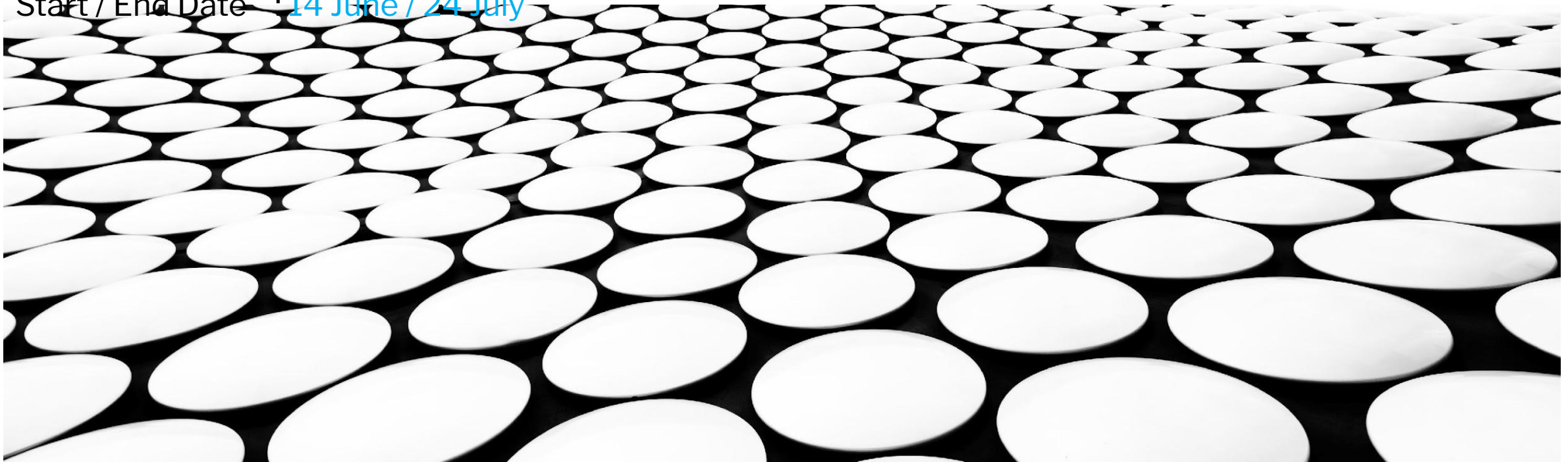


STUDENT DETAILS

Name : Ayush Das Pattanaik
SkillsBuild Email ID :
2041016183.ayushdaspattnaik@gmail.com
College Name : Siksha 'O' Anusandhan
College State : Odisha
Internship Domain : Cyber Security
Start / End Date : 14 June / 24 July



(my candid picture)



KEYLOGGER : CAPTURING KEY STROKES

- The objective of the project "Keylogger : Capturing Key Strokes" is to design and implement a keylogging application that records and logs keystrokes made by a user on a computer. The keylogger will capture key presses from the keyboard, including regular character keys, special keys (e.g., Shift, Ctrl), and function keys, and store them in a text file for further analysis. The project aims to explore the concept of keyloggers for educational purposes and responsible use in the domain of cybersecurity.

AGENDA

- Project setup and Environment : The keylogger is coded with Python, on a windows machine, using Python IDLE, VS Code .
- Building and Implementation : Implementing basic keylogging functionality, handling key events and capturing keystrokes, configuring logging options(e.g., log file format, encryption, etc.) .
- Legal and Ethical Aspects : Reviewing the legal implications of keyloggers, Understanding ethical guidelines for responsible use and users, Discussing potential lawful applications of keyloggers with end users.
- Future Enhancements and Considerations : Discussing potential use cases for the keylogger, customizations.

OVERVIEW

- The purpose of the "Keylogger Capturing Key Strokes" project is to develop a keylogging application that can record and log keystrokes made by a user on a computer. The project aims to explore the concept of keyloggers for educational purposes and responsible use in the field of cybersecurity.
- The project's scope encompasses the design and implementation of a Python-based keylogging application. The keylogger will capture all keyboard inputs made by the user while the application is running. This includes regular character keys, special keys (e.g., Shift, Ctrl), and function keys. The captured keystrokes will be stored in a text file, organized with timestamps for further analysis.
- The project's objective is to implement a keylogger and save the keystrokes entered by the user in real-time while the application is active, in a text file for further analysis.



