Edit Selection View Go Run Terminal Help
Search
RUN AND DEBUG
VARIABLES
WATCH
CALL STACK
Running
BREAKPOINTS
Raised Exceptions
Uncaught Exceptions
User Uncaught Exceptions
AI.py
C: > Users > mssan > OneDrive > New folder > AI.py > handle_query
def handle_query(query):
if 'wikipedia' in query.lower():
speak("Searching Wikipedia...")
query = query.replace("wikipedia", "")
try:
result = wikipedia.summary(query, sentences=2)
speak("According to Wikipedia:")
speak(result)
return result
except wikipedia.exceptions.DisambiguationError as e:
return f"Too many results for {query}, please be more specific."
except wikipedia.exceptions.PageError:
return "Sorry, I couldn't find any information on that topic."
elif 'google search' in query:
query = query.replace("google search", "")
perform_google_search(query)
elif 'time' in query:
return datetime.datetime.now().strftime("The current time is %H:%M")
elif 'date' in query:
return datetime.datetime.now().strftime("Today's date is %B %d, %Y")
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python Debug Console
PS C:\Users\mssan\OneDrive\New folder> & 'c:\Users\mssan\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\mssan\.vscode\extensions\ms-python.debugpy-2024.6.0-win32-x64\bundled\libs\debugpy\adapter/../..\debugpy\launcher' '62365' '--' 'C:\Users\mssan\OneDrive\New folder\AI.py'
Listening...
You said: artificial intelligence Google search
None
Listening...
Ln 62, Col 24 Spaces: 4 UTF-8 CRLF Python 3.12.1 64-bit Go Live Prettier
Search
ENG IN
18:44
18-05-2024

