Project Title

Voice Assistant
using python

By
Maximus. R
Department of Information Technology,
Panimalar Institute of Technology

Overview:

- Project Statement & Definition
- Requirements
- Evaluation & Training

01

Project Statement & Definition

02

Requirements

03

Evaluation & Training

Execution & Output

Project Statement & Definition : *01*

Voice assistants are AI-driven digital platforms that understand and respond to spoken commands or questions. They utilize natural language processing (NLP) and machine learning algorithms to interpret user inputs and execute tasks.

Common examples include Amazon's Alexa, Apple's Siri, Google Assistant, and Microsoft's Cortana. Voice assistants can perform various functions such as setting reminders, answering questions, controlling smart home devices, providing weather updates, playing music, and more.

Project Statement & Definition : *02*

They continue to evolve with advancements in AI, offering increasingly personalized and contextually relevant responses to enhance user experiences in diverse applications from smartphones to smart speakers and beyond.

01

Project
Statement
&
Definition

02

Requirements

03

Evaluation
&
Training

Execution & Output

Requirements : *01*

Libraries

subprocess, wolframalpha, pyttsx3, tkinter, json, operator, speech_recognition, datetime, wikipedia, webbrowser, os, winshell, pyjokes, feedparser, smtplib, ctypes, time, requests, shutil, twilio, clint, ecapture, bs4, and win32com.

Applications

Python (*Installed in the Local Directory*)
Browsers(*Preferable: Chrome, Opera*)

Configuration

Email (*Change in the Code*)
File Path(*For Music, Documents, etc*)

01

Project Statement & Definition

02

Requirements

03

Evaluation & Training

Execution & Output

Evaluation & Training : *01*

Integration

It integrates with external services like WolframAlpha, Wikipedia, Twilio, and OpenWeatherMap, enhancing its capabilities to provide accurate information and perform various tasks.

Functionality

The code offers a wide range of functionalities such as opening web pages, searching Wikipedia, sending emails, retrieving news, and performing system operations like shutdowns and hibernations.

User Interaction

The voice assistant engages in dialogue with users, greeting them, asking for their name, and responding to queries in a conversational manner.

Customization

Users can customize certain aspects like the assistant's name and background.

Evaluation & Training : *02*

Integration

It integrates with external services like WolframAlpha, Wikipedia, Twilio, and OpenWeatherMap, enhancing its capabilities to provide accurate information and perform various tasks.

User Interaction

The voice assistant engages in dialogue with users, greeting them, asking for their name, and responding to queries in a conversational manner.

Functionality

The code offers a wide range of functionalities such as opening web pages, searching Wikipedia, sending emails, retrieving news, and performing system operations like shutdowns and hibernations.

Customization

Users can customize certain aspects like the assistant's name and background.

Evaluation & Training : *03*

Integration

It integrates with external services like WolframAlpha, Wikipedia, Twilio, and OpenWeatherMap, enhancing its capabilities to provide accurate information and perform various tasks.

Functionality

The code offers a wide range of functionalities such as opening web pages, searching Wikipedia, sending emails, retrieving news, and performing system operations like shutdowns and hibernations.

User Interaction

The voice assistant engages in dialogue with users, greeting them, asking for their name, and responding to queries in a conversational manner.

Customization

Users can customize certain aspects like the assistant's name and background.

Evaluation & Training : *04*

Integration

It integrates with external services like WolframAlpha, Wikipedia, Twilio, and OpenWeatherMap, enhancing its capabilities to provide accurate information and perform various tasks.

Functionality

The code offers a wide range of functionalities such as opening web pages, searching Wikipedia, sending emails, retrieving news, and performing system operations like shutdowns and hibernations.

User Interaction

The voice assistant engages in dialogue with users, greeting them, asking for their name, and responding to queries in a conversational manner.

Customization

Users can customize certain aspects like the assistant's name and background.

Evaluation & Training : *05*

Speech Recognition

Experiment with different engines for better accuracy.

NLU Enhancement

Implement advanced NLP techniques for understanding user intent.

Task Expansion

Add support for new tasks and integrate with more APIs.

Performance Optimisation

Profile and optimize critical sections for better performance.

