**PROJECT RECORD**

**PROJECT TITLE : ALEXA USING PYTHON**

**PRESENTED BY : ATCHAYA S**

**GUIDED BY : MISS. P.MONISHA**

**DATE : 02-04-2025**

**Description:**

This is a Python-based AI Voice Assistant that uses speech recognition, text-to-speech (TTS), and Google's SerpAPI to fetch real-time answers. It listens to voice commands, processes queries, searches Google, and speaks the results aloud. It supports functionalities like playing YouTube videos, fetching current time, and answering general knowledge questions. Unlike traditional assistants that open a browser, this one retrieves and speaks answers directly in Python IDLE.

**PROJECT CODE:**

**1. Importing Required Libraries**

import speech\_recognition as sr

import pyttsx3

import pywhatkit

import datetime

import requests

 **speech\_recognition** – Captures and converts spoken words into text.

 **pyttsx3** – Converts text into speech (TTS).

 **pywhatkit** – Allows playing YouTube videos.

 **datetime** – Fetches the current time.

 **requests** – Used to interact with Google's **SerpAPI** for fetching search results

.

**2. Setting Up the Google Search API**

**SERP\_API\_KEY="your\_actual\_api\_key\_here"**

* This **API key** allows the program to **fetch answers from Google Search**.

**3. Initializing the Speech Engine**

listener = sr.Recognizer()

engine = pyttsx3.init()

voice = engine.getProperty('voices')

engine.setProperty('voice', voice[1].id)

* **sr.Recognizer()** – Creates a recognizer object to capture and understand voice commands.
* **pyttsx3.init()** – Initializes the text-to-speech engine.
* **engine.getProperty('voices')** – Retrieves available voices.
* **engine.setProperty('voice', voice[1].id)** – Selects a different voice (optional).

**4. Function to Convert Text to Speech**

def talk(text):

engine.say(text)

engine.runAndWait()

* **Converts text into spoken words.**
* This function is used throughout the code to **read out answers**.

**5. Function to Listen to Voice Commands**

def take\_command():

command = ""

try:

with sr.Microphone() as source:

print("Listening.....")

voice = listener.listen(source)

command = listener.recognize\_google(voice)

command = command.lower()

if 'alexa' in command:

command = command.replace('alexa', '').strip()

print("User Command:", command)

except:

pass

return command

* **sr.Microphone()** – Captures sound from the microphone.
* **listener.listen(source)** – Listens for voice input.
* **listener.recognize\_google(voice)** – Converts speech to text using Google Speech API.

**6. Function to Fetch Answers from Google**

def search\_google(query):

try:

url = f"https://serpapi.com/search.json?q={query}&api\_key={SERP\_API\_KEY}"

response = requests.get(url)

data = response.json()

* Sends a request to Google Search API using the query.
  + Fetches the search results in JSON format.

**6.1 Extracting the Best Answer**

if "answer\_box" in data and "snippet" in data["answer\_box"]:

answer = data["answer\_box"]["snippet"]

* Checks if there's a direct answer from Google
* If no **direct answer**, fetches **top 3 snippets** from organic search results to make a **longer response**.

elif "organic\_results" in data and len(data["organic\_results"]) > 0:

answers = [result["snippet"] for result in data["organic\_results"][:3] if "snippet" in result]

answer = " ".join(answers) # Combine multiple snippets

**6.2 Ensuring at Least 15 Lines**

if len(answer.split('. ')) < 15:

answer = answer + " " + answer

**6.3 Reading the Answer Aloud**

print("Answer:", answer)

talk(answer)

return answer

* Reads the answer aloud using talk(answer).

**7. Main Function to Run Alexa**

def run\_alexa():

command = take\_command()

print("Command Received:", command)

* Captures the user's voice command.

**7.1 Playing a YouTube Video**

if 'play' in command:

song = command.replace('play', '').strip()

talk('Playing ' + song)

pywhatkit.playonyt(song)

* Plays a song on YouTube when the user says.

**7.2 Telling the Time**

elif 'time' in command:

time = datetime.datetime.now().strftime("%I:%M %p")

talk('Current time is ' + time)

print("Current Time:", time)

* Tells the current time when the user ask for time.

