Information Gathering Tool Project

Cyber Security_Ethical Hacking Domain Project 1
Your Name: Harsh Sanjay Padishalwar
Date: 1-11-24

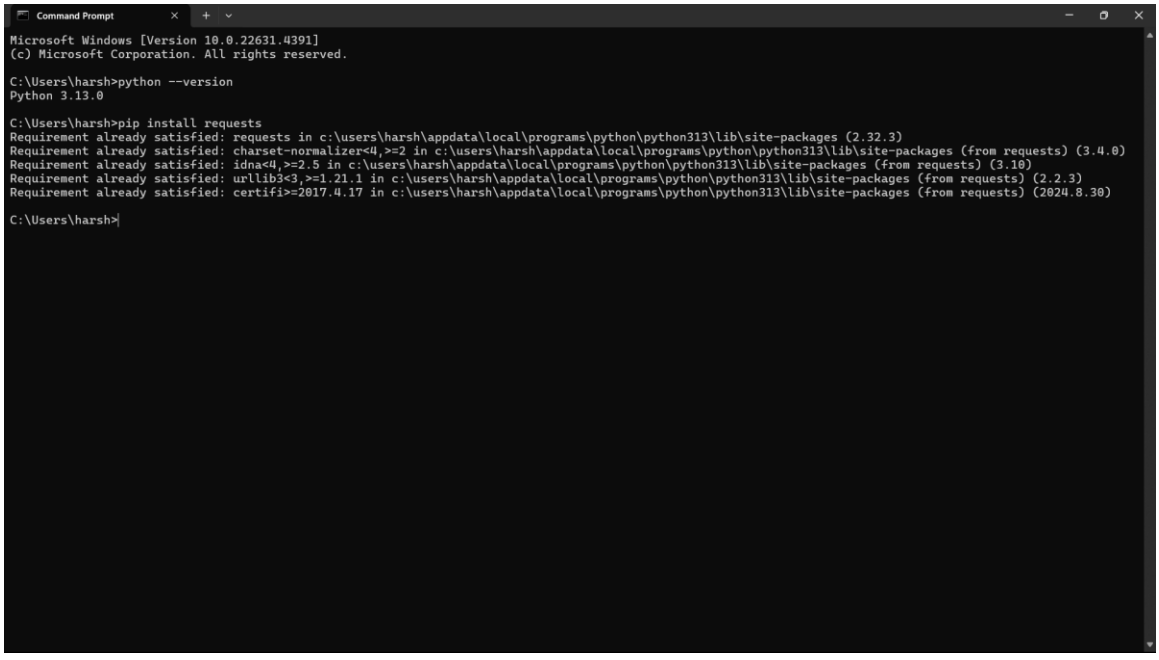
1. Introduction

This project aims to develop a Python-based tool that retrieves and displays IP address and location data for a specified website URL. Utilizing essential Python libraries and an external IP information API, this tool processes a URL to obtain its IP address and fetches location details including city, country, and organization.

The following libraries are used in this project:
- sys: for command-line argument handling
- requests: for making HTTP requests to the ipinfo.io API
- json: for parsing data in JSON format
- socket: for retrieving the IP address of the provided website URL

2. Environment Setup

To begin, Python and the necessary libraries must be installed:

