**Project Report: KEYLOGGER FOR SESSION-BASED KEYSTROKE LOGGING**

**1. Project Overview:**

Developed a keystroke logging tool using Python and the pynput library, which records user keystrokes into a log file with session tracking. This tool captures keyboard events in real-time and organizes them by session, logging each keypress until the ESC key is pressed to end logging.

**2. Objectives:**

* To track user keystrokes in real-time.
* To store keystroke data in a text file (key\_log.txt).
* To allow for multiple sessions, each marked with a timestamp when a session starts.
* To provide a way for the user to stop the logging by pressing the **ESC** key.

**3. Tools and Technologies Used:**

* **Python**: The programming language used for the project.
* **pynput Library**: A Python library used to monitor and control input devices such as keyboards and mice.
* **Text File (key\_log.txt)**: The output file where the keystrokes are saved.

**4. Working of the Project:**

The keystroke logger functions by listening to keyboard events and logging each key pressed during the session. Here's how it works step-by-step:

1. **Session Initialization**:
   * When the script starts, it opens the key\_log.txt file in append mode. A new session marker ("New session started") is written to the file, indicating the beginning of a new logging session.
2. **Key Press Detection**:
   * The program listens for the **on\_release** event, which triggers every time a key is released.
   * For every key released, the key is converted to a string and stripped of unnecessary characters such as quotes to ensure cleaner logging.
3. **Logging Keystrokes**:
   * Each captured keystroke is written to key\_log.txt in real-time as the user types. The logging continues until the **ESC** key is pressed.
4. **Session Termination**:
   * When the **ESC** key is pressed, the listener stops, effectively ending the logging session. The program will exit gracefully at this point.
5. **5. Code Excution:**

* **Session Start**:
  + The session starts by writing the "New session started" entry into the log file (key\_log.txt) in append mode. This keeps a record of when each session begins.
* **Key Release Event**:
  + The on\_release function captures key releases. It converts the key to a string and removes any single quotes, ensuring that only the key itself is recorded.
* **ESC Key Handling**:
  + If the **ESC** key is detected, the listener is terminated by returning False, which ends the session and stops further logging.
* **Real-Time Logging**:
  + For every key pressed, the corresponding key string is appended to the key\_log.txt file. This allows for logging the sequence of keys typed by the user.
* **Listener Setup**:
  + The keyboard.Listener listens for key release events, triggering the on\_release function whenever a key is released. The listener is active until the program is terminated with the **ESC** key.

**6. Output:**

* The output of the project is stored in the file key\_log.txt. Each session is separated by a new entry, such as "New session started". The logged keys are appended to this file, capturing every keystroke during the session.