

# PYTHON PROJECT REPORT



Project Title : Web Automation

Name : Jaison Paul Ebinezar

Skill : Python

ID : CO2309052681168

Qualification : DEEE(2021-2023)

# JAISON PAUL EBINEZAR



## PROJECT:

- Web Automation Using in Python
- python learning webpage(DJANGO)

## EDUCATION:

### DIPLOMA IN EEE

Priyadharshini polytechnic college ,  
Tirupattur.

(2021 -2023)

### SSLC

Grace Baptist school,Tirupattur.

(2019 - 2020)

## SOFT SKILLS:

- Python
- Django
- Html,Css,Javascript,Bootstrap
- Mysql,dbbrowser

To secure responsible career opportunity to fully utilize my knowledge and skills while making a significant contribution to success of the success of the company and to gain my knowledge and experience


## PROFILE

|               |                  |
|---------------|------------------|
| Student Id    | :CO230905Z681168 |
| Age           | : 20             |
| Father Name   | :Dhanasekaran    |
| Mother Name   | : Malar          |
| Nationality   | : Indian         |
| Religion      | : Christian      |
| Marital satus | : Single         |

## LANGUAGE KNOWN

Tamil(Read,Write,Speak)  
English(Read,Write,Speak)

## CONTACT

 9843798016

 jaisonpaulebinezar84@gmail.com

 3/148,Thillai Nagar,Vaniyambadi Road,  
Thirupathur District -635601.

# Web Automation

Web automation in python is the process of using python scripts to control web browsers and perform tasks on the web<sup>12</sup>. To do web automation in python, you need to<sup>12</sup>:

- Install pyttsx3, `pyttsx3` is a text-to-speech conversion library in Python. Unlike alternative libraries, it works offline and is compatible with both Python 2 and 3. An application invokes the `pyttsx3.init()` factory function to get a reference to a `pyttsx3. Engine` instance. it is a very easy to use tool which converts the entered text into speech. The `pyttsx3` module supports two voices first is female and the second is male which is provided by “sapi5” for windows. It supports three TTS engines : Install a web driver, such as `ChromeDriver` or `FirefoxDriver`, to communicate with the browser.
- Install `SpeechRecognition` library acts as a wrapper for several popular speech APIs and is thus extremely flexible. One of these—the Google Web Speech API—supports a default API key that is hard-coded into the `SpeechRecognition` library. That means you can get off your feet without having to sign up for a service.
- In Python, date and time are not data types of their own, but a module named `DateTime` can be imported to work with the date as well as time. Python `Datetime` module comes built into Python, so there is no need to install it externally.
- In Python, date and time are not data types of their own, but a module named `DateTime` can be imported to work with the date as well as time. Python `Datetime` module comes built into Python, so there is no need to install it externally.

- In this tutorial you'll learn advanced Python web automation techniques: using with a "headless" browser, exporting the scraped data to CSV files, and wrapping your scraping code in a Python class.
- Python has a number of libraries that can be used for speech recognition. One such library is SpeechRecognition1. It is a full-featured and easy-to-use Python speech recognition library that supports several engines and APIs, both online and offline. The library supports the following speech recognition engines/APIs: CMU Sphinx, Google Speech Recognition, Google Cloud Speech API, Wit.ai, Microsoft Azure Speech, Microsoft Bing Voice Recognition (Deprecated), Houndify API, IBMSpeech to Text, Snowboy Hotword Detection, Tensorflow, Vosk API, OpenAI whisper, and Whisper API1.
- To install the library, you can use pip by running the following command in your terminal:

### Python

- pip install SpeechRecognition
- AI-generated code. Review and use carefully. More info on FAQ.
- After installing, you can quickly try it out by running the following command `Python -m speech_recognition` AI-generated code. Review and use carefully..
- For more details on how to use the library, you can refer to the library reference or the examples provided in the repository1.

