import pyttsx3 #A text-to-speech conversion library in Python.

import speech\_recognition as sr #Used for speech-to-text conversion.

import datetime #To get the current time.

import webbrowser #To open URLs in a browser.

import os #Provides a way to interact with the operating system.

import comtypes.client #Used for COM interface.

import wikipedia #To fetch Wikipedia articles.

engine = pyttsx3.init('sapi5') # sapi5 TTS engine using SAPI5 (Microsoft's Speech API).

voices = engine.getProperty('voices')#Retrieves available voices.

engine.setProperty('voice', voices[0].id) #Sets the TTS engine to use the first available voice.

# Speak Function

def speak(audio):#speak(audio): Converts text (audio) to speech and plays it.

engine.say(audio)

engine.runAndWait()

#Greeting Function

def wishme():#wishme(): Greets the user based on the current hour. It then introduces itself.

hour = int(datetime.datetime.now().hour)

if hour >= 0 and hour < 12:

speak("Good Morning")

elif hour >= 12 and hour < 18:

speak("Good Afternoon")

else:

speak("Good Evening")

speak("I am Saran AI. Please tell me how may I help you.")

def takecommand():#takecommand(): Listens to the user's voice and converts it to text using Google's speech recognition.

r = sr.Recognizer()

with sr.Microphone() as source:

print("Listening...")

r.pause\_threshold = 1

audio = r.listen(source)

try:

print("Recognizing...")

query = r.recognize\_google(audio, language="en-in")

print(f"User said: {query}\n")

except Exception as e:

print("Say that again please...")

return "None"

return query

#Main Function

if \_\_name\_\_ == "\_\_main\_\_": #if \_\_name\_\_ == "\_\_main\_\_":: Ensures that wishme() is called only when the script is run directly.

wishme()

while True:

query = takecommand().lower()

if "wikipedia" in query:

speak('Searching Wikipedia...')

query = query.replace('wikipedia', "")

results = wikipedia.summary(query, sentences=2)

speak("According to Wikipedia")

print(results)

speak(results)

import webbrowser

elif "open google" in query:

webbrowser.open("https://google.com")

elif "open youtube" in query:

webbrowser.open("https://youtube.com")

elif "open github" in query:

webbrowser.open("https://github.com")

elif "open twitter" in query:

webbrowser.open("https://twitter.com")

elif "open facebook" in query:

webbrowser.open("https://facebook.com")

elif "open instagram" in query:

webbrowser.open("https://instagram.com")

elif "open reddit" in query:

webbrowser.open("https://reddit.com")

elif "open wikipedia" in query:

webbrowser.open("https://wikipedia.org")

elif "open amazon" in query:

webbrowser.open("https://amazon.com")

elif "open bing" in query:

webbrowser.open("https://bing.com")

elif "open yahoo" in query:

webbrowser.open("https://yahoo.com")

elif "open netflix" in query:

webbrowser.open("https://netflix.com")

elif "open spotify" in query:

webbrowser.open("https://spotify.com")

elif "open pinterest" in query:

webbrowser.open("https://pinterest.com")

elif "open quora" in query:

webbrowser.open("https://quora.com")

elif "open tumblr" in query:

webbrowser.open("https://tumblr.com")

elif "open ebay" in query:

webbrowser.open("https://ebay.com")

elif "open stackoverflow" in query:

webbrowser.open("https://stackoverflow.com")

elif "open imdb" in query:

webbrowser.open("https://imdb.com")

elif "open youtube" in query:

webbrowser.open("youtube.com")

elif "open google" in query:

webbrowser.open("google.com")

elif "open stackoverflow" in query:

webbrowser.open("stackoverflow.com")

elif "open linkedin" in query:

webbrowser.open("linkedin.com")

elif "the time" in query:

strTime = datetime.datetime.now().strftime("%H:%M:%S")

speak(f"Sir, the time is {strTime}")

#if \_\_name\_\_ == "\_\_main\_\_":: Ensures that wishme() is called only when the script is run directly.

'''while True:: Keeps the assistant running to continuously listen for commands.

query = takecommand().lower():: Takes the user's command and converts it to lowercase for easier comparison.

Various if statements:

"wikipedia": Searches Wikipedia and reads a summary.

"open youtube": Opens YouTube in the web browser.

"open google": Opens Google in the web browser.

"open stackoverflow": Opens Stack Overflow in the web browser.

"open linkedin": Opens LinkedIn in the web browser.

"the time": Tells the current time.'''