**IRIS – VIRTUAL AI ASSISTANT**

**Overview:**

IRIS is a virtual AI assistant that can perform various tasks based on voice commands. The assistant can open applications, search the web, play music, and much more.

**Key Features:**

- Open and close applications like YouTube, Notepad, Calculator, and web browsers.

- Perform web searches and fetch information from Wikipedia.

- Play music and videos.

- Send WhatsApp messages.

- Provide current time, date, day, and month.

- Take screenshots and photos using the camera.

- Tell jokes and give random advice.

**Modules Used:**

- `sys`: To exit the program.

- `pyttsx3`: For text-to-speech conversion.

- `speech\_recognition`: For recognizing voice commands.

- `datetime`: To get the current date and time.

- `webbrowser`: To open web pages.

- `requests`: To make HTTP requests for fetching jokes and advice.

- `pywhatkit`: To play YouTube videos and send WhatsApp messages.

- `os`: To interact with the operating system.

- `subprocess`: To run subprocesses like opening applications.

- `pyautogui`: To take screenshots.

- `json`: To parse JSON data.

- `ecapture`: To capture images using the camera.

- `wikipedia`: To fetch summaries from Wikipedia.

**Functions:**

1. play\_youtube(query):

- Plays the specified YouTube video using `pywhatkit`.

2. get\_random\_advice():

- Fetches a random piece of advice from the "Advice Slip" API.

3. get\_random\_joke():

- Fetches a random joke from "icanhazdadjoke" API.

4. screenshot():

- Takes a screenshot and saves it to a specified directory.

5. open\_camera():

- Opens the camera application.

6. open\_notepad():

- Opens Notepad.

7. open\_cmd():

- Opens Command Prompt.

8. open\_calculator():

- Opens Calculator.

9. send\_whatsapp\_message(number, message):

- Sends a WhatsApp message using `pywhatkit`.

10. get\_news():

- Fetches the latest news headlines from "The Times of India" using the News API.

11. speak(audio):

- Converts text to speech using `pyttsx3`.

12. listen():

- Listens to the user's voice and returns the recognized text using `speech\_recognition`.

13. wishMe():

- Greets the user based on the time of day and introduces itself.

14. takeCommand():

- Listens to the user's voice command and returns the recognized text.

**Main Program Execution:**

1. wishMe() is called to greet the user.

2. while True loop starts to continuously listen for voice commands.

3. takeCommand() is called to get the user's command.

4. Based on the command, different functions are executed, such as opening applications, searching the web, fetching jokes or advice, and so on.

5. if "exit" in query: breaks the loop and exits the program.

Example Commands:

- "Open YouTube"

- "Play on YouTube"

- "What is the time?"

- "Tell me a joke"

- "Take a screenshot"

- "Send a WhatsApp message"

**Conclusion:**

This project demonstrates how to build a virtual assistant using Python. By integrating various libraries and APIs, IRIS can perform a wide range of tasks, making it a versatile and useful tool. The code is structured to be easily understandable and extendable for adding more features in the future.