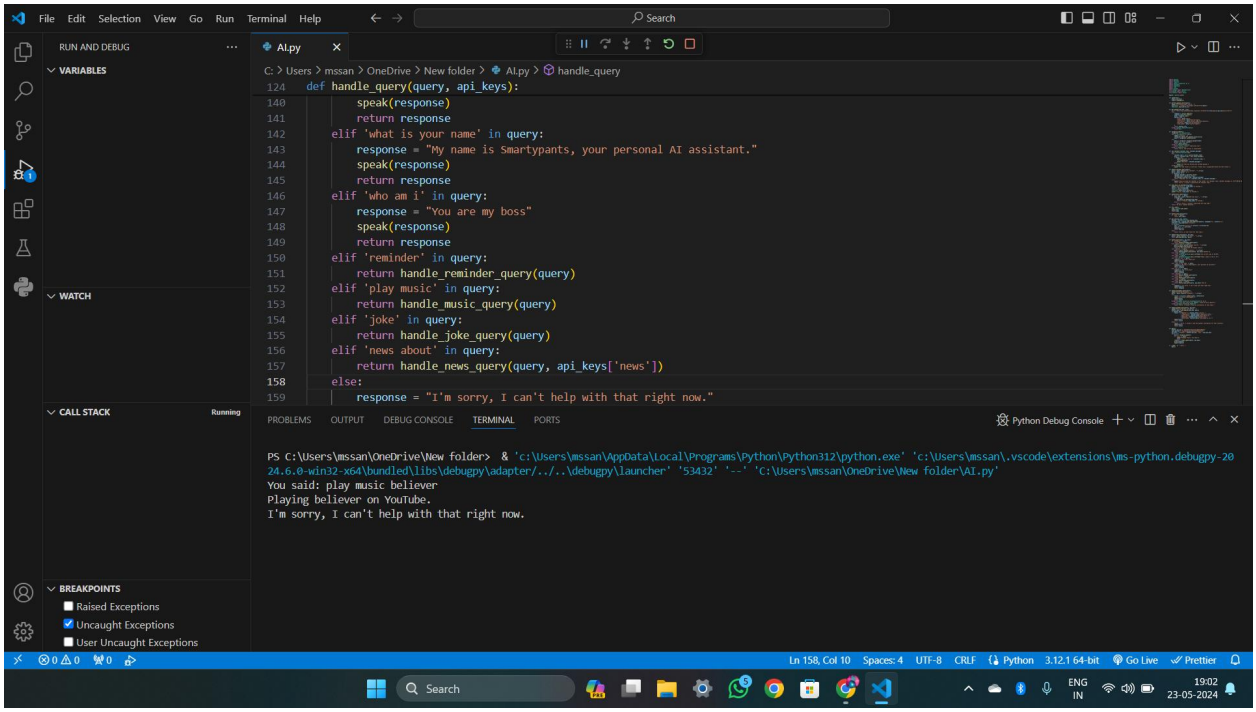

artificial intelligence - Google S
google.com/search?q=artificial%20intelligence
Apps YouTube Google gmail LinkedIn github
All Bookmarks
Google
artificial intelligence
All Images News Videos Shopping More
தமிழ் இல் In English
Artificial intelligence is the simulation of human intelligence processes by machines, especially computer systems. Specific applications of AI include expert systems, natural language processing, speech recognition and machine vision.
TechTarget
https://www.techtarget.com › definition › AI-Artificial-I...
What is Artificial Intelligence (AI)? Everything You Need to Know
About featured snippets • Feedback
People also ask
What is AI with example?
What is AI used in today?
What are the 4 types of AI technology?
Artificial intelligence
Field of study
Artificial intelligence, in its broadest sense, is intelligence exhibited by machines, particularly computer systems. Wikipedia
செயற்கை நுண்ணறிவு அல்லது செயற்கை அறிதிறன் மனிதனுக்கு
Search
ENG IN
18:44
18-05-2024