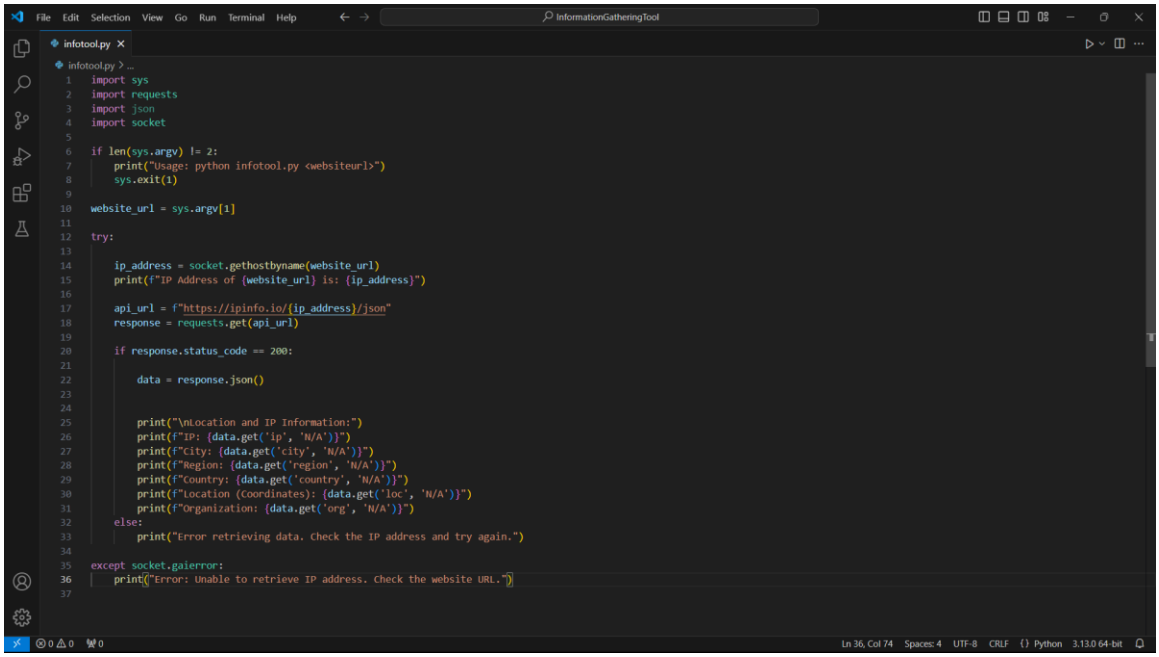


3. Writing the Code

This section describes the code structure used in this project. The script, named `infotool.py`, is designed to:

1. Accept a website URL from the command line.
2. Retrieve the IP address of the website using the `socket` library.
3. Make an API request to ipinfo.io using the `requests` library to gather additional information about the IP address, such as location and organization.
4. Parse the JSON response from the API to extract and display relevant information.

The script includes error handling to manage incorrect URLs or network issues. Ensure each section of code is tested as it is written.

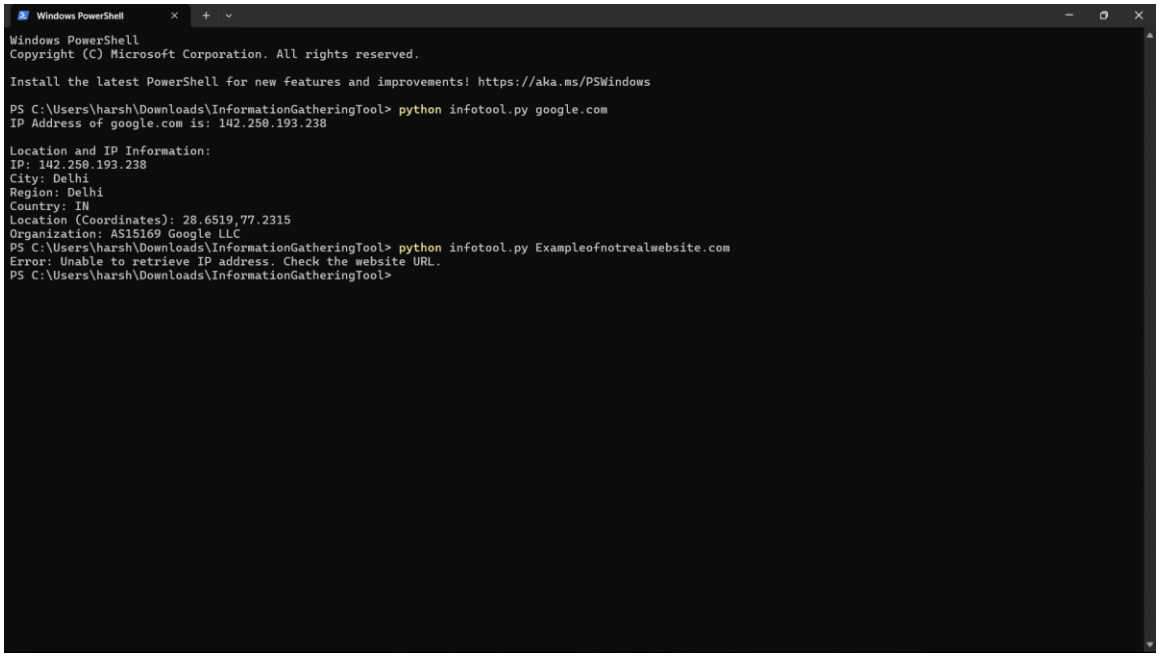


4. Running the Script
To execute the script, open a command prompt or terminal in the directory where `infotool.py` is saved. Run the script by entering the command:

`python infotool.py <websiteurl>`

For example:

`python infotool.py google.com`
This command fetches the IP address and additional information for the provided website URL.



