

Dharmsinh Desai University

Department of MCA

Semester - II

**Python Programming**

Termwork Report

Submitted by

|  |  |
| --- | --- |
| **Roll No** | **Name** |
| **MCA034** | **Aditya .S. Nair** |

**Python Termwork**

**Name :** Nair Aditya Sunil

**Roll :** MCA034

**Title :** Voice Based Assistant

**Problem Definition:**

An automated assistant to control your device by just saying the

commands.

**Outcome:**

Using voice commands user will be able to perform various operations.

**Commands:**

1. Open website

2. Time

3. Greet

4. Search the web

5. Play Music

6. Set Reminder

7. Send email

8. Search Wikipedia

9. Take Notes

10.Define a word

**Code**

# Importing packages

import warnings

import pyttsx3

import speech\_recognition as sr

import smtplib

import os

import datetime

import calendar

import random

import subprocess

import webbrowser

import wikipedia

import pywhatkit

from selenium import webdriver

warnings.filterwarnings("ignore")

# Making text to speech engine

engine = pyttsx3.init()

# Giving voice to our assistant

voices = engine.getProperty('voices') #getting details of current voice

engine.setProperty('voice', voices[0].id) #changing index, changes voices. 1 for female

# Function to make engine talk

def talk(audio):

engine.say(audio)

engine.runAndWait()

##

# Function to receive audio from the user

def get\_Audio():

record = sr.Recognizer()

# Using microphone to get audio input

with sr.Microphone() as source:

print("\n")

print("Listening : ")

audio = record.listen(source)

# Using google speech recognition to recognise the speech

data = " "

try:

data = record.recognize\_google(audio)

except sr.UnknownValueError:

talk("Unable To Understand")

print("Unable To Understand")

return " "

except sr.RequestError as e:

talk("Error From Google Speech Recognition")

print("Error From Google Speech Recognition")

return " "

print("Lucifer : "+data)

return data

##

def note(text):

date = datetime.datetime.now()

file\_name = str(date).replace(":", "-") + "-note.txt"

with open(file\_name, "w") as f:

f.write(text)

subprocess.Popen(["notepad.exe", file\_name])

##

def send\_email(to, content):

server = smtplib.SMTP("smtp.gmail.com", 587)

server.ehlo()

server.starttls()

# Enable low security in gmail

server.login("nairaditya2003@gmail.com", "zzjhccdkqageyzte")

server.sendmail("nairaditya2003@gmail.com", to, content)

server.close()

# Making wake word for the assistant

def call(text):

call\_assistant = "jarvis"

text = text.lower()

if call\_assistant in text:

return True

return False

##

def get\_today():

# Get current date time

now = datetime.datetime.now()

date\_now = datetime.datetime.today()

week\_now = calendar.day\_name[date\_now.weekday()]

month\_now = now.month

day\_now = now.day

months = ['January','February', 'March', 'April', 'May', 'June','July', 'August', 'September', 'October', 'November', 'December']

ordinals = ['1', '2', '3', '4', '5', '6', '7', '8', '9', '10','11', '12', '13', '14', '15', '16', '17', '18', '19', '20','21', '22', '23', '24', '25', '26', '27', '28', '29', '30','31']

return f'Today is {week\_now},{months[month\_now -1]} ,{ordinals[day\_now-1]}.'

##

def greet(text):

greetings = [

"hello",

"hi",

"hey",

"how are you?"

]

response = [

"Hello",

"Hi",

"Hey",

]

for word in text.lower().split():

if word in greetings:

return random.choice(response)

return ""

##

def wiki(text):

# Convert the input text to lowercase for case insensitivity

text\_lower = text.lower()

if "who is" in text\_lower:

# Find the index of "who is" in the lowercased text

index = text\_lower.find("who is")

if index != -1:

# Extract the term following "who is"

return text[index + len("who is"):].strip()

elif "what is" in text\_lower:

# Find the index of "what is" in the lowercased text

index = text\_lower.find("what is")

if index != -1:

