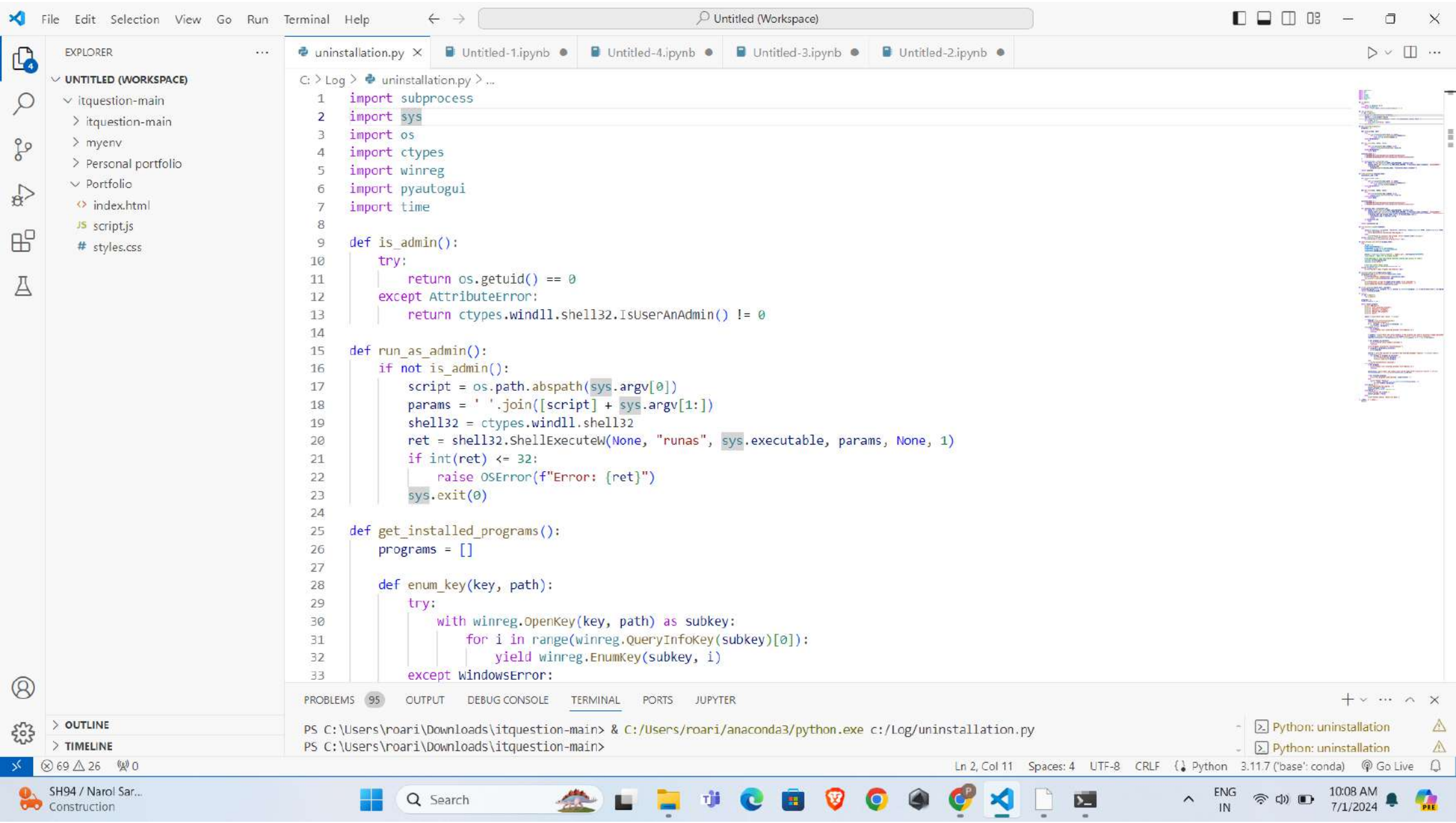
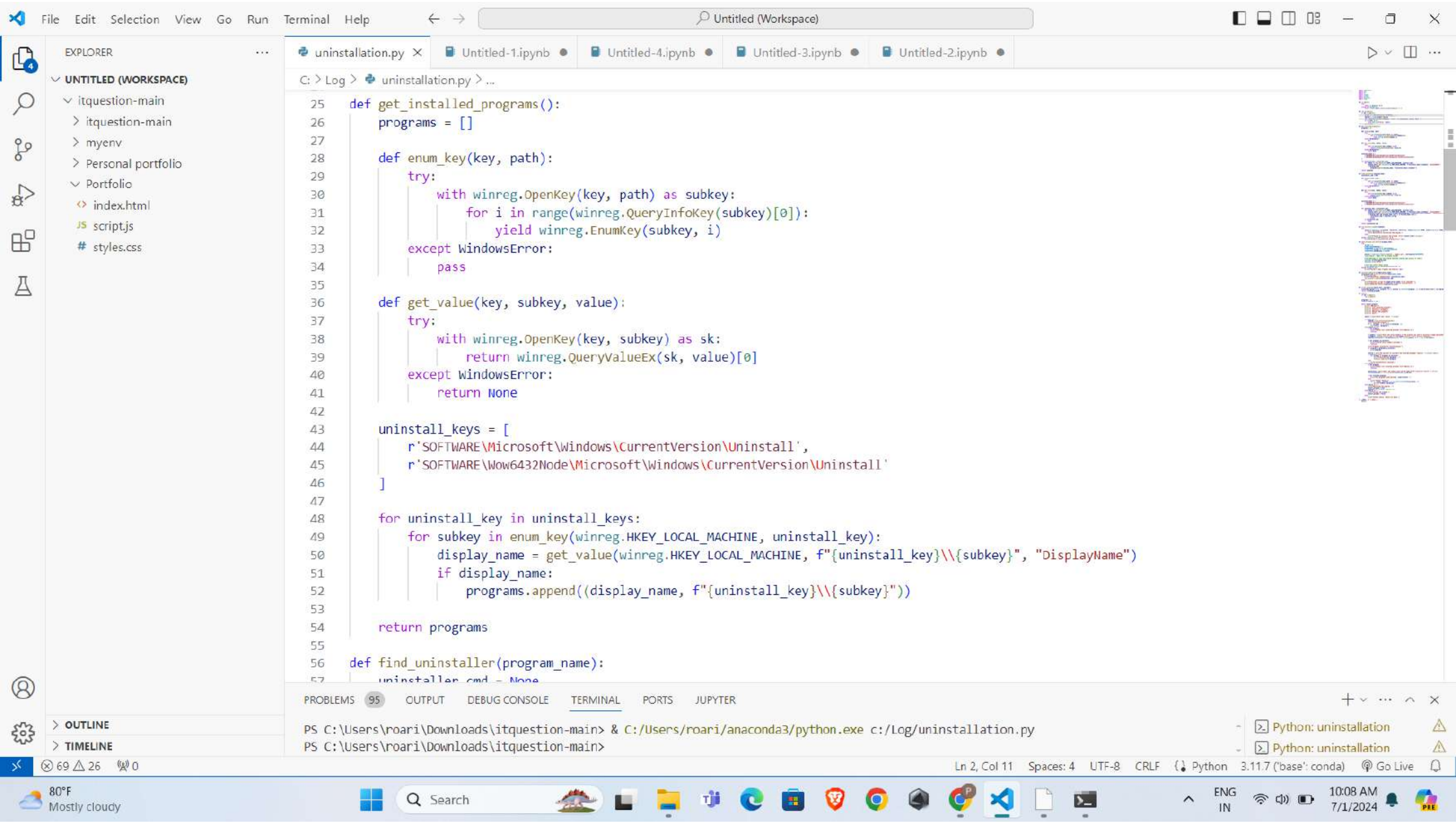
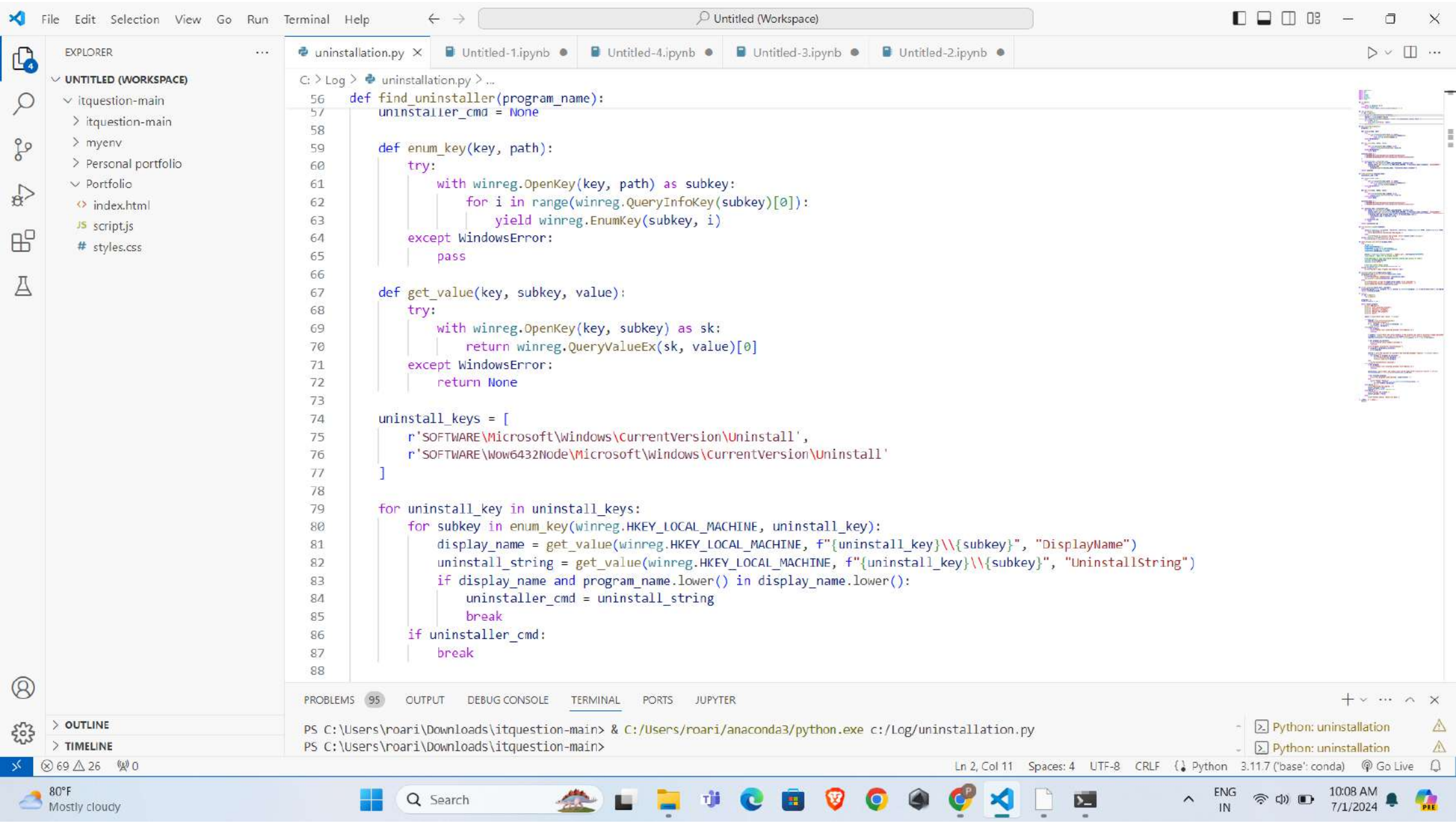
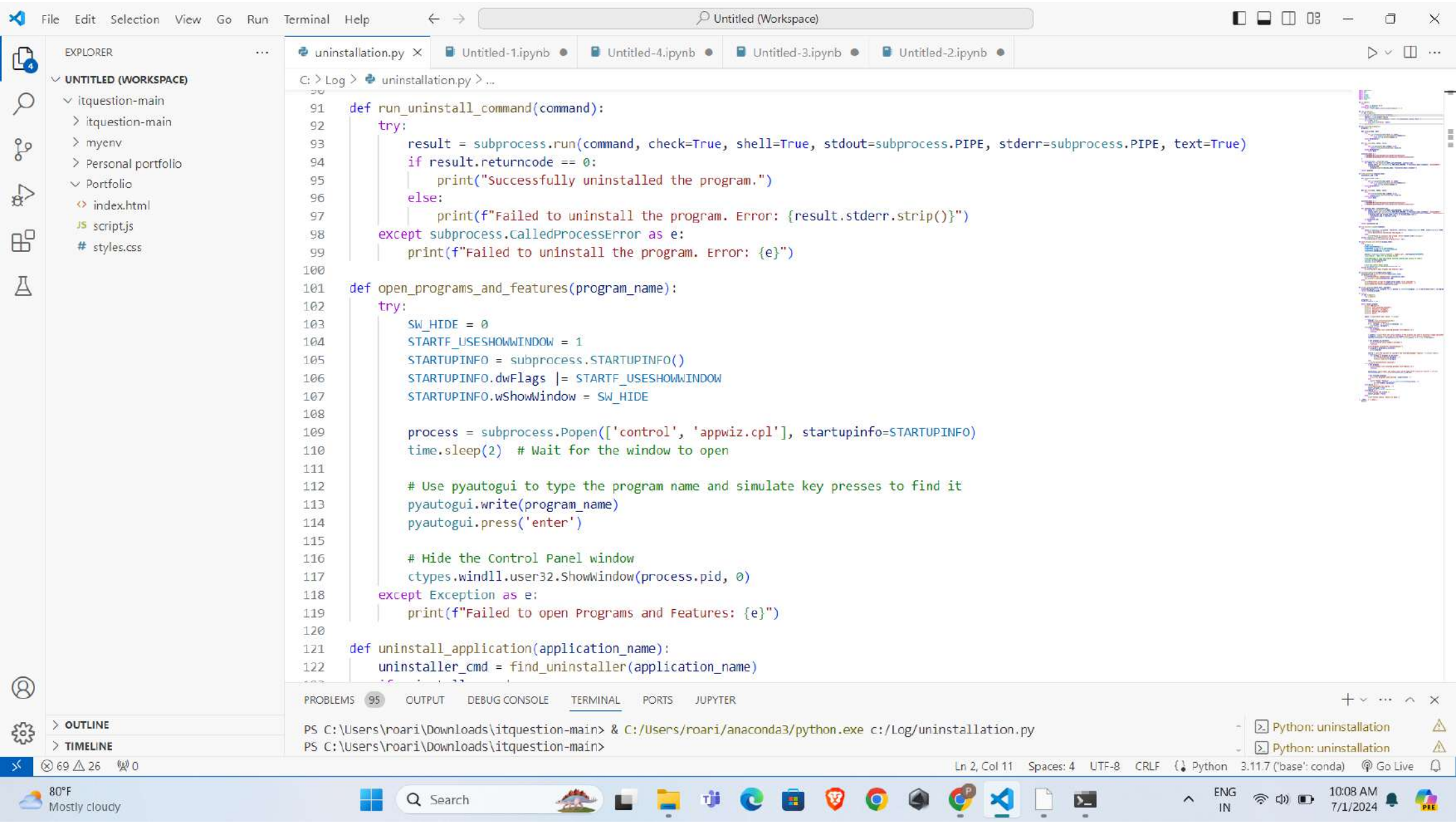
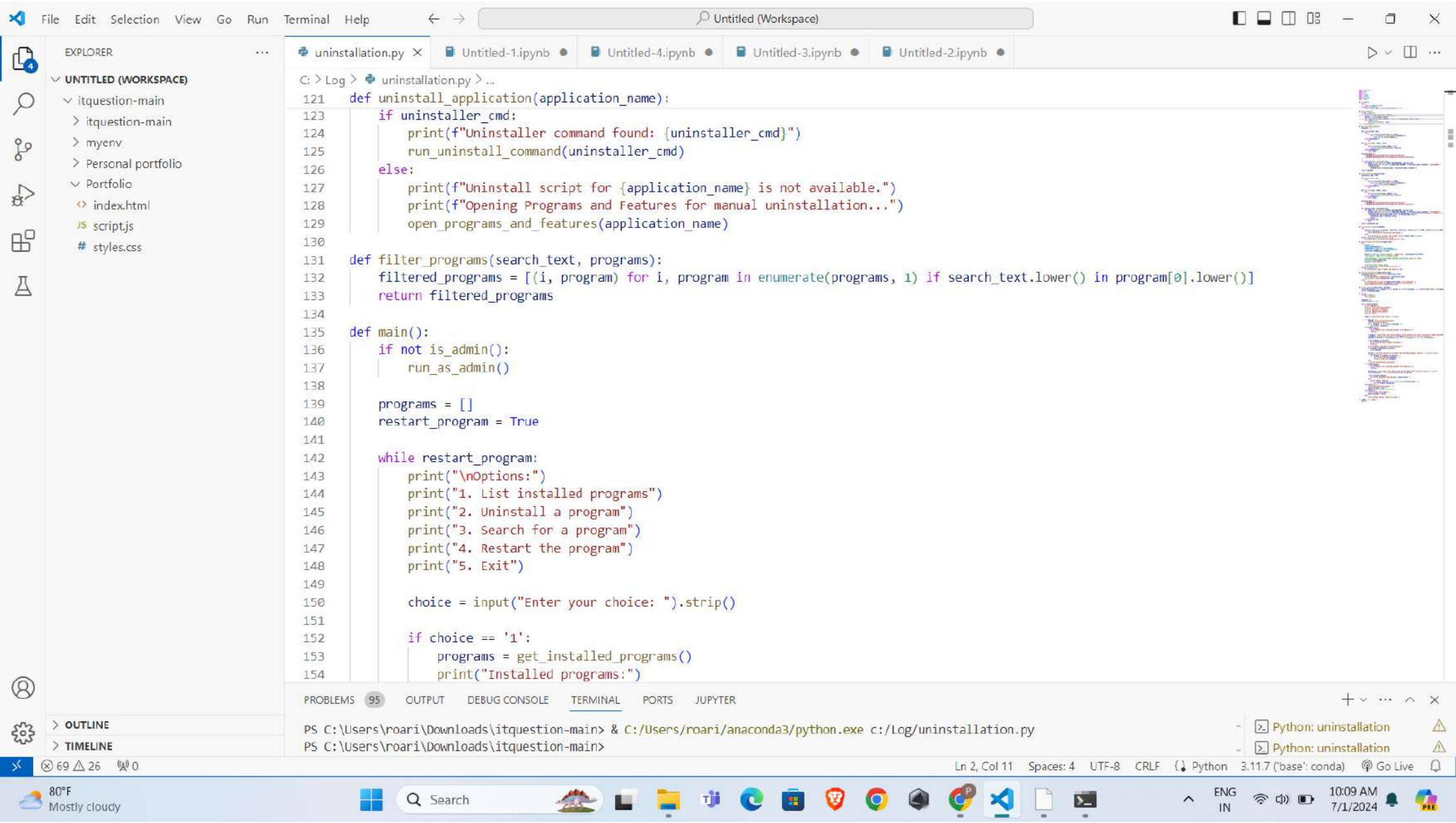
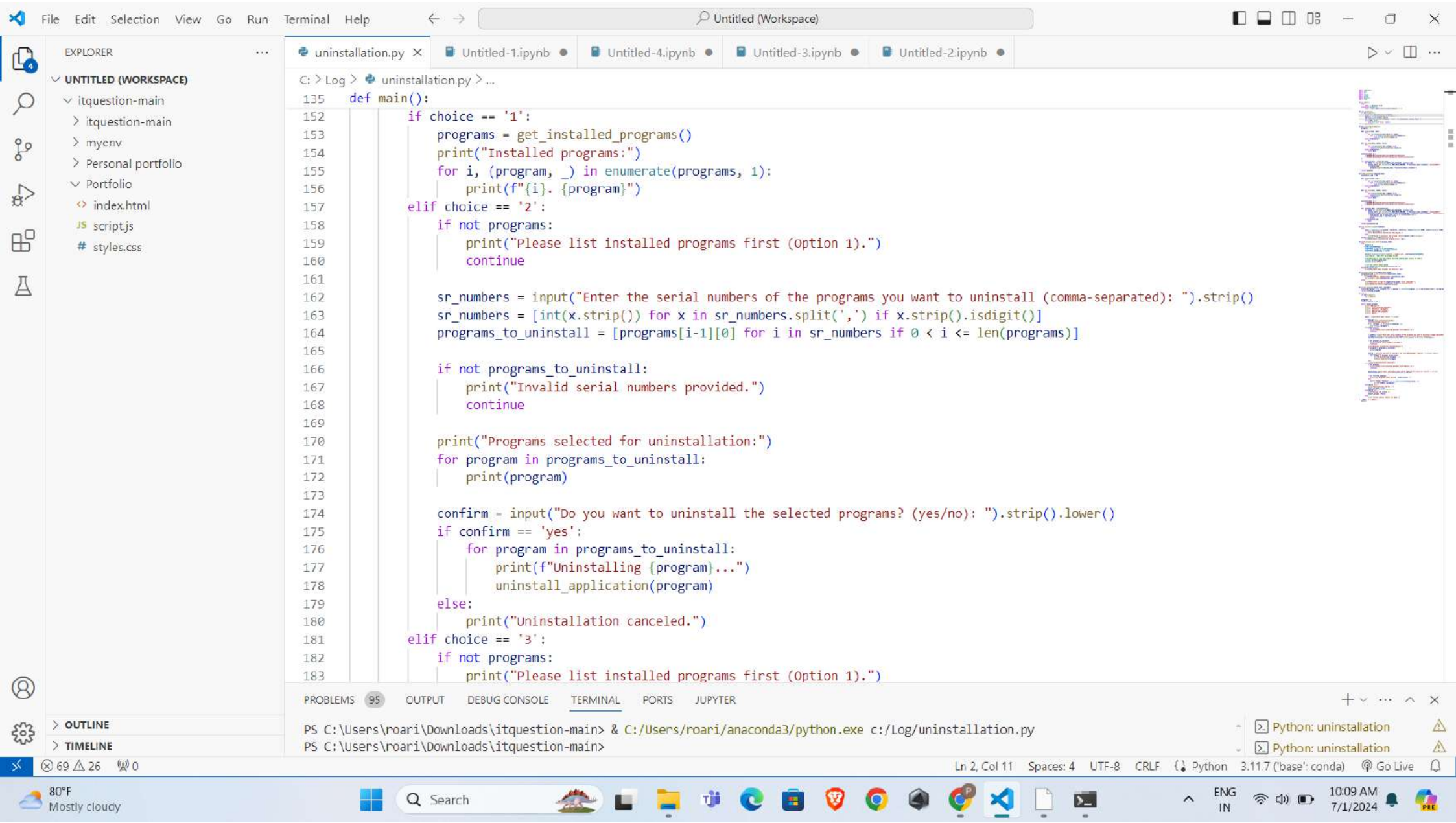

File Edit Selection View Go Run Terminal Help
Untitled (Workspace)
EXPLORER
UNTITLED (WORKSPACE)
itquestion-main
itquestion-main
myenv
Personal portfolio
Portfolio
index.html