### Source Code Link:

<https://www.mediafire.com/file/d5dy05sku0iqalo/Project+source+code.pdf/file>

# WEB AUTOMATION

## USES OF PYTHON

- Desktop Applications
- Web Application
- Game Development
- Machine Learning
- Artificial Intelligence

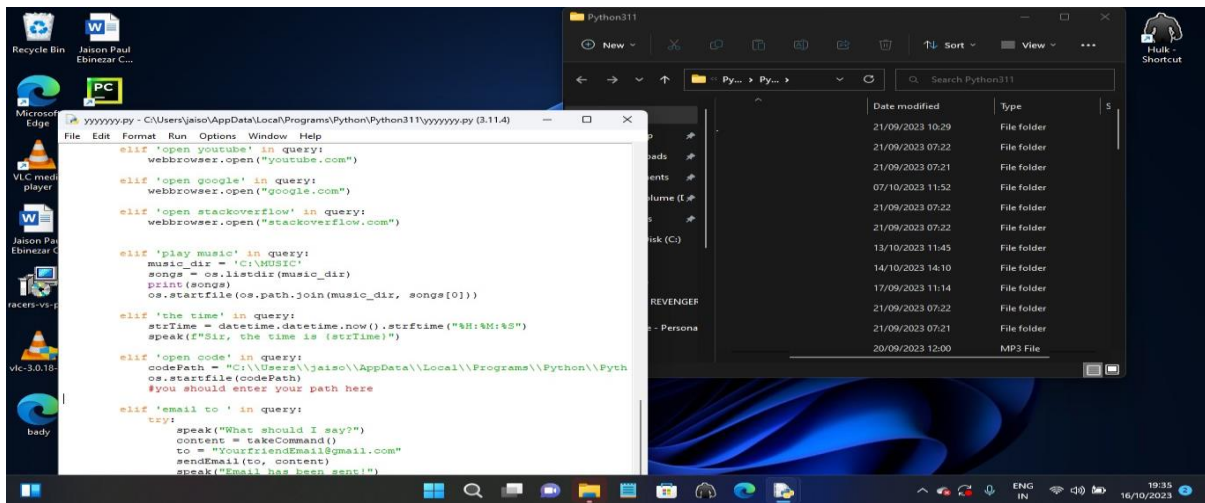
## LIST OF MODULES

- Searching Wikipedia
- YouTube
- Face book
- Stack Overflow
- Play Music
- Time
- Serching Google