Upon execution, the tool displays the following information:
- IP Address
- City
- Region
- Country
- Coordinates (latitude and longitude)
- Organization

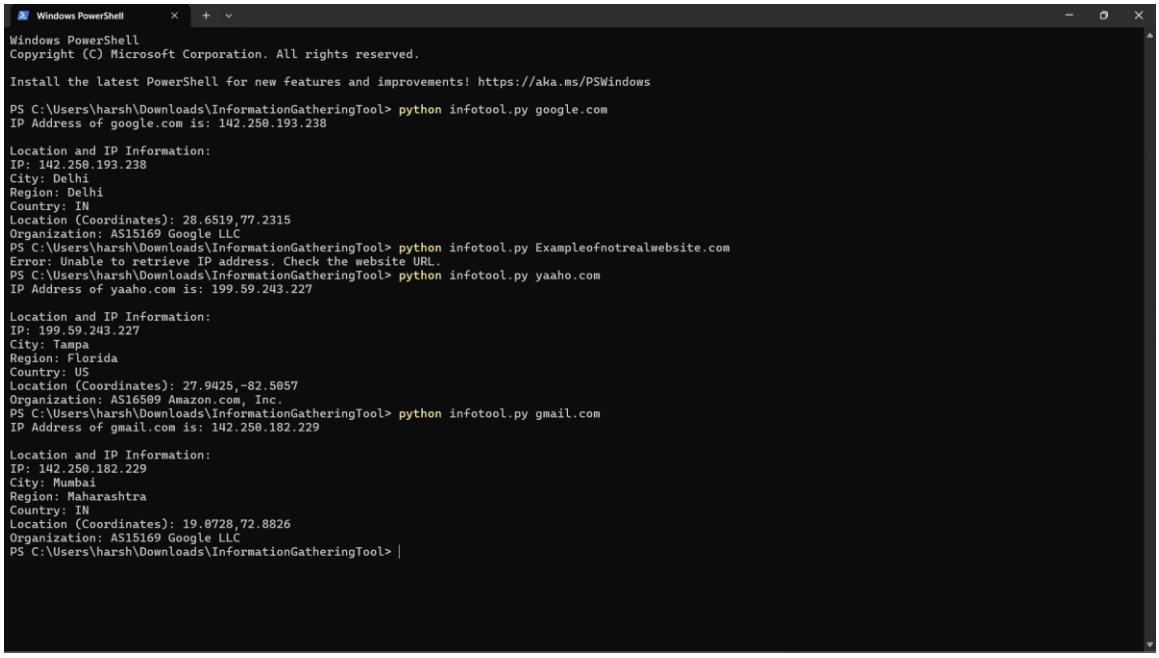
The information is presented in a clear and organized format for easy reading.

The tool also includes error handling to manage cases where the provided URL is invalid. If an incorrect or non-existent URL is entered, the tool will display an error message indicating the issue. This feature helps in guiding the user to provide valid input.

5. Additional Output Examples

To demonstrate the tool's reliability, additional tests can be run with different URLs. This shows consistent functionality and accuracy in retrieving IP and location information.

Examples:
- Running the tool for yahoo.com.
- Running the tool for example.com.



6. Conclusion

This project successfully demonstrates the creation of an information-gathering tool that collects IP address and location data for a given website URL. By combining Python's socket capabilities with an external API, the tool is able to deliver accurate and organized information about any accessible URL. The project was an opportunity to work with APIs, handle JSON data, and manage network error handling, all of which are essential skills in cybersecurity and ethical hacking domains.

Challenges encountered included managing potential network errors and ensuring proper data formatting. Future improvements could include expanding the tool's functionality to provide even more detailed information, such as historical IP data or ISP information.