# Extract the term following "what is"

return text[index + len("what is"):].strip()

return None # Return None if no valid query pattern is found

##

name = "Lucifer"

print("Jarvis : Hello "+name)

talk("Hello "+name)

while True:

try:

text = get\_Audio()

speak = ""

if call(text):

speak = speak + greet(text)

print("Jarvis : "+speak+ " " + name)

talk(speak + name)

elif ("date" in text) or ("day" in text) or ("month" in text):

today = get\_today()

speak = speak + today

print("Jarvis : "+speak)

talk(speak)

elif "time" in text:

current\_time = datetime.datetime.now()

meridiem = ""

if current\_time.hour >= 12:

meridiem = "p m"

hour = current\_time.hour - 12

else:

meridiem = "a m"

hour = current\_time.hour

minute = str(current\_time.minute)

speak = speak + "the time is " + str(hour) + " " + minute + " " + meridiem

print("Jarvis : "+speak)

talk(speak)

elif "wikipedia" in text or "Wikipedia" in text:

print("Jarvis : What do you want to search in wikipedia ?")

talk("What do you want to search in wikipedia ?")

search = get\_Audio()

if "who is" in text:

person = wiki(text)

info = wikipedia.summary(person,sentences=2)

speak = speak + info

print("Javis : "+speak)

talk(speak)

else:

person = wiki(search)

info = wikipedia.summary(person,sentences=2)

speak = speak + info

print("Jarvis : "+speak)

# talk(speak)

talk(speak)

elif "open" in text.lower():

if "chrome" in text.lower():

speak = speak + "Opening Google Chrome"

talk(speak)

os.startfile(

r"C:\Program Files\Google\Chrome\Application\chrome.exe"

)

elif "youtube" in text.lower():

speak = speak + "Opening Youtube\n"

talk("Opening Youtube")

print("Jarvis : Opening Youtube")

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

# Search

elif "google" in text.lower():

talk("What do you want to search in Google?")

print("Jarvis: What do you want to search in Google?")

search = get\_Audio()

print("Jarvis : Opening Google")

talk("Opening Google")

search\_query\_encoded = "+".join(search.split())

search\_url = f"https://www.google.com/search?q={search\_query\_encoded}"

webbrowser.open(search\_url)

print(f"Jarvis : Opening {search} in Google.")

talk(f"Opening {search} in Google")

elif "word" in text.lower():

speak = speak + "Opening Microsoft Word"

print("Jarvis : " +speak)

talk(speak)

os.startfile(

r"C:\Program Files\Microsoft Office\root\Office16\WINWORD.exe"

)

elif "excel" in text.lower():

speak = speak + "Opening Microsoft Excel"

print("Jarvis : " +speak)

talk(speak)

os.startfile(

r"C:\Program Files\Microsoft Office\root\Office16\EXCEL.EXE"

)

elif "powerpoint" in text.lower():

speak = speak + "Opening Microsoft PowerPoint"

print("Jarvis : " +speak)

talk(speak)

os.startfile(

r"C:\Program Files\Microsoft Office\root\Office16\POWERPNT.EXE"

)

elif "email" in text or "gmail" in text or "mail" in text:

try:

talk("What should I say")

print("Jarvis : What should I say ?")

content = get\_Audio()

talk("Enter Receivers Email")

to = input("Jarvis : Enter Receivers Email : ")

send\_email(to,content)

speak = speak + "Email Has Been Sent"

print("Jarvis : Email Has Been Sent")

talk(speak)

except Exception as e:

print(e)

print("Jarvis : I am not able to sent this email")

talk("I am not able to sent this email")

elif "set reminder" in text.lower():

print("Jarvis : What do you want me to remember ? ")

talk("What do you want me to remember?")