script.js
styles.css
uninstallation.py
Untitled-1.ipynb
Untitled-4.ipynb
Untitled-3.ipynb
Untitled-2.ipynb
C: > Log > uninstallation.py > ...
import subprocess
import sys
import os
import ctypes
import winreg
import pyautogui
import time
def is_admin():
try:
return os.getuid() == 0
except AttributeError:
return ctypes.windll.shell32.IsUserAnAdmin() != 0
def run_as_admin():
if not is_admin():
script = os.path.abspath(sys.argv[0])
params = ' '.join([script] + sys.argv[1:])
shell32 = ctypes.windll.shell32
ret = shell32.ShellExecuteW(None, "runas", sys.executable, params, None, 1)
if int(ret) <= 32:
raise OSError(f"Error: {ret}")
sys.exit(0)
def get_installed_programs():
programs = []
def enum_key(key, path):
try:
with winreg.OpenKey(key, path) as subkey:
for i in range(winreg.QueryInfoKey(subkey)[0]):
yield winreg.EnumKey(subkey, i)
except WindowsError:
PROBLEMS 95
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
JUPYTER
PS C:\Users\roari\Downloads\itquestion-main> & C:/Users/roari/anaconda3/python.exe c:/Log/uninstallation.py
PS C:\Users\roari\Downloads\itquestion-main>
Python: uninstallation
Python: uninstallation
OUTLINE
TIMELINE
69 26 0
Ln 2, Col 11 Spaces: 4 UTF-8 CRLF Python 3.11.7 ('base': conda) Go Live
SH94 / Narol Sar...