WHO ARE THE END USERS OF THIS PROJECT?

- Cybersecurity Enthusiasts and Researchers : Students, cybersecurity enthusiasts, researchers, and professionals in the cybersecurity field. They use the project as an educational tool to understand how keyloggers work, their legitimate applications, and the potential risks associated with them. These users are motivated by curiosity and a desire to enhance their knowledge of cybersecurity concepts.
- Ethical Hackers and Security Professionals : Ethical hackers, security professionals, and penetration testers can also be potential end users of the keylogger project. They may use the keylogging application in a controlled environment to test and assess the security of computer systems and networks. For these professionals, understanding the workings of keyloggers is essential for identifying and mitigating potential vulnerabilities in systems. However, they use the keylogger solely for ethical purposes and within the bounds of their ethical responsibilities.

YOUR SOLUTION AND ITS VALUE PROPOSITION

- Ethical Use and User Consent : Unlike malicious keyloggers used for unauthorized monitoring, my keylogging application emphasizes ethical use.
- Educational Value : My solution caters to cybersecurity enthusiasts, researchers, students, and professionals looking to expand their knowledge in the field. By understanding the workings of keyloggers, users can gain insights into potential vulnerabilities and learn how to protect against malicious keylogging attempts.
- Platform Compatibility : My solution is designed to be compatible with popular operating systems like Windows, macOS, and Linux. This ensures broader accessibility, allowing users from different platforms to engage in cybersecurity education through the keylogger project.
- In conclusion, this project offers a valuable and educational tool for understanding keyloggers and their applications in cybersecurity. Through its ethical use, user consent, and responsible development, the solution aims to raise awareness, enhance knowledge, and foster a culture of responsible cybersecurity practices among its users.

HOW DID YOU CUSTOMIZE THE PROJECT AND MAKE IT YOUR OWN

- Replaced “” with spaces for easy review and analysis of the saved log file.
- Added functionality of capturing the user's clipboard, user clipboard will also get logged into a txt extension file.
- Added functionality to get information of the user's device, like its processor, public ip, private ip, windows version, and save it in a txt file.
- Added iterative statements with added functionality which helps to capture screenshots, and audio in png and wav format respectively, in regular intervals for a specified amount of rounds while the keylogger is running.
- Additionally, the python source code can be modified as an exe file to be used in a windows machine, or modified to be used in a macOS.

MODELLING

- pynput module : Key and Listener functions
- win32clipboard module : logging the clipboard contents
- platform and os module : logging device information
- PIL module : logging screenshots in .png format
- sounddevice module : logging audio in .wav format
- Encryption can be implemented for added security. Optionally, remote access functionality can be modeled for sending captured data to a specific location.
- Thorough testing and validation are conducted to ensure proper functioning. Throughout the modeling process, ethical and legal considerations are paramount, ensuring the responsible use of the keylogger.

RESULTS

- The keylogger was implemented on a windows device, it was run from the python IDLE.
- While running it was able to capture every keystrokes made on the device, and saved on key_log.txt file.
- Screenshots were taken and stored in the provided destination with a png extension.
- Audio of 10 seconds was also recorded and stored in wav extension.
- System device was also stored in a txt file.
- The keylogger remained undetected from the firewall as it was not sending any data outside the private network.
- When additional feature of sending the stored data to outside of private network was added, it was detected and flagged by the firewall/antivirus.
- The keylogger when tried to connect and send data outside of private network, it was flagged, hence it shows the firewall was able to detect it.