msg = get\_Audio()

talk("You asked me to remind you that : " +msg)

print("Jarvis : You asked me to remind you that : " +msg)

rem\_file = open('reminder.txt','a')

rem\_file.write(msg)

rem\_file.close()

print("Jarvis : Reminder Saved")

talk("Reminder Saved")

elif "show reminders" in text.lower():

with open('reminder.txt', 'r') as rem\_file:

print("Jarvis: These Are Your Reminders")

reminders = rem\_file.read()

print(reminders)

talk("These Are Your Reminders: " + reminders)

elif "play music" in text.lower():

print("Jarvis : Which song you want me to play?")

talk("Which Song You Want me to play")

song = get\_Audio()

print("Jarvis : Playing " + song)

talk("Playing " + song)

pywhatkit.playonyt(song)

elif "take notes" in text.lower():

print("Jarvis : Tell Me ")

talk("Tell Me")

msg = get\_Audio()

notes\_file = open('notes.txt','a')

notes\_file.write(msg)

notes\_file.close()

print("Jarvis : Notes Saved")

talk("Notes Saved")

elif "show notes" in text.lower():

print("Jarvis : Here Are Your Notes ")

talk("Here Are Your Notes")

notes\_file = open('notes.txt','r')

notes = notes\_file.read()

print(notes)

notes\_file.close()

# Exit the program

elif "exit" in text.lower():

print("Jarvis : Goodbye!")

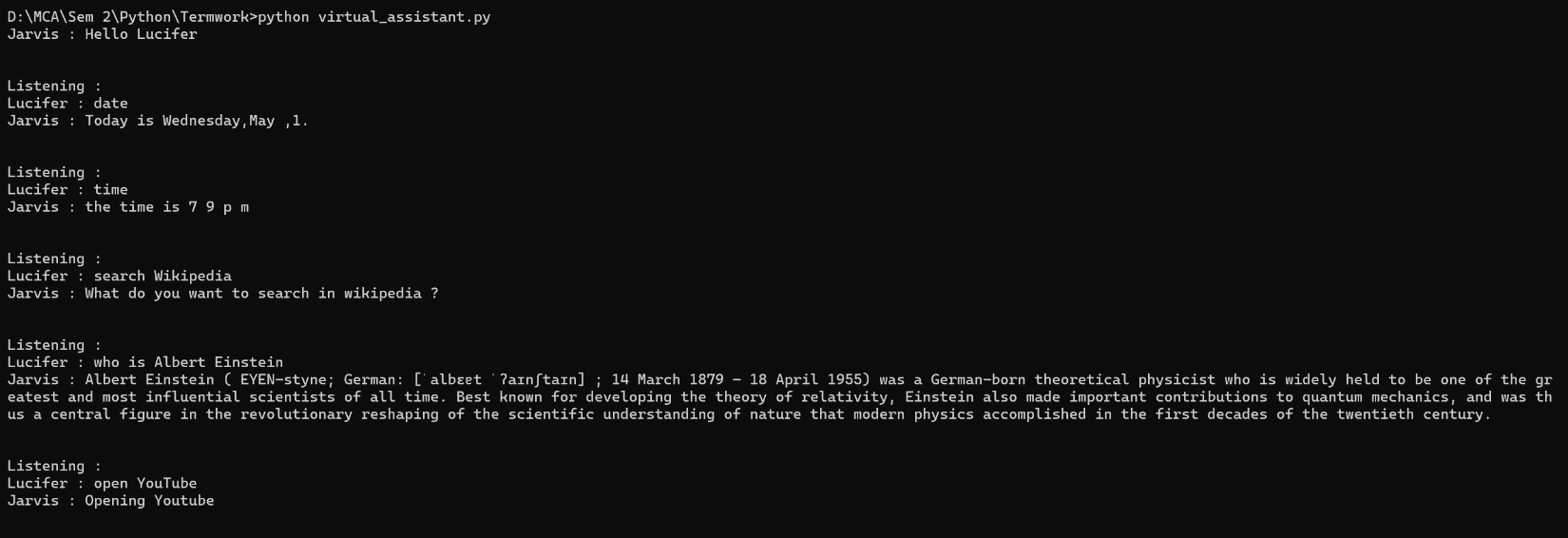
talk("Goodbye!")