Construction
Search
ENG
IN
10:08 AM
7/1/2024

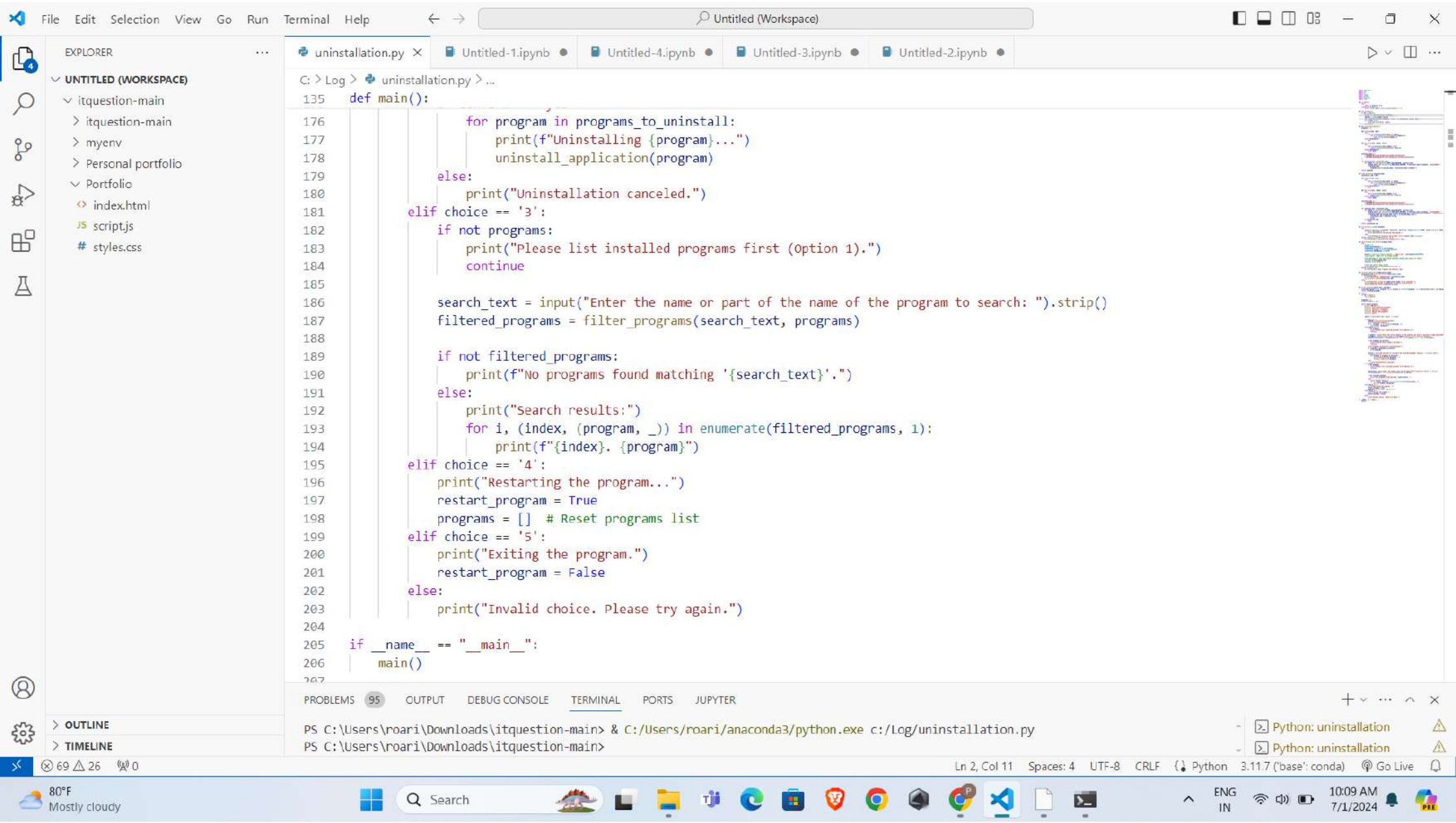

File Edit Selection View Go Run Terminal Help
Untitled (Workspace)
EXPLORER
UNTITLED (WORKSPACE)
itquestion-main
itquestion-main
myenv
Personal portfolio
Portfolio
index.html
script.js
styles.css
uninstallation.py
Untitled-1.ipynb
Untitled-4.ipynb
Untitled-3.ipynb
Untitled-2.ipynb
C: > Log > uninstallation.py > ...
def get_installed_programs():
programs = []
def enum_key(key, path):
try:
with winreg.OpenKey(key, path) as subkey:
for i in range(winreg.QueryInfoKey(subkey)[0]):
yield winreg.EnumKey(subkey, i)
except WindowsError:
pass
def get_value(key, subkey, value):
try:
with winreg.OpenKey(key, subkey) as sk:
return winreg.QueryValueEx(sk, value)[0]
except WindowsError:
return None
uninstall_keys = [
r'SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall',
r'SOFTWARE\Wow6432Node\Microsoft\Windows\CurrentVersion\Uninstall'
]
for uninstall_key in uninstall_keys:
for subkey in enum_key(winreg.HKEY_LOCAL_MACHINE, uninstall_key):
display_name = get_value(winreg.HKEY_LOCAL_MACHINE, f"{uninstall_key}\\{subkey}", "DisplayName")
if display_name:
programs.append((display_name, f"{uninstall_key}\\{subkey}"))
return programs
def find_uninstaller(program_name):
PROBLEMS 95 OUTPUT DEBUG CONSOLE TERMINAL PORTS JUPYTER
PS C:\Users\roari\Downloads\itquestion-main> & C:/Users/roari/anaconda3/python.exe c:/Log/uninstallation.py
PS C:\Users\roari\Downloads\itquestion-main>
Python: uninstallation
Python: uninstallation
OUTLINE
TIMELINE
69 26 0
Ln 2, Col 11 Spaces: 4 UTF-8 CRLF Python 3.11.7 ('base': conda) Go Live
80°F
Mostly cloudy
Search
ENG
IN
10:08 AM
7/1/2024


File Edit Selection View Go Run Terminal Help
Untitled (Workspace)
EXPLORER
UNTITLED (WORKSPACE)
itquestion-main
itquestion-main
myenv
Personal portfolio
Portfolio
index.html
script.js
styles.css
uninstallation.py
Untitled-1.ipynb
Untitled-4.ipynb
Untitled-3.ipynb
Untitled-2.ipynb
C: > Log > uninstallation.py > ...
def find_uninstaller(program_name):
uninstaller_cmd = None
def enum_key(key, path):
try:
with winreg.OpenKey(key, path) as subkey:
for i in range(winreg.QueryInfoKey(subkey)[0]):
yield winreg.EnumKey(subkey, i)
except WindowsError:
pass
def get_value(key, subkey, value):
try:
with winreg.OpenKey(key, subkey) as sk:
return winreg.QueryValueEx(sk, value)[0]
except WindowsError:
return None
uninstall_keys = [
r'SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall',
r'SOFTWARE\Wow6432Node\Microsoft\Windows\CurrentVersion\Uninstall'
]
for uninstall_key in uninstall_keys:
for subkey in enum_key(winreg.HKEY_LOCAL_MACHINE, uninstall_key):
display_name = get_value(winreg.HKEY_LOCAL_MACHINE, f"{uninstall_key}\\{subkey}", "DisplayName")
uninstall_string = get_value(winreg.HKEY_LOCAL_MACHINE, f"{uninstall_key}\\{subkey}", "UninstallString")
if display_name and program_name.lower() in display_name.lower():
uninstaller_cmd = uninstall_string
break
if uninstaller_cmd:
break
PROBLEMS 95
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
JUPYTER
PS C:\Users\roari\Downloads\itquestion-main> & C:/Users/roari/anaconda3/python.exe c:/Log/uninstallation.py
PS C:\Users\roari\Downloads\itquestion-main>
Python: uninstallation
Python: uninstallation
OUTLINE
TIMELINE
69 26 0
Ln 2, Col 11 Spaces: 4 UTF-8 CRLF Python 3.11.7 ('base': conda) Go Live
80°F
Mostly cloudy
Search
ENG
IN
10:08 AM
7/1/2024


File Edit Selection View Go Run Terminal Help
Untitled (Workspace)
EXPLORER
UNTITLED (WORKSPACE)
itquestion-main
itquestion-main
myenv
Personal portfolio
Portfolio
index.html
script.js
styles.css
uninstallation.py
Untitled-1.ipynb
Untitled-4.ipynb
Untitled-3.ipynb
Untitled-2.ipynb
C: > Log > uninstallation.py > ...
