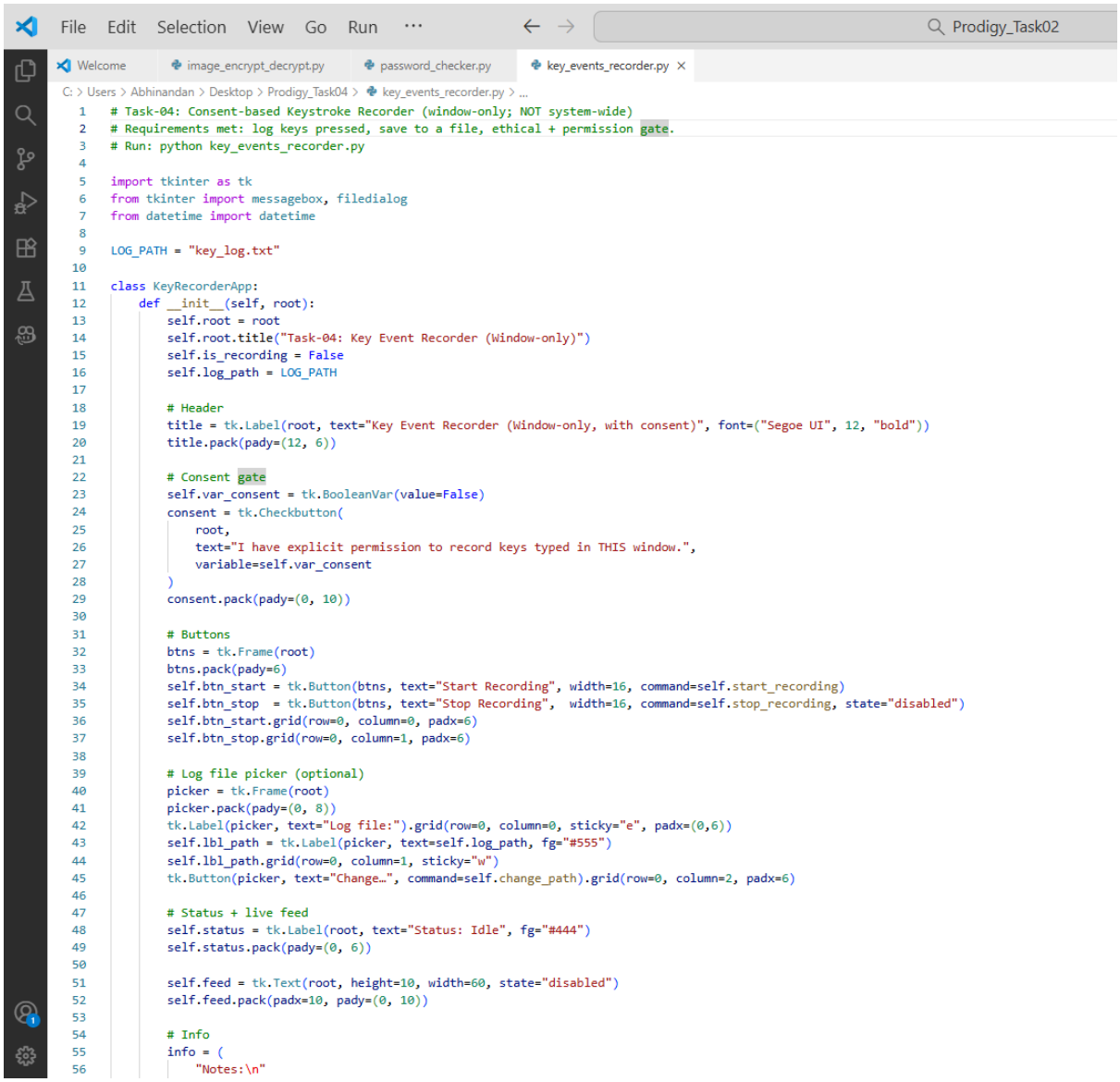
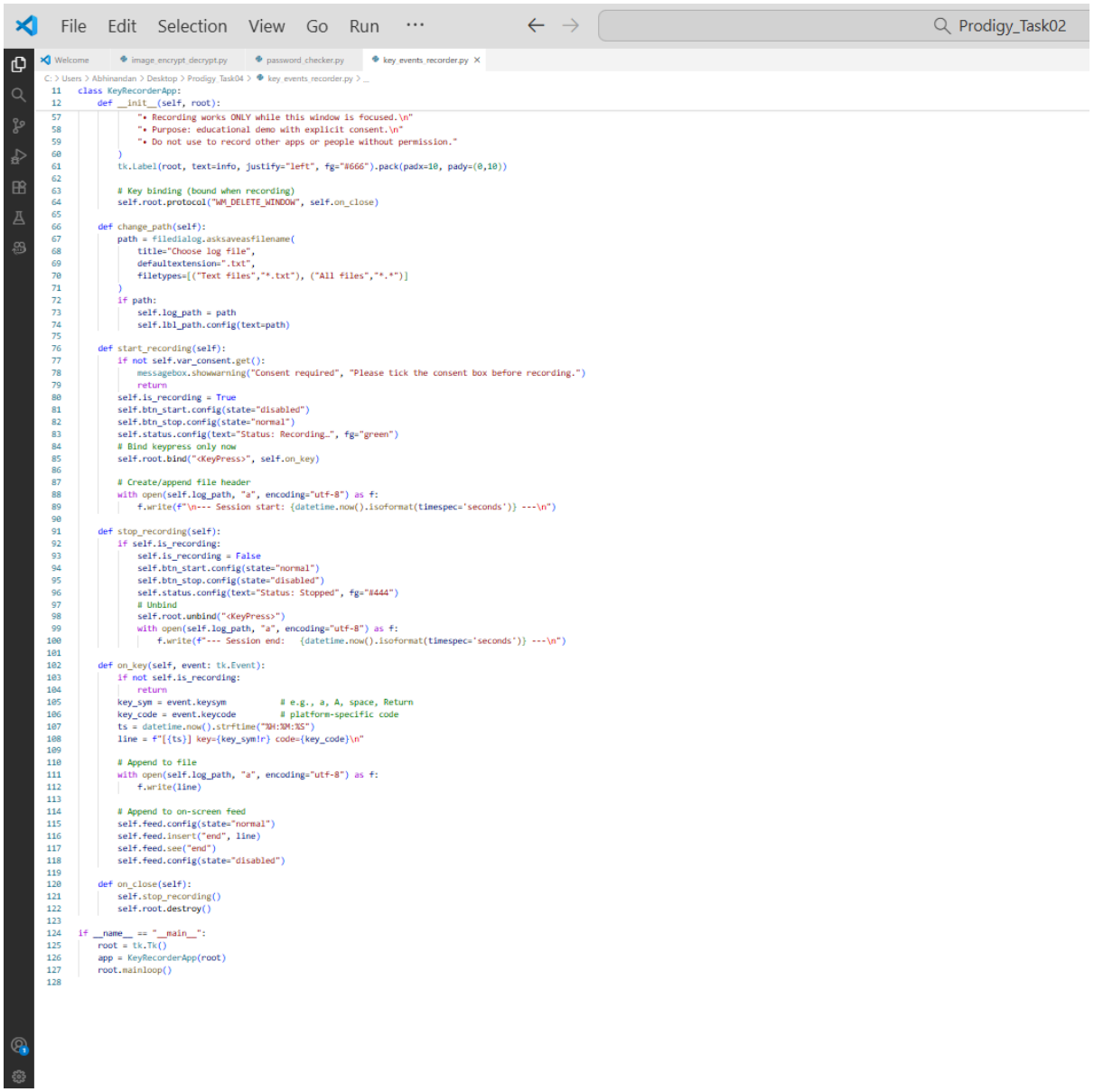
