

PROJECT REPORT:VOICE ASSISTANT IN PYTHON

1. Introduction

This project implements a voice assistant using Python. It uses speech recognition to listen for commands, text-to-speech for replies, and integrates additional libraries for tasks like playing YouTube videos, fetching Wikipedia summaries, and telling jokes.

2. Objectives

- Build an interactive voice assistant
- Enable voice-based commands
- Use APIs and libraries to perform tasks like information retrieval and media playback

3. System Requirements

- Python 3.x
- Internet connection
- Microphone
- Libraries: speech_recognition, pyttsx3, pywhatkit, wikipedia, pyjokes

4. Conclusion

This project demonstrates a functional voice assistant using Python. The assistant can be expanded further with more features such as smart home integration, reminders, or offline speech models.

FRIDAY PYTHON CODE

```
import speech_recognition as sr
import pyttsx3
import pywhatkit
import wikipedia
import pyjokes
```

```
import datetime
import pytz

listener = sr.Recognizer()
engine = pyttsx3.init()

voices = engine.getProperty('voices')

try:
    engine.setProperty('voice', voices[1].id)
except IndexError:
    engine.setProperty('voice', voices[0].id)

def talk(text):
    """Converts text to speech and prints it to the console."""
    print("Friday:", text)
    engine.say(text)
    engine.runAndWait()

def take_command():
    """Listens to user's voice, recognizes the command using Google API, and processes it."""
    command = ""
    try:
        with sr.Microphone() as source:
            print("Listening...")
            listener.adjust_for_ambient_noise(source)
            voice = listener.listen(source)

        command = listener.recognize_google(voice)

        command = command.lower()
        if 'Friday' in command:
            command = command.replace('Friday', "").strip()
        print("You said:", command)

    except sr.UnknownValueError:
        print("Sorry, I didn't understand that.")
        talk("Sorry, I didn't catch that. Could you please repeat?")
        command = ""

    except sr.RequestError:
        print("Network error. Check your internet connection.")
```

```
talk("I seem to have a network problem. Please check your internet connection.")
command = ""
except Exception as e:
    print(f"An unexpected error occurred: {e}")
    command = ""

return command

def run_Friday():
    """Processes the recognized command and responds accordingly."""
    command = take_command()
    if not command:
        return

    if 'play' in command:
        song = command.replace('play', "").strip()
        talk("Playing " + song)
        pywhatkit.playonyt(song)

    elif 'time' in command:
        IST = pytz.timezone('Asia/Kolkata')

        now = datetime.datetime.now(IST)
        time = now.strftime("%I:%M %p")
        talk(f"The current time in India is {time}")

    elif 'date' in command:
        today = datetime.date.today().strftime('%A, %B %d, %Y')
        talk(f"Today is {today}")

    elif 'who is' in command or 'who the heck is' in command:
        person = command.replace('who the heck is', "").replace('who is', "").strip()
        try:
            info = wikipedia.summary(person, sentences=1, auto_suggest=False, redirect=True)
            print(info)
            talk(info)
        except wikipedia.exceptions.PageError:
            talk(f"Sorry, I couldn't find any information on {person}")
        except wikipedia.exceptions.DisambiguationError:
            talk(f"The term {person} is ambiguous. Please be more specific.")
```

```
elif 'what is' in command or 'tell me about' in command or 'search for' in command:
```

```
    query = command.replace('what is', "").replace('tell me about', "").replace('search for', "")  
    .strip()
```

```
    if query:  
        talk(f"Searching the web for {query}")
```

```
        pywhatkit.search(query)
```

```
    else:  
        talk("What would you like me to search for?")
```

```
elif 'are you single' in command or 'married' in command:
```

```
    talk("I am in a relationship with Wi-Fi.")
```

```
elif 'joke' in command:
```

```
    talk(pyjokes.get_joke())
```

```
else:
```

```
    talk("I'm sorry, I am not programmed to handle that specific command yet.")
```

```
while True:
```

```
    run_Friday()
```

```
1 import speech_recognition as sr  
2 import pyttsx3  
3 import pywhatkit  
4 import wikipedia  
5 import pyjokes  
6 import datetime  
7 import pytz  
8  
9  
10 listener = sr.Recognizer()  
11 engine = pyttsx3.init()  
12  
13  
14 voices = engine.getProperty('voices')  
15  
16 try:  
17     engine.setProperty('voice', voices[1].id)  
18 except IndexError:  
19     engine.setProperty('voice', voices[0].id)  
20  
21  
22  
23 def talk(text):  
24     """Converts text to speech and prints it to the console."""  
25     print("Friday:", text)  
26     engine.say(text)  
27     engine.runAndWait()
```

```

29     def take_command():
30         """Listens to user's voice, recognizes the command using Google API, and processes it."""
31         command = ""
32         try:
33             with sr.Microphone() as source:
34                 print("Listening...")
35                 listener.adjust_for_ambient_noise(source)
36                 voice = listener.listen(source)
37
38
39                 command = listener.recognize_google(voice)
40
41                 command = command.lower()
42                 if 'Friday' in command:
43                     command = command.replace('Friday', '').strip()
44                     print("You said:", command)
45
46         except sr.UnknownValueError:
47
48             print("Sorry, I didn't understand that.")
49             talk("Sorry, I didn't catch that. Could you please repeat?")
50             command = ""
51         except sr.RequestError:
52
53             print("Network error. Check your internet connection.")
54             talk("I seem to have a network problem. Please check your internet connection.")
55             command = ""
56         except Exception as e:
57
58             print(f"An unexpected error occurred: {e}")
59             command = ""
60
61     return command
62
63 def run_Friday():
64     """Processes the recognized command and responds accordingly."""
65     command = take_command()
66     if not command:
67         return
68
69
70     if 'play' in command:
71         song = command.replace('play', '').strip()
72         talk("Playing " + song)
73         pywhatkit.playonyt(song)
74
75
76     elif 'time' in command:
77
78         IST = pytz.timezone('Asia/Kolkata')
79
80         now = datetime.datetime.now(IST)
81         time = now.strftime('%I:%M %p')
82         talk(f"The current time in India is {time}")
83
84

```

```

85     elif 'date' in command:
86         today = datetime.date.today().strftime('%A, %B %d, %Y')
87         talk(f"Today is {today}")
88
89
90     elif 'who is' in command or 'who the heck is' in command:
91         person = command.replace('who the heck is', '').replace('who is', '').strip()
92         try:
93
94             info = wikipedia.summary(person, sentences=1, auto_suggest=False, redirect=True)
95             print(info)
96             talk(info)
97         except wikipedia.exceptions.PageError:
98             talk(f"Sorry, I couldn't find any information on {person}")
99         except wikipedia.exceptions.DisambiguationError:
100             talk(f"The term {person} is ambiguous. Please be more specific.")
101
102
103    elif 'what is' in command or 'tell me about' in command or 'search for' in command:
104
105        query = command.replace('what is', '').replace('tell me about', '').replace('search for', '').strip()
106
107        if query:
108            talk(f"Searching the web for {query}")
109
110            pywhatkit.search(query)
111        else:
112            talk("What would you like me to search for?")
113

```

```

115    elif 'are you single' in command or 'married' in command:
116        talk("I am in a relationship with Wi-Fi.")
117    elif 'joke' in command:
118        talk(pyjokes.get_joke())
119
120
121    else:
122        talk("I'm sorry, I am not programmed to handle that specific command yet.")
123
124
125 while True:
126     run_Friday()
127
128
129

```