01

Project
Statement
&
Definition

02

Requirements

03

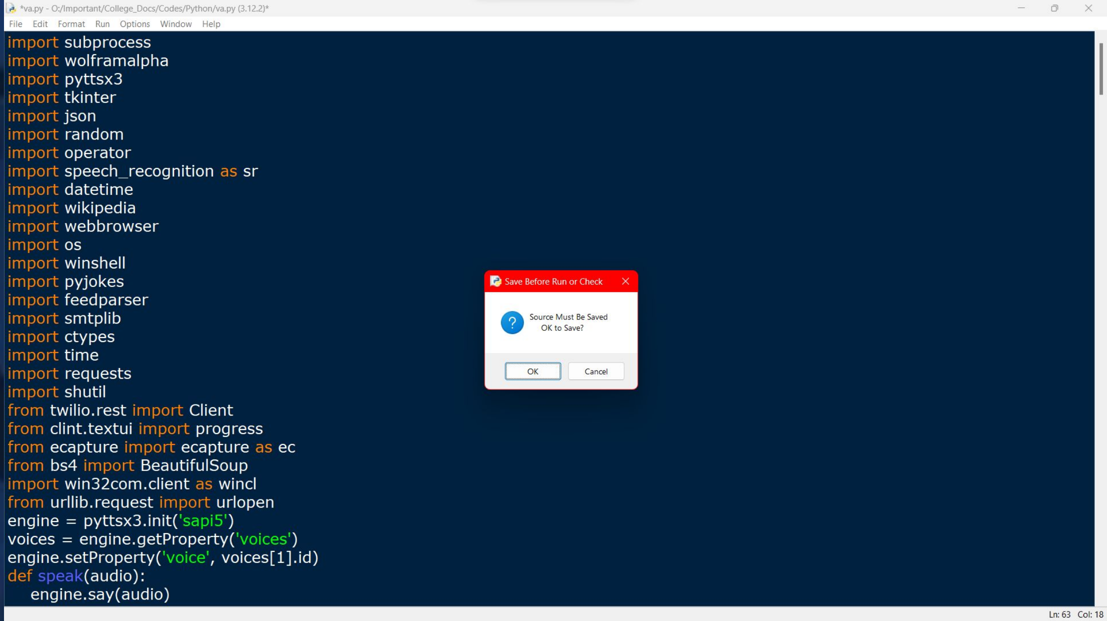
Evaluation
&
Training

Execution & Output

Execution & Output : *01*

Execution

Use the latest version of Python after installing all the required modules



va.py - O:/Important/College_Docs/Codes/Python/va.py (3.12.2)

```python
import subprocess
import wolframalpha
import pyttsx3
import tkinter
import json
import random
import operator
import speech_recognition as sr
import datetime
import wikipedia
import webbrowser
import os
import winshell
import pyjokes
import feedparser
import smtplib
import ctypes
import time
import requests
import shutil
from twilio.rest import Client
from clint.textui import progress
from ecapture import ecapture as ec
from bs4 import BeautifulSoup
import win32com.client as wincl
from urllib.request import urlopen
engine = pyttsx3.init('sapi5')
voices = engine.getProperty('voices')
engine.setProperty('voice', voices[1].id)
def speak(audio):
    engine.say(audio)
```