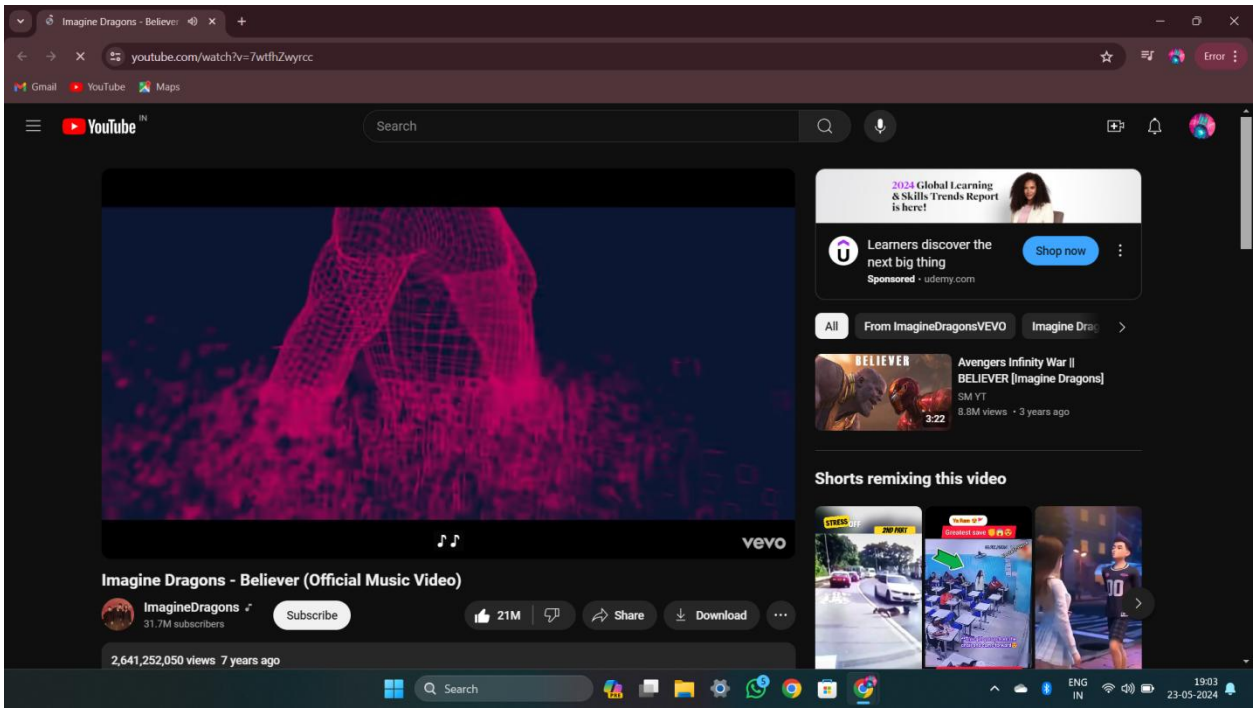

File Edit Selection View Go Run Terminal Help
Search
RUN AND DEBUG
VARIABLES
WATCH
CALL STACK
Running
BREAKPOINTS
Raised Exceptions
Uncaught Exceptions
User Uncaught Exceptions
AI.py
C: > Users > mssan > OneDrive > New folder > AI.py > handle_query
def handle_query(query, api_keys):
elif 'date' in query:
return datetime.datetime.now().strftime("Today's date is %B %d, %Y")
elif 'how are you' in query:
response = "I'm fine, thank you"
speak(response)
return response
elif 'what is your name' in query:
response = "My name is Smartypants, your personal AI assistant."
speak(response)
return response
elif 'who am i' in query:
response = "You are my boss"
speak(response)
return response
elif 'reminder' in query:
return handle_reminder_query(query)
elif 'play music' in query:
return handle_music_query(query)
elif 'joke' in query:
return handle_joke_query(query)
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python Debug Console
PS C:\Users\mssan\OneDrive\New folder> & 'c:\Users\mssan\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\mssan\.vscode\extensions\ms-python.debugpy-2024.6.0-win32-x64\bundled\libs\debugpy\adapter/../..\debugpy\launcher' '53175' '--' 'C:\Users\mssan\OneDrive\New folder\AI.py'
You said: reminder for taking medicine at 6:52 p.m.
Reminder set for 6:52 p.m. to reminder for taking medicine.
Ln 150, Col 11 Spaces: 4 UTF-8 CRLF Python 3.12.1 64-bit Go Live Prettier
Search
ENG IN
18:52
23-05-2024