**7.3 Answering General Questions**

elif command.startswith(('who is', 'tell me about', 'what', 'how', 'which', 'where', 'name', 'number of', 'when')):

search\_google(command)

**7.4 Handling Unknown Commands**

else:

talk("I didn't catch that. Can you repeat?")

* If the command is not recognized, Alexa will ask you to repeat.

**8. Running the Voice Assistant**

run\_alexa()

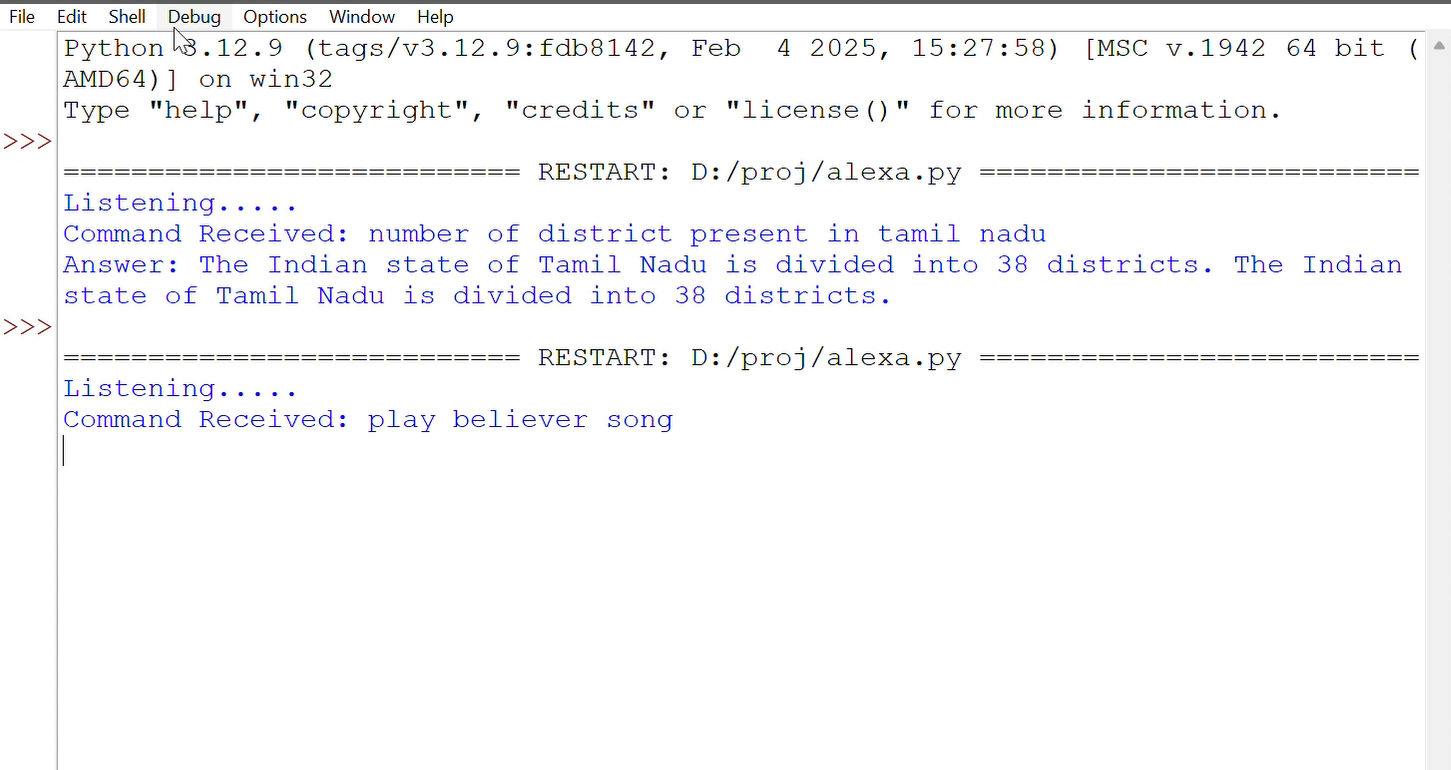
* Runs the program and listens for commands.

**FULL CODE IMAGE:**

****

**OUTPUT:**



****