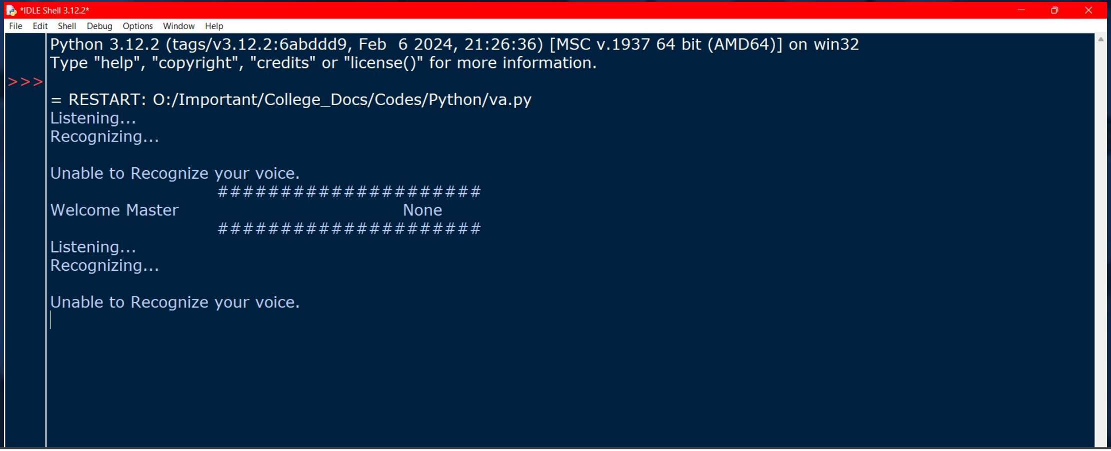
Save Before Run or Check

Source Must Be Saved
OK to Save?

Execution & Output : *02*

Output

The VA requires a microphone. Make sure to have it connected.



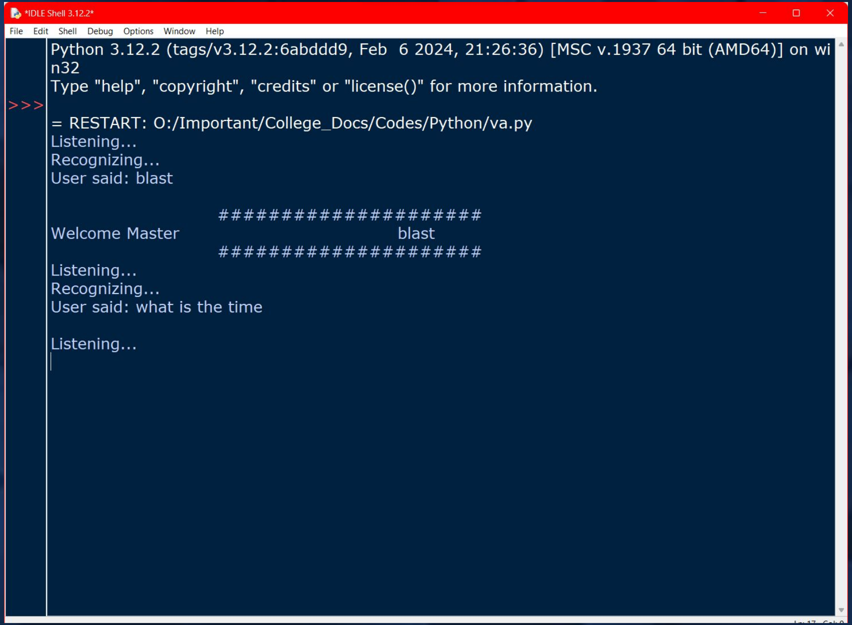
```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: O:/Important/College_Docs/Codes/Python/va.py
Listening...
Recognizing...

Unable to Recognize your voice.
                        ####################
Welcome Master                      None
                        ####################
Listening...
Recognizing...

Unable to Recognize your voice.
```

This image is to show, if there is no microphone, there will be no response

Execution & Output : *03*

Output

The output should look like this.



```
Python 3.12.2 (tags/v3.12.2:6abddd9, Feb  6 2024, 21:26:36) [MSC v.1937 64 bit (AMD64)] on wi
n32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: O:/Important/College_Docs/Codes/Python/va.py
Listening...
Recognizing...
User said: blast


                        #####################
Welcome Master                                  blast
                        #####################
Listening...
Recognizing...
User said: what is the time

Listening...
```

Conclusion

The provided voice assistant code offers a foundational framework for building a versatile and interactive assistant system. While it demonstrates key functionalities such as speech recognition, natural language understanding, and task execution, there are areas for improvement in terms of code structure, error handling, security, and user experience.

By iteratively refining the code, enhancing its capabilities, and addressing user feedback, the voice assistant can evolve into a more robust, efficient, and user-friendly tool that meets the diverse needs of its users. With continuous training and iteration, the voice assistant has the potential to become a valuable asset in simplifying daily tasks and enhancing productivity.