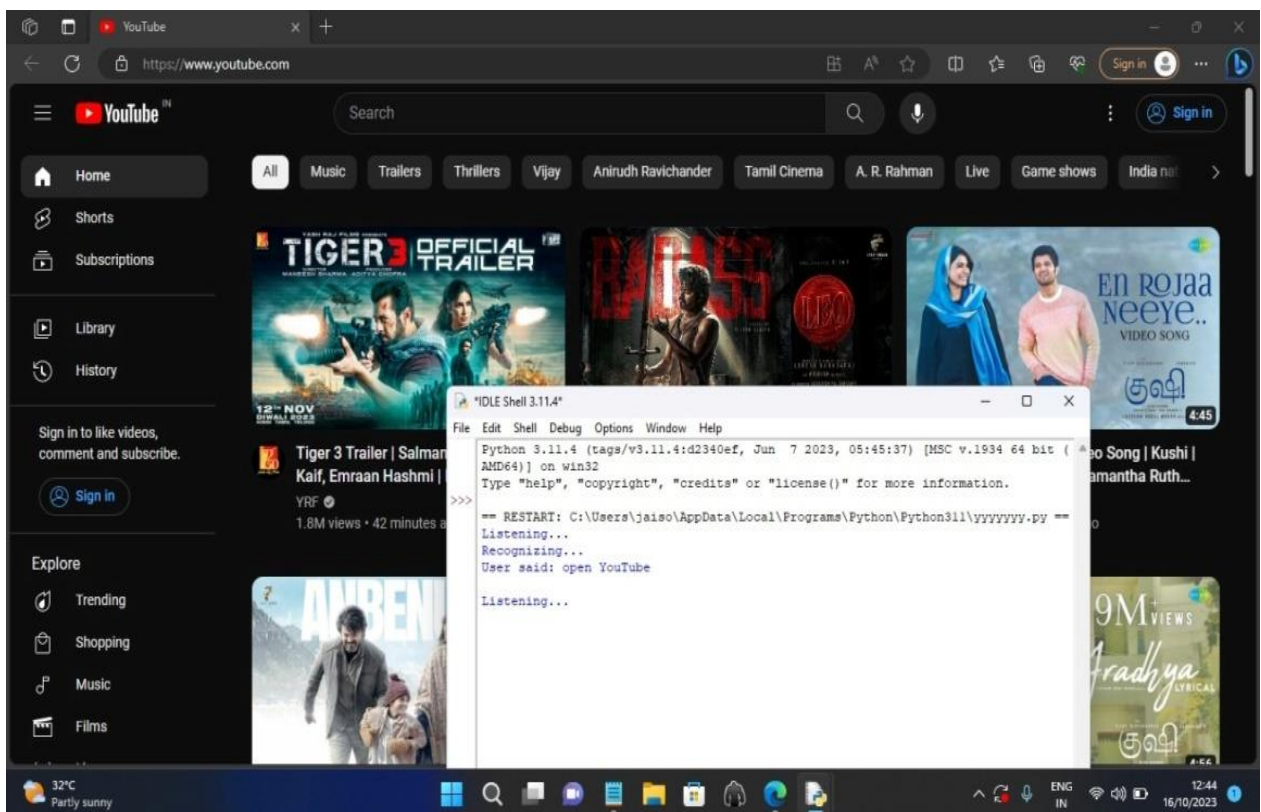
## LIST OF IMPORT

- Import Pyttsx3
- Import Speech Recognition
- Import Date And Time
- Import Wikipedia
- Import Web Browser
- Import Os
- Import Smtplib