LINKS

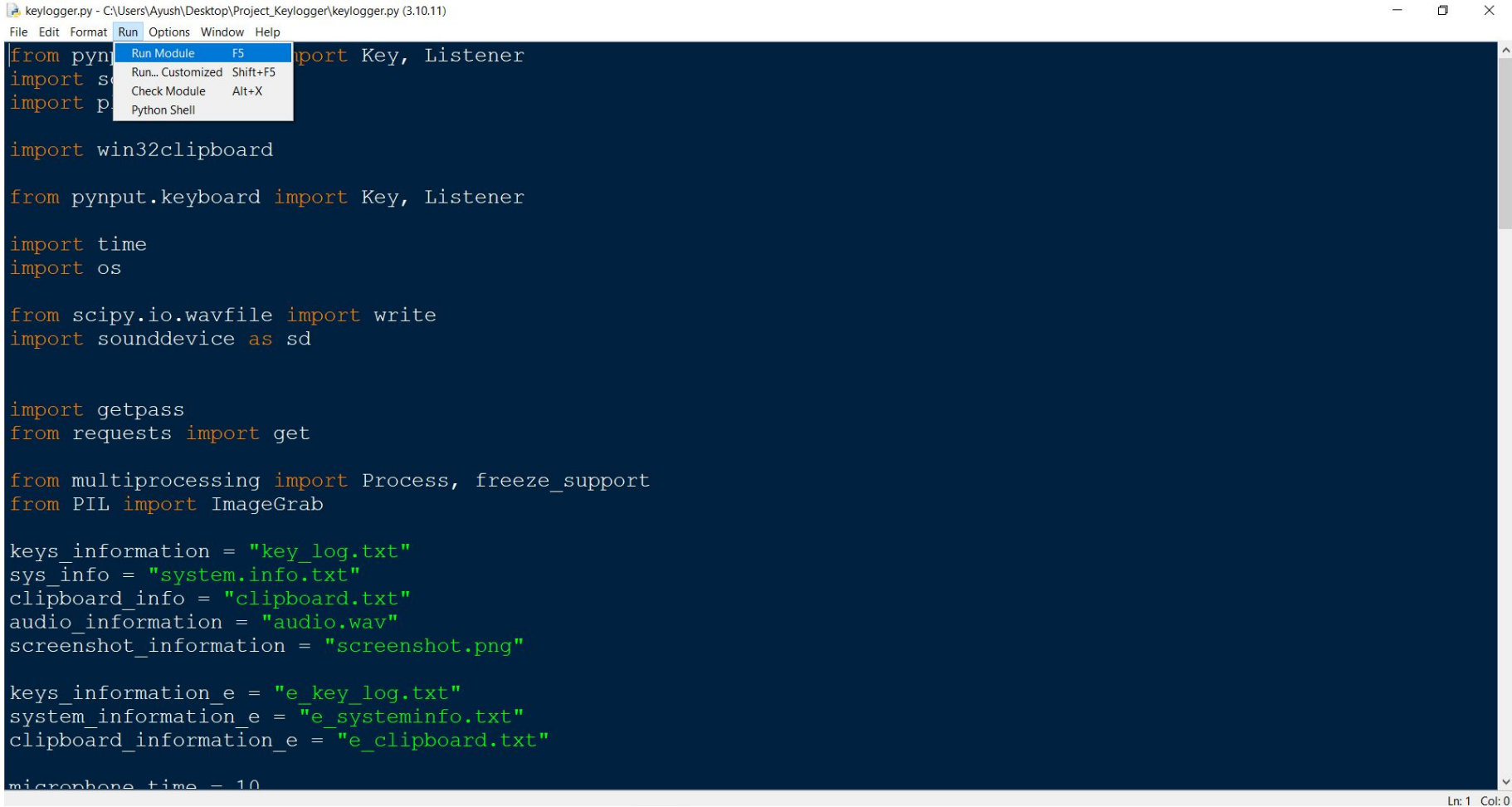
[My Keylogger on GitHub](#)

[Python MIME reference](#)

[Keylogger reference](#)

[My LinkedIn](#)