def run_uninstall_command(command):
try:
result = subprocess.run(command, check=True, shell=True, stdout=subprocess.PIPE, stderr=subprocess.PIPE, text=True)
if result.returncode == 0:
print("Successfully uninstalled the program.")
else:
print(f"Failed to uninstall the program. Error: {result.stderr.strip()}")
except subprocess.CalledProcessError as e:
print(f"Failed to uninstall the program. Error: {e}")
def open_programs_and_features(program_name):
try:
SW_HIDE = 0
STARTF_USESHOWWINDOW = 1
STARTUPINFO = subprocess.STARTUPINFO()
STARTUPINFO.dwFlags |= STARTF_USESHOWWINDOW
STARTUPINFO.wShowWindow = SW_HIDE
process = subprocess.Popen(['control', 'appwiz.cpl'], startupinfo=STARTUPINFO)
time.sleep(2) # Wait for the window to open
# Use pyautogui to type the program name and simulate key presses to find it
pyautogui.write(program_name)
pyautogui.press('enter')
# Hide the Control Panel window
ctypes.windll.user32.ShowWindow(process.pid, 0)
except Exception as e:
print(f"Failed to open Programs and Features: {e}")
def uninstall_application(application_name):
uninstaller_cmd = find_uninstaller(application_name)
PROBLEMS 95 OUTPUT DEBUG CONSOLE TERMINAL PORTS JUPYTER
PS C:\Users\roari\Downloads\itquestion-main> & C:/Users/roari/anaconda3/python.exe c:/Log/uninstallation.py
PS C:\Users\roari\Downloads\itquestion-main>
Python: uninstallation
Python: uninstallation
OUTLINE
TIMELINE
69 26 0
Ln 2, Col 11 Spaces: 4 UTF-8 CRLF Python 3.11.7 ('base': conda) Go Live
80°F Mostly cloudy
Search
ENG IN
10:08 AM 7/1/2024


File Edit Selection View Go Run Terminal Help
Untitled (Workspace)
EXPLORER
UNTITLED (WORKSPACE)
itquestion-main
itquestion-main
myenv
Personal portfolio
Portfolio
index.html
script.js
styles.css
uninstallation.py
Untitled-1.ipynb
Untitled-4.ipynb
Untitled-3.ipynb
Untitled-2.ipynb
C: > Log > uninstallation.py > ...
def uninstall_application(application_name):
if uninstaller_cmd:
print(f"Uninstaller command found: {uninstaller_cmd}")
run_uninstall_command(uninstaller_cmd)
else:
print(f"Uninstall script for {application_name} is not available.")
print(f"Opening Programs and Features for manual uninstallation...")
open_programs_and_features(application_name)
def filter_programs(search_text, programs):
filtered_programs = [(i, program) for i, program in enumerate(programs, 1) if search_text.lower() in program[0].lower()]
return filtered_programs
def main():
if not is_admin():
run_as_admin()
programs = []
restart_program = True
while restart_program:
print("\nOptions:")
print("1. List installed programs")
print("2. Uninstall a program")
print("3. Search for a program")
print("4. Restart the program")
print("5. Exit")
choice = input("Enter your choice: ").strip()
if choice == '1':
programs = get_installed_programs()
print("Installed programs:")
PROBLEMS 95 OUTPUT DEBUG CONSOLE TERMINAL PORTS JUPYTER
PS C:\Users\roari\Downloads\itquestion-main> & C:/Users/roari/anaconda3/python.exe c:/Log/uninstallation.py
PS C:\Users\roari\Downloads\itquestion-main>
Python: uninstallation
Python: uninstallation
OUTLINE
TIMELINE
69 26 0
Ln 2, Col 11 Spaces: 4 UTF-8 CRLF Python 3.11.7 ('base': conda) Go Live
80°F
Mostly cloudy
Search
ENG
IN
10:09 AM
7/1/2024


File Edit Selection View Go Run Terminal Help
Untitled (Workspace)
EXPLORER
UNTITLED (WORKSPACE)
itquestion-main
itquestion-main
myenv
Personal portfolio
Portfolio
index.html
script.js
styles.css
uninstallation.py
Untitled-1.ipynb
Untitled-4.ipynb
Untitled-3.ipynb
Untitled-2.ipynb
C: > Log > uninstallation.py > ...
def main():
if choice == '1':
programs = get_installed_programs()
print("Installed programs:")
for i, (program, _) in enumerate(programs, 1):
print(f"{i}. {program}")
elif choice == '2':
if not programs:
print("Please list installed programs first (Option 1).")
continue
sr_numbers = input("Enter the serial numbers of the programs you want to uninstall (comma-separated): ").strip()
sr_numbers = [int(x.strip()) for x in sr_numbers.split(',') if x.strip().isdigit()]
programs_to_uninstall = [programs[i-1][0] for i in sr_numbers if 0 < i <= len(programs)]
if not programs_to_uninstall:
print("Invalid serial numbers provided.")
continue
print("Programs selected for uninstallation:")
for program in programs_to_uninstall:
print(program)
confirm = input("Do you want to uninstall the selected programs? (yes/no): ").strip().lower()
if confirm == 'yes':
for program in programs_to_uninstall:
print(f"Uninstalling {program}...")
uninstall_application(program)
else:
print("Uninstallation canceled.")
elif choice == '3':
if not programs:
print("Please list installed programs first (Option 1).")
PROBLEMS 95 OUTPUT DEBUG CONSOLE TERMINAL PORTS JUPYTER
PS C:\Users\roari\Downloads\itquestion-main> & C:/Users/roari/anaconda3/python.exe c:/Log/uninstallation.py
PS C:\Users\roari\Downloads\itquestion-main>
Python: uninstallation
Python: uninstallation
OUTLINE
TIMELINE
Ln 2, Col 11 Spaces: 4 UTF-8 CRLF Python 3.11.7 ('base': conda) Go Live
80°F Mostly cloudy
Search
ENG IN
10:09 AM 7/1/2024


File Edit Selection View Go Run Terminal Help
Untitled (Workspace)
EXPLORER
UNTITLED (WORKSPACE)
itquestion-main
itquestion-main
myenv
Personal portfolio
Portfolio
index.html
script.js
styles.css
uninstallation.py
Untitled-1.ipynb
Untitled-4.ipynb
Untitled-3.ipynb
Untitled-2.ipynb
C: > Log > uninstallation.py > ...
def main():
for program in programs_to_uninstall:
print(f"Uninstalling {program}...")