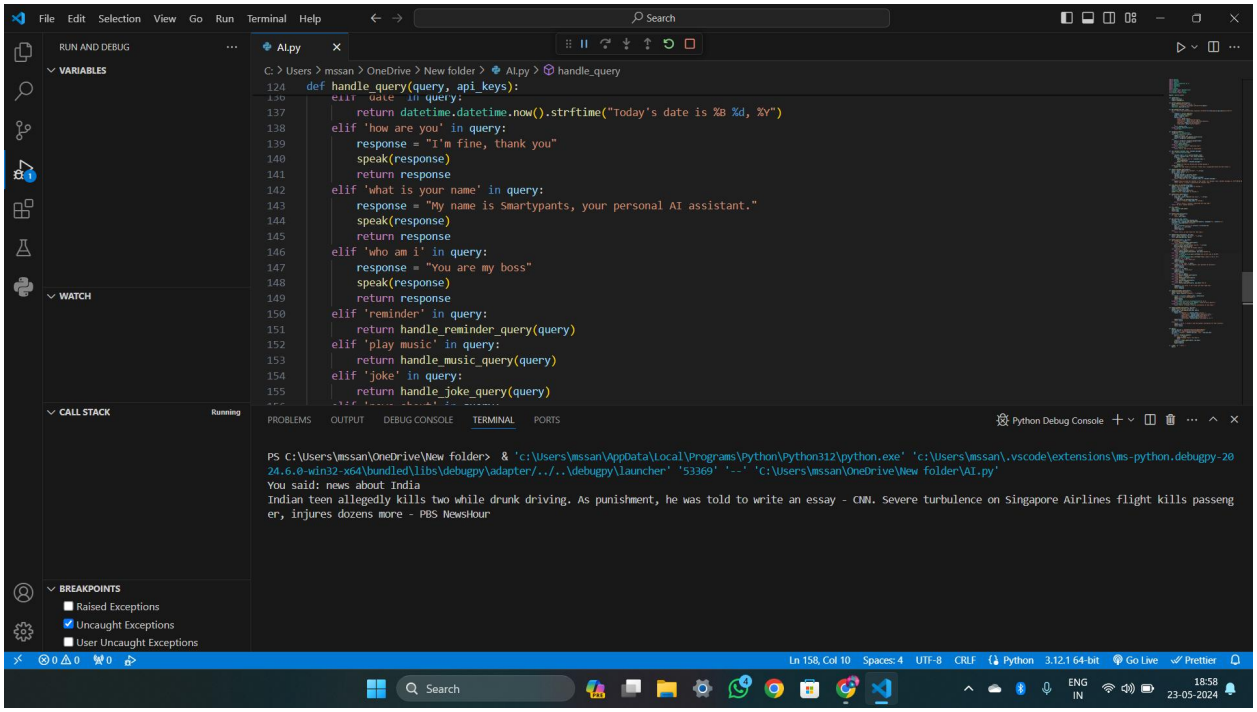

File Edit Selection View Go Run Terminal Help
Search
RUN AND DEBUG
VARIABLES
WATCH
CALL STACK
Running
BREAKPOINTS
Raised Exceptions
Uncaught Exceptions
User Uncaught Exceptions
AI.py
C: > Users > mssan > OneDrive > New folder > AI.py > handle_weather_query
def handle_query(query, api_keys):
elif 'date' in query:
return datetime.datetime.now().strftime("Today's date is %B %d, %Y")
elif 'how are you' in query:
response = "I'm fine, thank you"
speak(response)
return response
elif 'what is your name' in query:
response = "My name is Smartypants, your personal AI assistant."
speak(response)
return response
elif 'who am i' in query:
response = "You are my boss"
speak(response)
return response
elif 'set reminder' in query:
return handle_reminder_query(query)
elif 'play music' in query:
return handle_music_query(query)
elif 'joke' in query:
return handle_joke_query(query)
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python Debug Console
PS C:\Users\mssan\OneDrive\New folder> & 'c:\Users\mssan\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\mssan\.vscode\extensions\ms-python.debugpy-2024.6.0-win32-x64\bundled\libs\debugpy\adapter/../..\debugpy\launcher' '54173' '--' 'C:\Users\mssan\OneDrive\New folder\AI.py'
You said: Namakkal weather
Weather in Namakkal: Temperature: 25.85°C, Description: light rain, Humidity: 91%, Wind Speed: 1.21 m/s.
Ln 177, Col 1 Spaces: 4 UTF-8 CRLF Python 3.12.1 64-bit Go Live Prettier
Search
ENG IN
18:48
23-05-2024