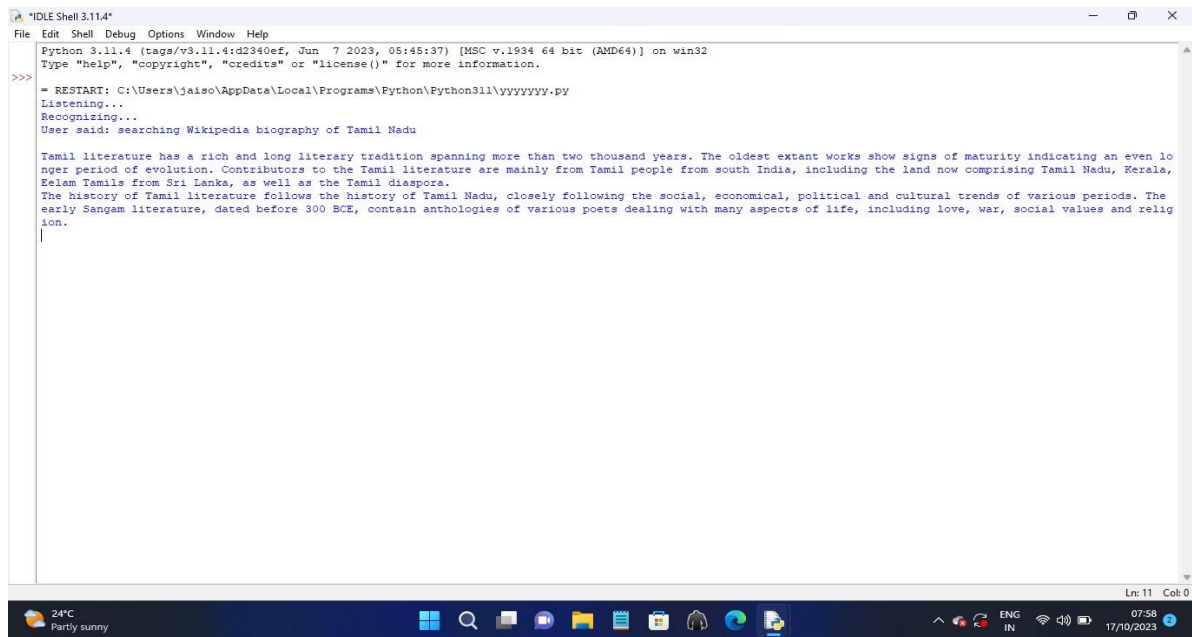
# OUTPUTPAGE



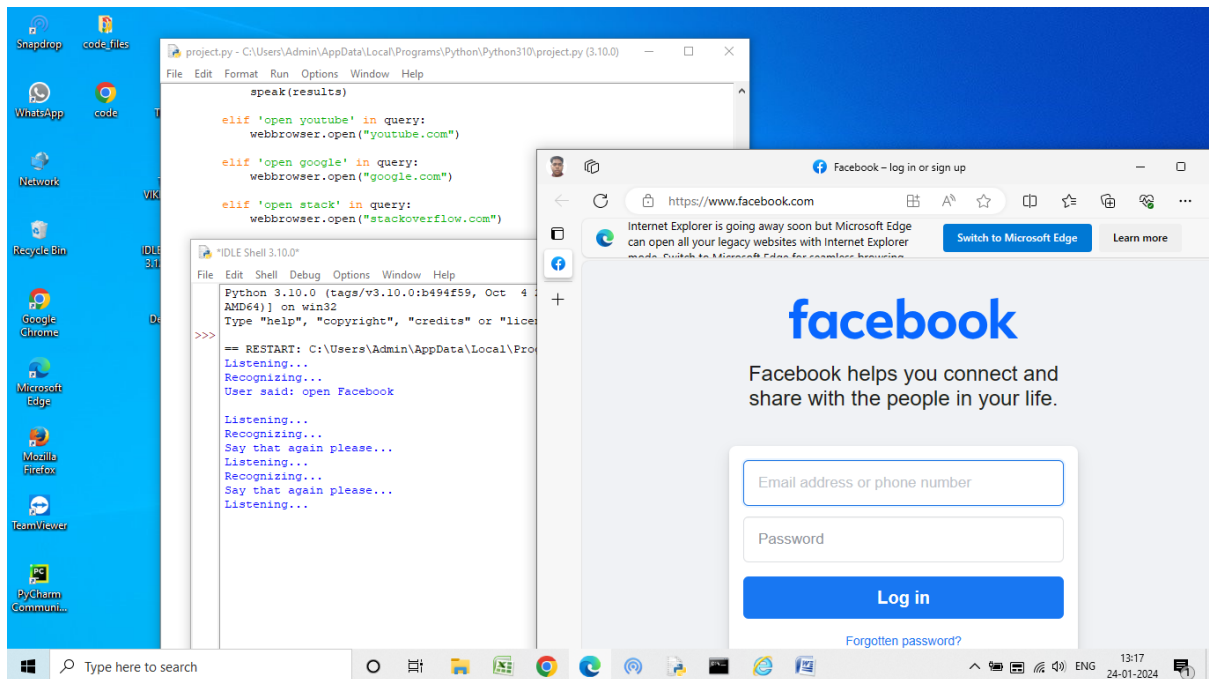
OPEN YOUTUBE



# Searching Wikipedia



# Open Facebook





The image shows a Windows 10 desktop environment with a blue background. Several application icons are visible on the left side of the desktop, including Snapdrop, code.files, WhatsApp, code, Network, Recycle Bin, Google Chrome, Microsoft Edge, Mozilla Firefox, TeamViewer, PyCharm Community, and a folder named 'code'. Three application windows are open:

- Top Window:** A code editor window titled 'project.py - C:\Users\Admin\AppData\Local\Programs\Python\Python310\project.py (3.10.0)'. It contains a Python script for a simple web server using the 'http.server' module. The script listens on port 8080 and responds to requests with a message that includes the user's email address and the content of the request.
- Middle Window:** A web browser window displaying the Stack Overflow website. The address bar shows 'https://stackoverflow.com'. There is a notification banner about Internet Explorer being phased out.
- Bottom Window:** An 'IDLE Shell 3.10.0' terminal window showing the execution of the Python script. The output indicates that the server is running on port 8080 and is listening for connections. It shows several 'Recognizing...' messages and a 'User said: open stack' message, which corresponds to the 'while True' loop in the script.

The taskbar at the bottom of the screen shows the Start button, a search bar, and several pinned application icons. The system tray on the right shows the date and time as 12:45 on 24-01-2024.

## The screenshot shows a Windows desktop environment. In the background, a terminal window titled "project.py - C:\Users\Admin\AppData\Local\Programs\Python\Python310\project.py (3.10.0)" displays a Python script named "wishMe()". The script uses a while loop to interact with a user, taking input and providing feedback based on keywords like "help", "copyright", "credits", or "restart". It also interacts with a web browser to fetch music information from "keys-of-moon-white-petals(chosic.com)". Overlaid on top of the terminal is a music application window titled "My music". This window has tabs for "Songs", "Artists", and "Albums", with "Albums" currently selected. It features controls for shuffling all songs, sorting by date added, and filtering by genre. Below these controls are four album covers displayed in a row: "White Petals" by Keys of Moon, "An Everyday World" by Mark-J, "Virtues" by Amber Pacific, and "School of Swagg" by Young Chozen. At the bottom of the screen is the Windows taskbar, which includes the Start button, a search bar, and several pinned application icons. The system tray on the right shows the time as 12:55 on 24-01-2024 and various status icons.



## Conclusion

Tasks both general and in my field of data science. Pyttsx3 is a text-to-speech conversation library in python. The program will never lose focus and wander. The technique of automating the web with Python works great for many off to wikipedia. It will faithfully carry out the same exact series of steps with perfect consistency .

Student Sign

(Jaison Paul Ebinezar.D)

Staff Sign

(Radhika.R)