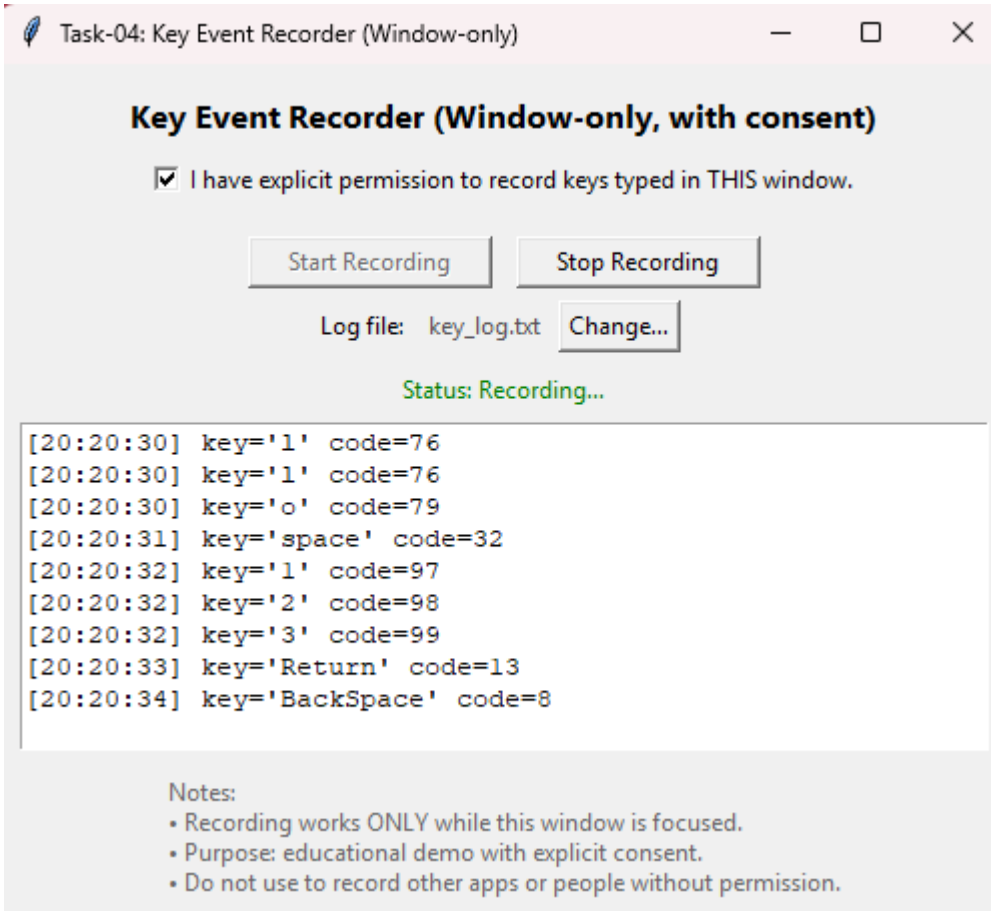
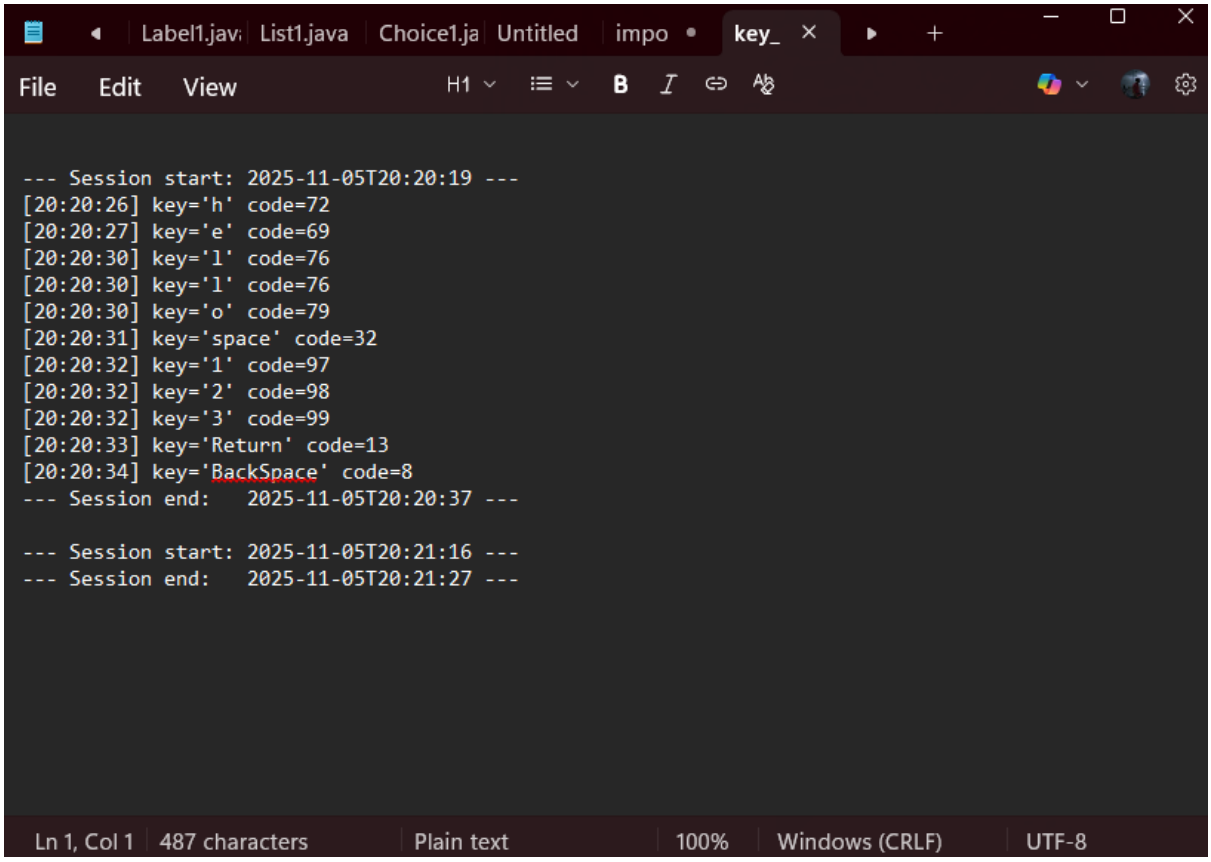
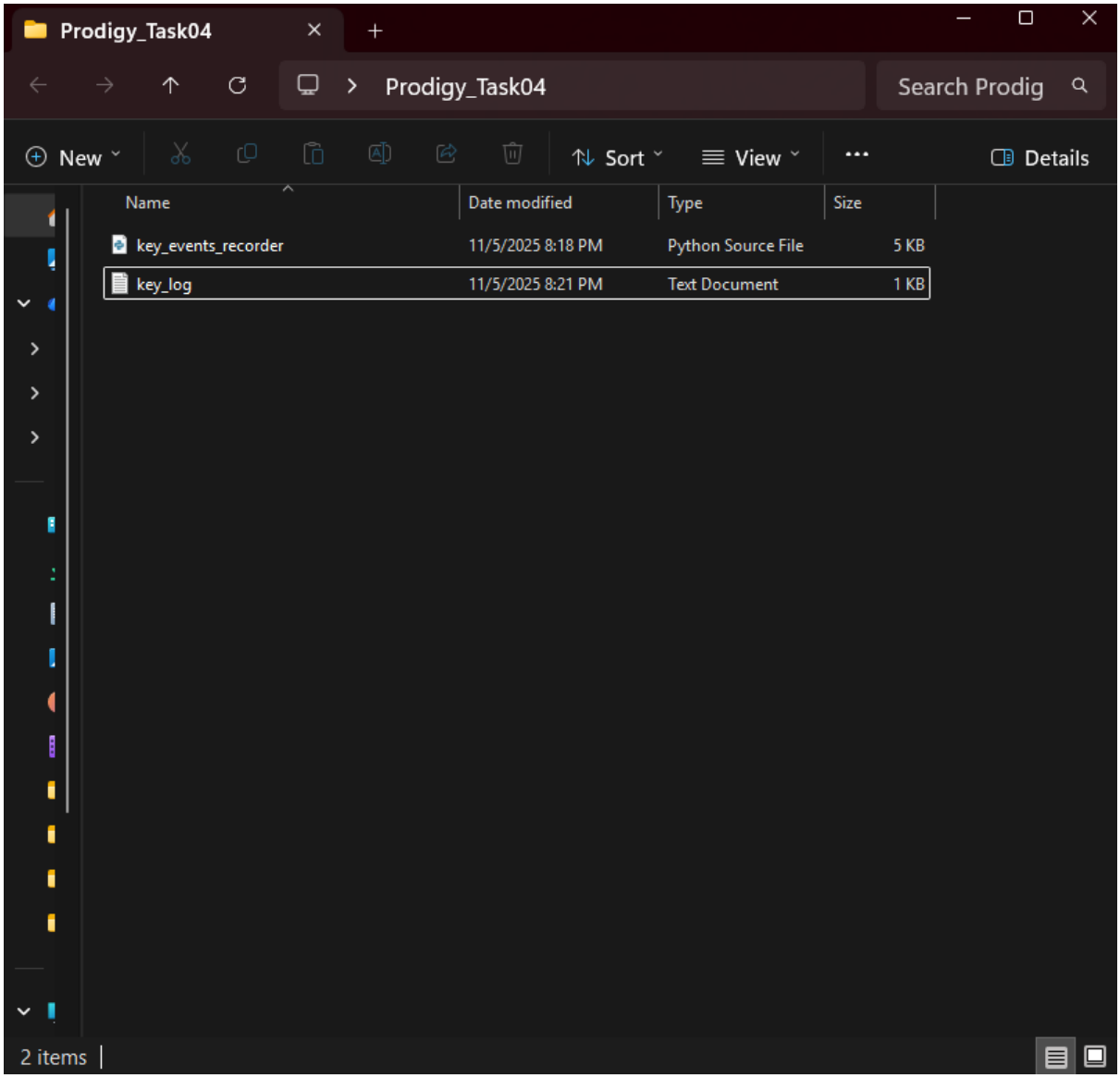
Title: Task-04 — Consent-based Keystroke Recorder (Window-Only)
Objective: Log keys pressed within a focused app window and save to a file with explicit consent and start/stop controls.