SOME VISUALS



The image shows a screenshot of a Python IDE window titled 'keylogger.py - C:\Users\Ayush\Desktop\Project_Keylogger\keylogger.py (3.10.11)'. The window has a menu bar with 'File', 'Edit', 'Format', 'Run', 'Options', 'Window', and 'Help'. The 'Run' menu is open, showing options: 'Run Module' (F5), 'Run... Customized' (Shift+F5), 'Check Module' (Alt+X), and 'Python Shell'. The code in the editor is a Python script for a keylogger. It includes imports for pynput, win32clipboard, time, os, scipy.io.wavfile, requests, multiprocessing, and PIL. It defines file paths for key logs, system info, clipboard, audio, and screenshots. It also sets up event listeners for key presses and system events. The script is currently at line 1, column 0.

```
keylogger.py - C:\Users\Ayush\Desktop\Project_Keylogger\keylogger.py (3.10.11)
File Edit Format Run Options Window Help
from pynput import Key, Listener
import sys
import pynput

import win32clipboard

from pynput.keyboard import Key, Listener

import time
import os

from scipy.io.wavfile import write
import sounddevice as sd

import getpass
from requests import get

from multiprocessing import Process, freeze_support
from PIL import ImageGrab

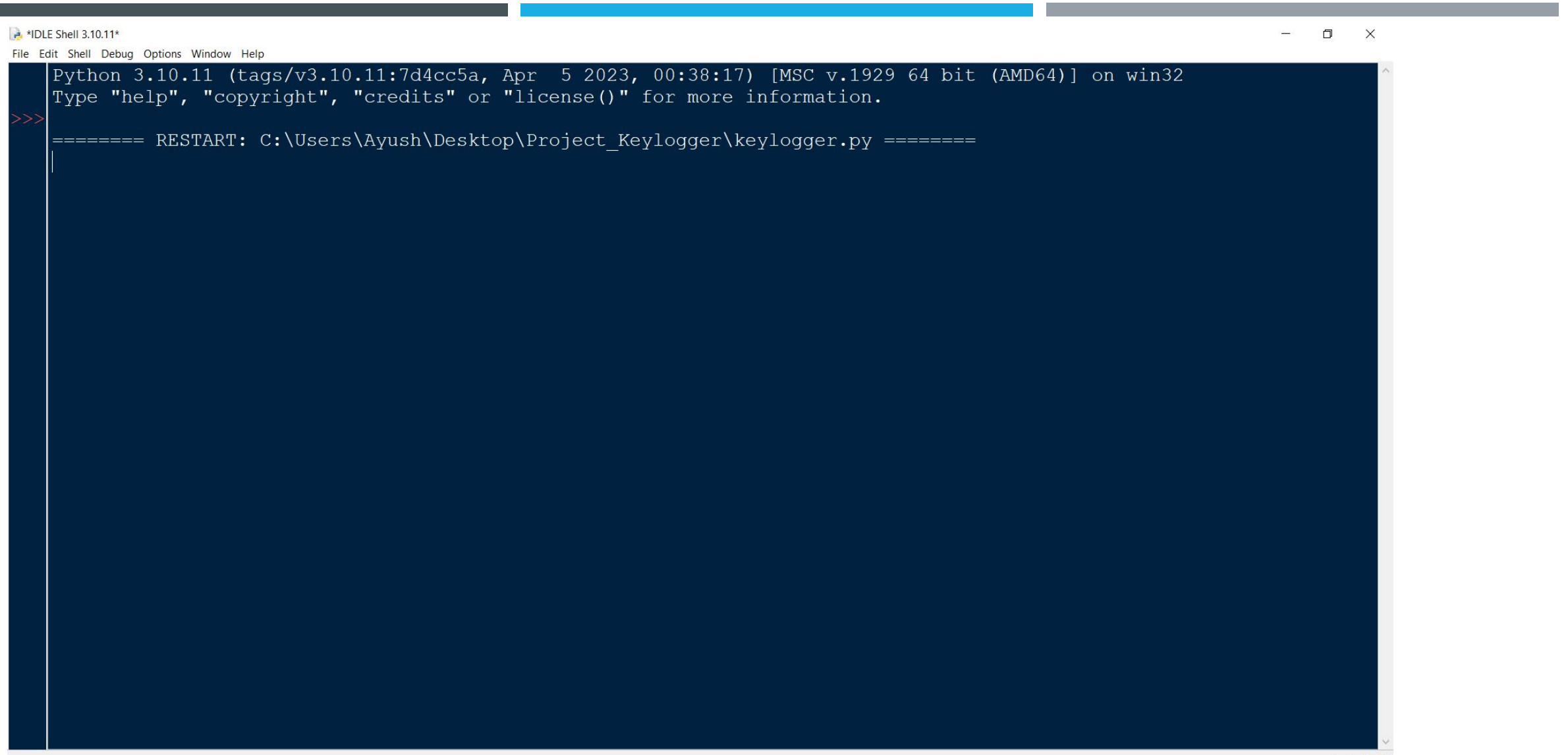
keys_information = "key_log.txt"
sys_info = "system.info.txt"
clipboard_info = "clipboard.txt"
audio_information = "audio.wav"
screenshot_information = "screenshot.png"

keys_information_e = "e_key_log.txt"
system_information_e = "e_systeminfo.txt"
clipboard_information_e = "e_clipboard.txt"

microphone_time = 10
```