break

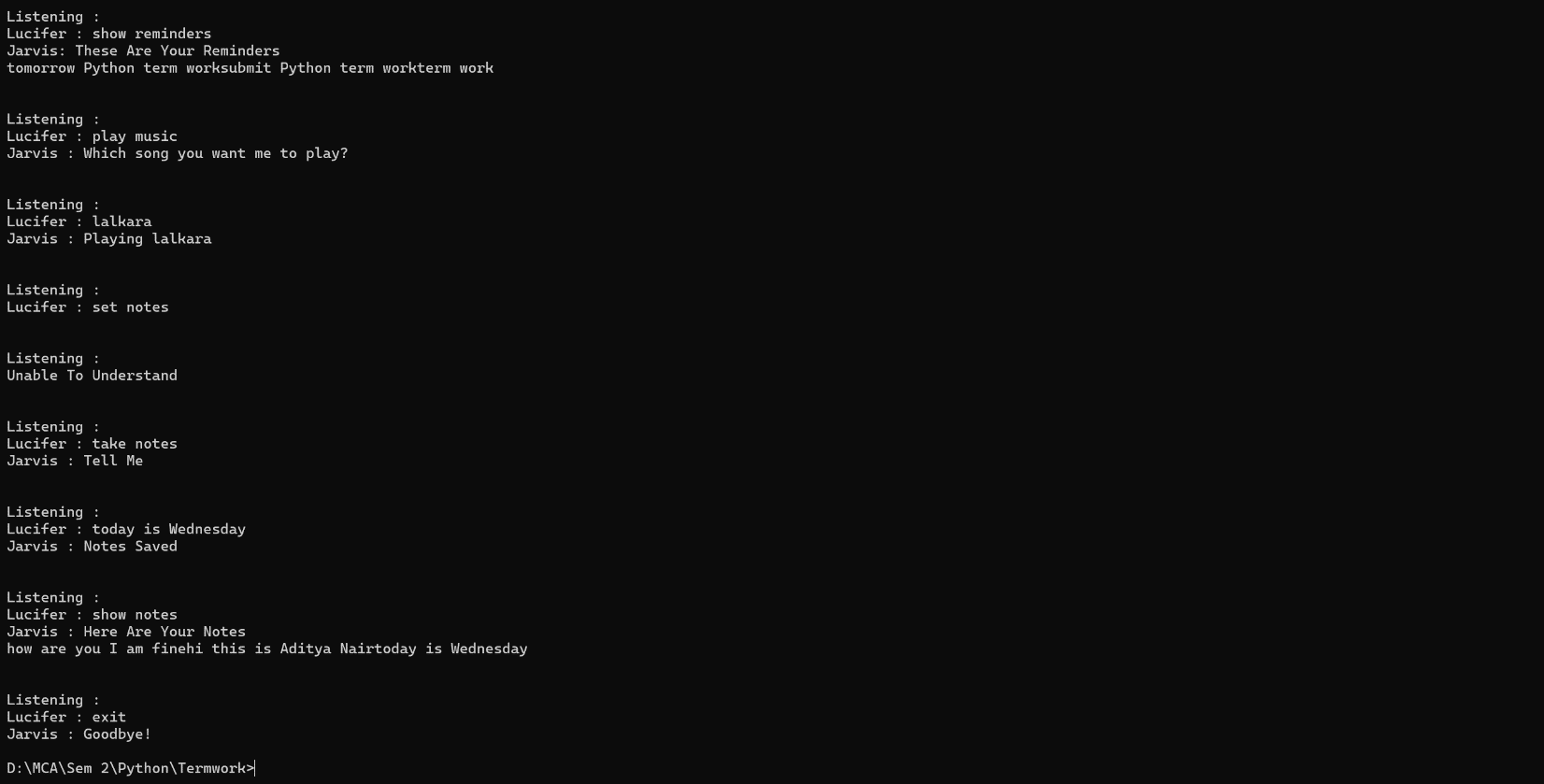
except:

talk("I don't know that")

**Output**

****

****

****