Ethics & Scope: Recording is limited to this window; shows consent checkbox and visible status; never record other users or apps without permission.
Implementation: Python tkinter; binds <KeyPress> on start; writes timestamped events to key_log.txt; unbinds on stop.
Screenshots:




File Edit Selection View Go Run ···
Prodigy_Task02
Welcome
image_encrypt_decrypt.py
password_checker.py
key_events_recorder.py
C: > Users > Abhinandan > Desktop > Prodigy_Task04 > key_events_recorder.py > ...
class KeyRecorderApp:
def __init__(self, root):
"• Recording works ONLY while this window is focused.\n"
"• Purpose: educational demo with explicit consent.\n"
"• Do not use to record other apps or people without permission."
tk.Label(root, text=info, justify="left", fg="#666").pack(padx=10, pady=(0,10))
# Key binding (bound when recording)
self.root.protocol("WM_DELETE_WINDOW", self.on_close)
def change_path(self):
path = filedialog.asksaveasfilename(
title="Choose log file",
defaultextension=".txt",
filetypes=[("Text files","*.txt"), ("All files","*.*")]
)
if path:
self.log_path = path
self.lbl_path.config(text=path)
def start_recording(self):
if not self.var_consent.get():
messagebox.showwarning("Consent required", "Please tick the consent box before recording.")
return
self.is_recording = True
self.btn_start.config(state="disabled")
self.btn_stop.config(state="normal")
self.status.config(text="Status: Recording…", fg="green")
# Bind keypress only now
self.root.bind("<KeyPress>", self.on_key)
# Create/append file header
with open(self.log_path, "a", encoding="utf-8") as f:
f.write(f"\n--- Session start: {datetime.now().isoformat(timespec='seconds')} ---\n")
def stop_recording(self):
if self.is_recording:
self.is_recording = False
self.btn_start.config(state="normal")
self.btn_stop.config(state="disabled")
self.status.config(text="Status: Stopped", fg="#444")
# Unbind
self.root.unbind("<KeyPress>")
with open(self.log_path, "a", encoding="utf-8") as f:
f.write(f"--- Session end: {datetime.now().isoformat(timespec='seconds')} ---\n")
def on_key(self, event: tk.Event):
if not self.is_recording:
return
key_sym = event.keysym   # e.g., a, A, space, Return
key_code = event.keycode   # platform-specific code
ts = datetime.now().strftime("%H:%M:%S")
line = f"[{ts}] key={key_sym!r} code={key_code}\n"
# Append to file
with open(self.log_path, "a", encoding="utf-8") as f:
f.write(line)
# Append to on-screen feed
self.feed.config(state="normal")
self.feed.insert("end", line)
self.feed.see("end")
self.feed.config(state="disabled")
def on_close(self):
self.stop_recording()
self.root.destroy()
if __name__ == "__main__":
root = tk.Tk()
app = KeyRecorderApp(root)
root.mainloop()


Task-04: Key Event Recorder (Window-only)
Key Event Recorder (Window-only, with consent)
I have explicit permission to record keys typed in THIS window.
Start Recording
Stop Recording
Log file: key_log.txt
Change...
Status: Recording...
[20:20:30] key='l' code=76
[20:20:30] key='l' code=76
[20:20:30] key='o' code=79
[20:20:31] key='space' code=32
[20:20:32] key='1' code=97
[20:20:32] key='2' code=98
[20:20:32] key='3' code=99
[20:20:33] key='Return' code=13
[20:20:34] key='BackSpace' code=8
Notes:
• Recording works ONLY while this window is focused.
• Purpose: educational demo with explicit consent.
• Do not use to record other apps or people without permission.


Label1.jav
List1.java
Choice1.ja
Untitled
impo
key_
File
Edit
View
H1
--- Session start: 2025-11-05T20:20:19 ---
[20:20:26] key='h' code=72
[20:20:27] key='e' code=69
[20:20:30] key='l' code=76
[20:20:30] key='l' code=76
[20:20:30] key='o' code=79
[20:20:31] key='space' code=32
[20:20:32] key='1' code=97
[20:20:32] key='2' code=98
[20:20:32] key='3' code=99
[20:20:33] key='Return' code=13
[20:20:34] key='BackSpace' code=8
--- Session end: 2025-11-05T20:20:37 ---
--- Session start: 2025-11-05T20:21:16 ---
--- Session end: 2025-11-05T20:21:27 ---
Ln 1, Col 1
487 characters
Plain text
100%
Windows (CRLF)
UTF-8



Conclusion: Demonstrated safe, consent-based keystroke logging for educational purposes.