Ln: 1 Col: 0

before running

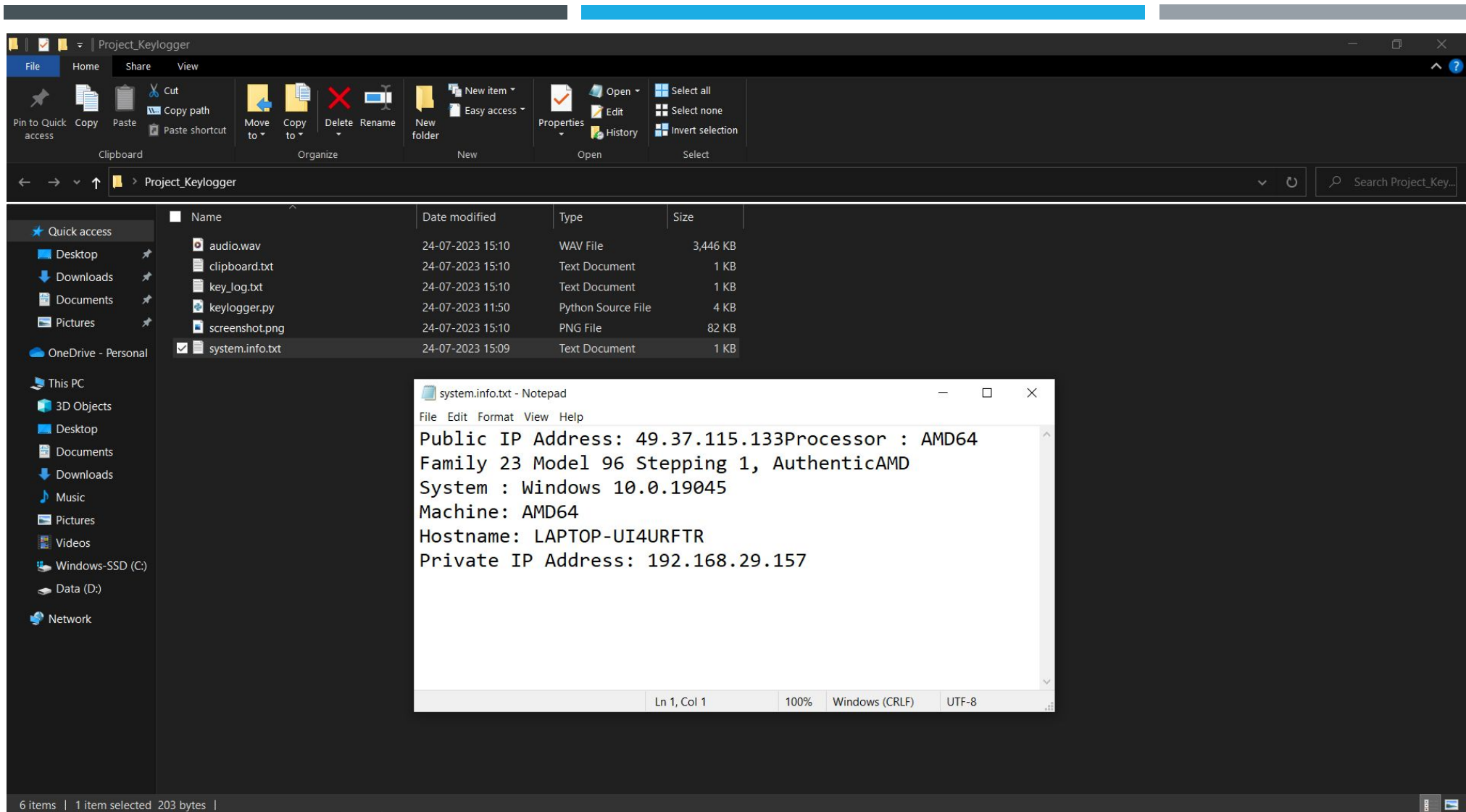


The screenshot shows a Python IDLE Shell window with a dark blue background and white text. The window title is '*IDLE Shell 3.10.11*'. The menu bar includes 'File', 'Edit', 'Shell', 'Debug', 'Options', 'Window', and 'Help'. The shell displays the following text:

```
Python 3.10.11 (tags/v3.10.11:7d4cc5a, Apr 5 2023, 00:38:17) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\Ayush\Desktop\Project_Keylogger\keylogger.py =====
|
```

The prompt '>>>' is shown in red. A vertical cursor is positioned on the line following the restart message. The window has standard Windows window controls (minimize, maximize, close) in the top right corner.

successful run



stored logs : key strokes, audio, screenshot, system information, clipboard content