uninstall_application(program)
else:
print("Uninstallation canceled.")
elif choice == '3':
if not programs:
print("Please list installed programs first (Option 1).")
continue
search_text = input("Enter the name or part of the name of the program to search: ").strip()
filtered_programs = filter_programs(search_text, programs)
if not filtered_programs:
print(f"No programs found matching '{search_text}'.")
else:
print("Search results:")
for i, (index, (program, _)) in enumerate(filtered_programs, 1):
print(f"{index}. {program}")
elif choice == '4':
print("Restarting the program...")
restart_program = True
programs = [] # Reset programs list
elif choice == '5':
print("Exiting the program.")
restart_program = False
else:
print("Invalid choice. Please try again.")
if __name__ == "__main__":
main()
PROBLEMS 95
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
JUPYTER
PS C:\Users\roari\Downloads\itquestion-main> & C:/Users/roari/anaconda3/python.exe c:/Log/uninstallation.py
PS C:\Users\roari\Downloads\itquestion-main>
Python: uninstallation
Python: uninstallation
OUTLINE
TIMELINE
69 26 0
Ln 2, Col 11 Spaces: 4 UTF-8 CRLF Python 3.11.7 ('base': conda) Go Live
80°F
Mostly cloudy
Search
ENG
IN
10:09 AM
7/1/2024

C:\Users\roari\anaconda3\python.exe

```
Options:
1. List installed programs
2. Uninstall a program
3. Search for a program
4. Restart the program
5. Exit
Enter your choice: 1
Installed programs:
1. Microsoft Office Home and Student 2021 - en-us
2. McAfee
3. Microsoft SQL Server 2022 (64-bit)
4. Microsoft SQL Server 2022 (64-bit)
5. Microsoft OneDrive
6. Microsoft OneNote - en-us
7. PostgreSQL 16
8. Proton VPN
9. psqlODBC 13.02.0000
10. R for Windows 4.3.2
11. WinRAR 7.00 (64-bit)
12. XAMPP
13. ASUS Ambient HAL
14. Microsoft Visual C++ 2013 x64 Additional Runtime - 12.0.40664
15. ARMOURY CRATE Service
16. ASUS Aac_NBDT HAL
17. Microsoft .NET Runtime - 6.0.31 (x64)
18. MySQL Shell 8.0.34
19. SQL Server 2022 SQL Diagnostics
20. Microsoft ODBC Driver 17 for SQL Server
21. ASUS Keyboard HAL
22. SQL Server 2022 Shared Management Objects
23. SQL Server 2022 Database Engine Shared
24. ASUS Hotplug Controller
25. Java 8 Update 202 (64-bit)
26. MySQL Server 8.0
27. ROG Live Service