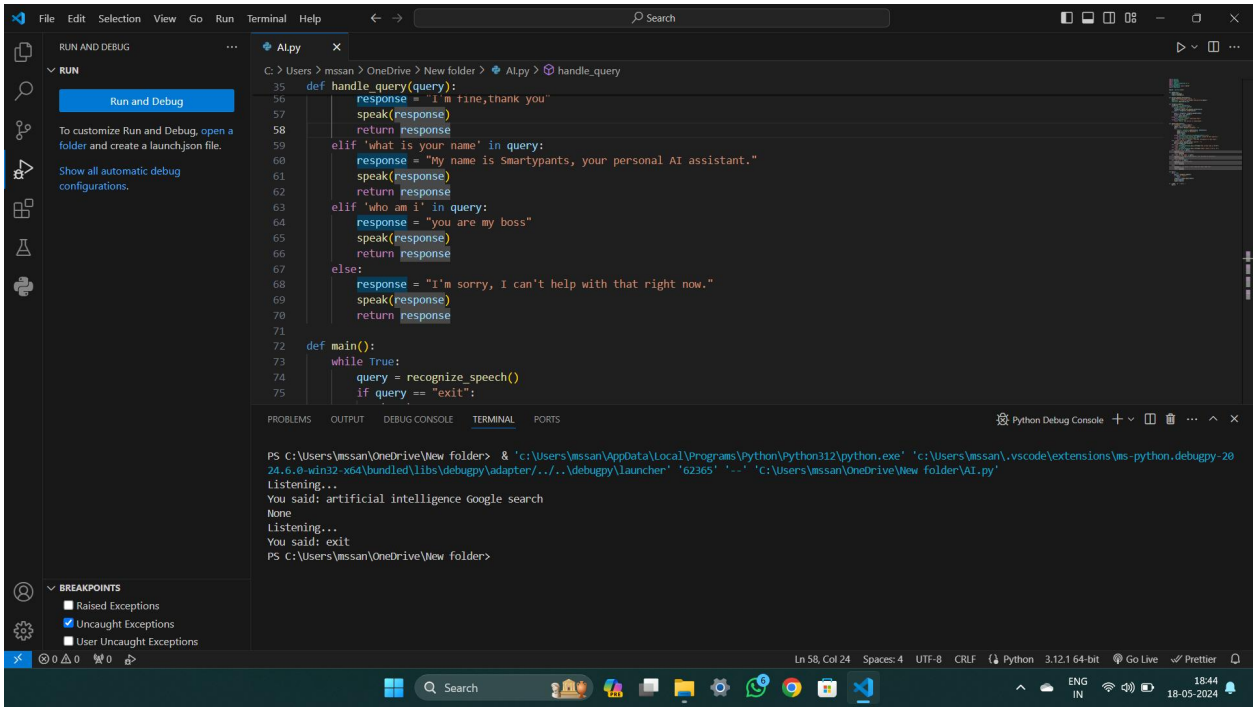

File Edit Selection View Go Run Terminal Help
Search
RUN AND DEBUG
VARIABLES
WATCH
CALL STACK
Running
BREAKPOINTS
Raised Exceptions
Uncaught Exceptions
User Uncaught Exceptions
AI.py
C: > Users > mssan > OneDrive > New folder > AI.py > handle_query
124 def handle_query(query, api_keys):
140 speak(response)
141 return response
142 elif 'what is your name' in query:
143 response = "My name is Smartypants, your personal AI assistant."
144 speak(response)
145 return response
146 elif 'who am i' in query:
147 response = "You are my boss"
148 speak(response)
149 return response
150 elif 'reminder' in query:
151 return handle_reminder_query(query)
152 elif 'play music' in query:
153 return handle_music_query(query)
154 elif 'joke' in query:
155 return handle_joke_query(query)
156 elif 'news about' in query:
157 return handle_news_query(query, api_keys['news'])
158 else:
159 response = "I'm sorry, I can't help with that right now."
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python Debug Console
PS C:\Users\mssan\OneDrive\New folder> & 'c:\Users\mssan\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\mssan\.vscode\extensions\ms-python.debugpy-2024.6.0-win32-x64\bundled\libs\debugpy\adapter/../..\debugpy\launcher' '53419' '--' 'C:\Users\mssan\OneDrive\New folder\AI.py'
You said: tell me a joke
Number of days since I have encountered an off-by-one error: 0.
Ln 158, Col 10 Spaces: 4 UTF-8 CRLF Python 3.12.1 64-bit Go Live Prettier
Search
ENG IN
19:01 23-05-2024


File Edit Selection View Go Run Terminal Help
Search
RUN AND DEBUG
VARIABLES
WATCH
CALL STACK
Running
BREAKPOINTS
Raised Exceptions
Uncaught Exceptions
User Uncaught Exceptions
AI.py
C: > Users > mssan > OneDrive > New folder > AI.py > handle_query
124 def handle_query(query, api_keys):
140 speak(response)
141 return response
142 elif 'what is your name' in query:
143 response = "My name is Smartypants, your personal AI assistant."
144 speak(response)
145 return response
146 elif 'who am i' in query:
147 response = "You are my boss"
148 speak(response)
149 return response
150 elif 'reminder' in query:
151 return handle_reminder_query(query)
152 elif 'play music' in query:
153 return handle_music_query(query)
154 elif 'joke' in query:
155 return handle_joke_query(query)
156 elif 'news about' in query:
157 return handle_news_query(query, api_keys['news'])
158 else:
159 response = "I'm sorry, I can't help with that right now."
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python Debug Console
PS C:\Users\mssan\OneDrive\New folder> & 'c:\Users\mssan\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\mssan\.vscode\extensions\ms-python.debugpy-2024.6.0-win32-x64\bundled\libs\debugpy\adapter/../..\debugpy\launcher' '53432' '--' 'C:\Users\mssan\OneDrive\New folder\AI.py'
You said: play music believer
Ln 158, Col 10 Spaces: 4 UTF-8 CRLF Python 3.12.1 64-bit Go Live Prettier
Search
ENG IN
19:01 23-05-2024


File Edit Selection View Go Run Terminal Help
Search
RUN AND DEBUG
VARIABLES
WATCH
CALL STACK
Running
BREAKPOINTS
Raised Exceptions
Uncaught Exceptions
User Uncaught Exceptions
AI.py
C: > Users > mssan > OneDrive > New folder > AI.py > handle_query
def handle_query(query, api_keys):
speak(response)
return response
elif 'what is your name' in query:
response = "My name is Smartypants, your personal AI assistant."
speak(response)
return response
elif 'who am i' in query:
response = "You are my boss"
speak(response)
return response
elif 'reminder' in query:
return handle_reminder_query(query)
elif 'play music' in query:
return handle_music_query(query)
elif 'joke' in query:
return handle_joke_query(query)
elif 'news about' in query:
return handle_news_query(query, api_keys['news'])
else:
response = "I'm sorry, I can't help with that right now."
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python Debug Console
PS C:\Users\mssan\OneDrive\New folder> & 'c:\Users\mssan\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\mssan\.vscode\extensions\ms-python.debugpy-2024.6.0-win32-x64\bundled\libs\debugpy\adapter/../..\debugpy\launcher' '53432' '--' 'C:\Users\mssan\OneDrive\New folder\AI.py'
You said: play music believer
Playing believer on YouTube.
I'm sorry, I can't help with that right now.
Ln 158, Col 10 Spaces: 4 UTF-8 CRLF Python 3.12.1 64-bit Go Live Prettier
Search
ENG IN
19:02
23-05-2024


Imagine Dragons - Believer
youtube.com/watch?v=7wtfhZwyrcc
Gmail YouTube Maps
YouTube IN
Search
2024 Global Learning & Skills Trends Report is here!
Learners discover the next big thing
Sponsored · udemy.com
Shop now
All
From ImagineDragonsVEVO
Imagine Dra
BELIEVER
3:22
Avengers Infinity War || BELIEVER [Imagine Dragons]
SM YT
8.8M views • 3 years ago
Shorts remixing this video
vevo
Imagine Dragons - Believer (Official Music Video)
ImagineDragons
31.7M subscribers
Subscribe
21M
Share
Download
2,641,252,050 views 7 years ago
Search
ENG IN
19:03
23-05-2024


File Edit Selection View Go Run Terminal Help
Search
RUN AND DEBUG
VARIABLES
WATCH
CALL STACK
Running
BREAKPOINTS
Raised Exceptions
Uncaught Exceptions
User Uncaught Exceptions
AI.py
C: > Users > mssan > OneDrive > New folder > AI.py > handle_query
def handle_query(query, api_keys):
elif 'date' in query:
return datetime.datetime.now().strftime("Today's date is %B %d, %Y")
elif 'how are you' in query:
response = "I'm fine, thank you"
speak(response)
return response
elif 'what is your name' in query:
response = "My name is Smartypants, your personal AI assistant."
speak(response)
return response
elif 'who am i' in query:
response = "You are my boss"
speak(response)
return response
elif 'reminder' in query:
return handle_reminder_query(query)
elif 'play music' in query:
return handle_music_query(query)
elif 'joke' in query:
return handle_joke_query(query)
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python Debug Console
PS C:\Users\mssan\OneDrive\New folder> & 'c:\Users\mssan\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\mssan\.vscode\extensions\ms-python.debugpy-2024.6.0-win32-x64\bundled\libs\debugpy\adapter/../..\debugpy\launcher' '53369' '--' 'C:\Users\mssan\OneDrive\New folder\AI.py'
You said: news about India
Indian teen allegedly kills two while drunk driving. As punishment, he was told to write an essay - CNN. Severe turbulence on Singapore Airlines flight kills passenger, injures dozens more - PBS NewsHour
Ln 158, Col 10 Spaces: 4 UTF-8 CRLF Python 3.12.1 64-bit Go Live Prettier
Search
ENG IN
18:58
23-05-2024


File Edit Selection View Go Run Terminal Help
Search
RUN AND DEBUG
RUN
Run and Debug
To customize Run and Debug, open a folder and create a launch.json file.
Show all automatic debug configurations.
BREAKPOINTS
Raised Exceptions
Uncaught Exceptions
User Uncaught Exceptions
AI.py
C: > Users > mssan > OneDrive > New folder > AI.py > handle_query
def handle_query(query):
response = "I'm fine,thank you"
speak(response)
return response
elif 'what is your name' in query:
response = "My name is Smartypants, your personal AI assistant."
speak(response)
return response
elif 'who am i' in query:
response = "you are my boss"
speak(response)
return response
else:
response = "I'm sorry, I can't help with that right now."
speak(response)
return response
def main():
while True:
query = recognize_speech()
if query == "exit":
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python Debug Console
PS C:\Users\mssan\OneDrive\New folder> & 'c:\Users\mssan\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\mssan\.vscode\extensions\ms-python.debugpy-2024.6.0-win32-x64\bundled\libs\debugpy\adapter/../..\debugpy\launcher' '62365' '--' 'C:\Users\mssan\OneDrive\New folder\AI.py'
Listening...
You said: artificial intelligence Google search
None
Listening...
You said: exit
PS C:\Users\mssan\OneDrive\New folder>
Ln 58, Col 24 Spaces: 4 UTF-8 CRLF Python 3.12.1 64-bit Go Live Prettier
Search
ENG IN
18:44
18-05-2024