28. SQL Server 2022 Shared Management Objects Extensions
29. Microsoft SQL Server 2012 Native Client
30. Python 3.12.4 Core Interpreter (64-bit)
31. Microsoft OLE DB Driver for SQL Server
32. Microsoft Visual C++ 2013 x64 Minimum Runtime - 12.0.40664
33. Microsoft .NET Host - 6.0.31 (x64)
34. SQL Server 2022 DMF
35. Microsoft Visual C++ 2008 Redistributable - x64 9.0.30729.6161
36. Microsoft SQL Server 2022 RsFx Driver
37. Python 3.12.4 Standard Library (64-bit)
38. Java SE Development Kit 8 Update 202 (64-bit)
39. SQL Server 2022 Database Engine Services
40. Microsoft Visual C++ 2022 X64 Minimum Runtime - 14.30.30704
```

80°F
Mostly cloudy
Search
ENG
IN
10:10 AM
7/1/2024

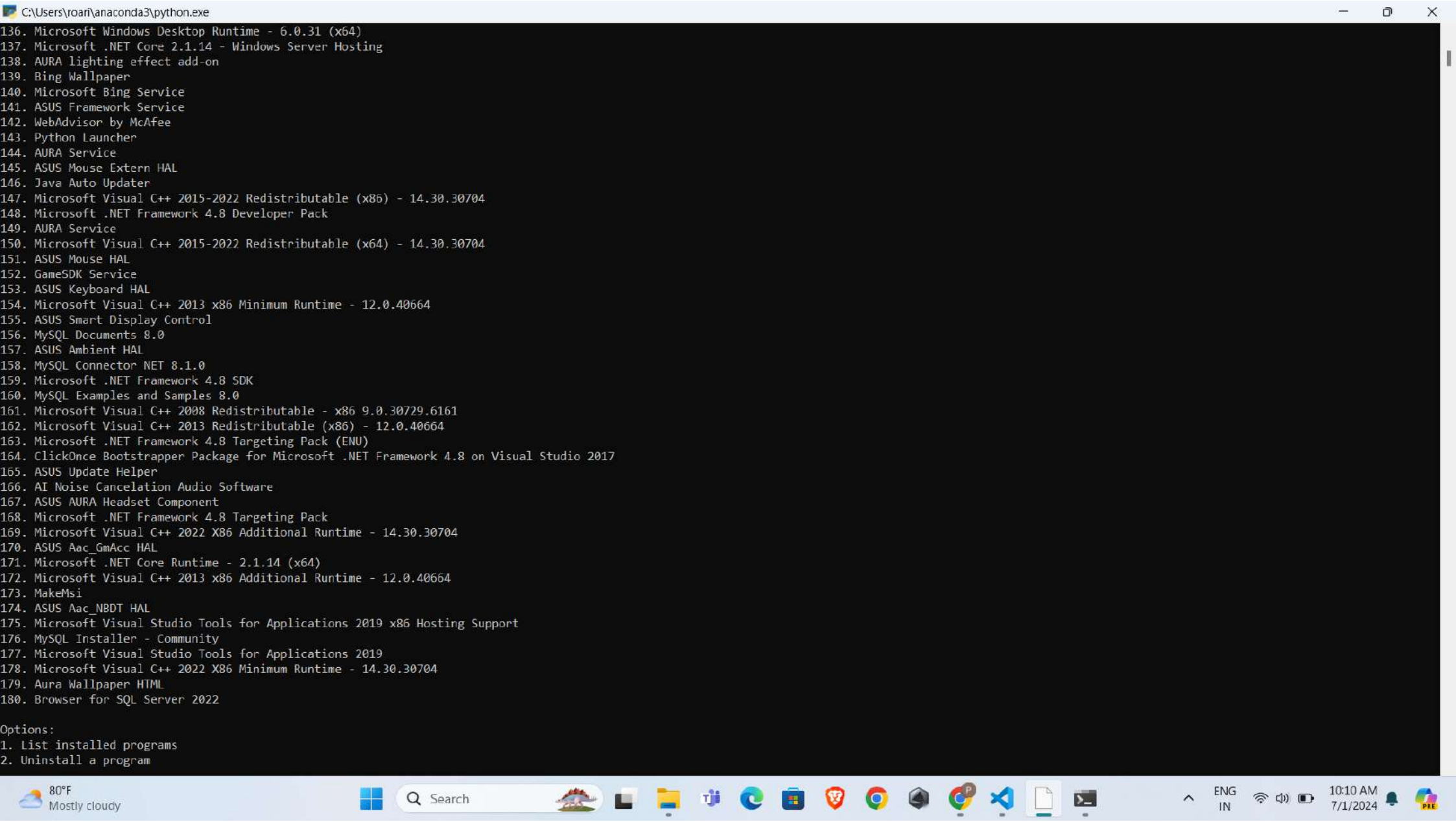
C:\Users\roari\anaconda3\python.exe

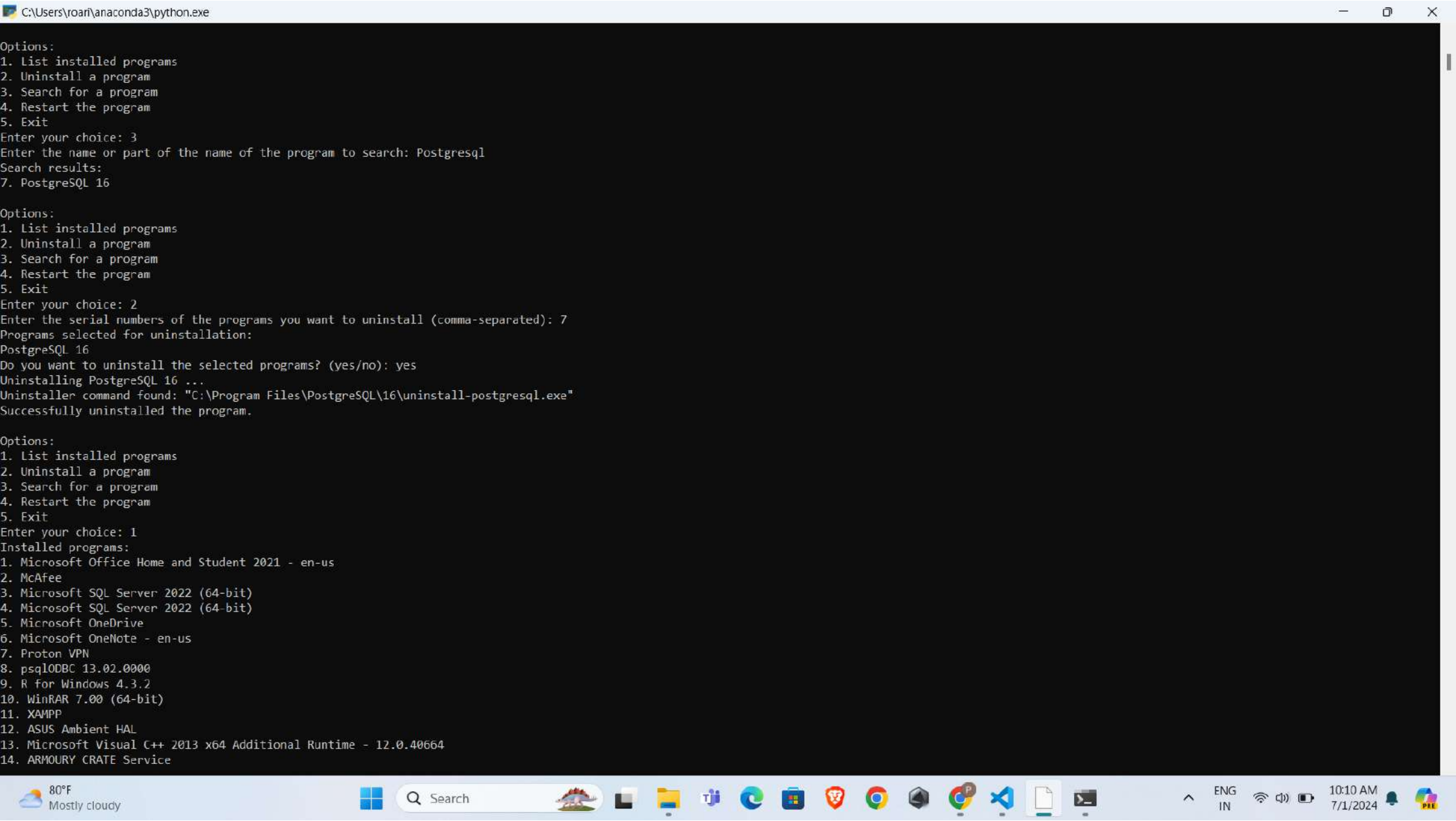
```
40. Microsoft Visual C++ 2022 X64 Minimum Runtime - 14.30.30704
41. SQL Server 2022 Common Files
42. Microsoft Visual C++ 2022 X64 Additional Runtime - 14.30.30704
43. SQL Server 2022 Shared Management Objects
44. MongoDB 7.0.11 2008R2Plus SSL (64 bit)
45. Python 3.12.4 Tcl/Tk Support (64-bit)
46. Python 3.12.4 Executables (64-bit)
47. Microsoft ODBC Driver 13 for SQL Server
48. SQL Server 2022 Connection Info
49. XLSTAT 2024
50. Python 3.12.4 Development Libraries (64-bit)
51. SQL Server 2022 Batch Parser
52. Beats filebeat 8.13.3 (x86_64)
53. SQL Server 2022 Common Files
54. Microsoft Visual Studio Tools for Applications 2019 x64 Hosting Support
55. Office 16 Click-to-Run Licensing Component
56. Office 16 Click-to-Run Extensibility Component
57. MySQL Router 8.0
58. SQL Server 2022 XEvent
59. ASUS Aac_GmAcc HAL
60. Microsoft .NET Host FX Resolver - 6.0.31 (x64)
61. AI Noise Cancelation Audio Software SDK
62. SQL Server 2022 Shared Management Objects Extensions
63. ASUS AURA Headset Component
64. Aura Wallpaper Service
65. Microsoft Teams Meeting Add-in for Microsoft Office
66. Microsoft VSS Writer for SQL Server 2022
67. Python 3.12.4 Test Suite (64-bit)
68. Python 3.12.4 Documentation (64-bit)
69. ASUS AURA Display Component
70. NVIDIA Graphics Driver 546.80
71. NVIDIA GeForce Experience 3.27.0.120
72. NVIDIA Optimus Update 39.5.0.0
73. NVIDIA PhysX System Software 9.21.0713
74. NVIDIA Update 39.5.0.0
75. NVIDIA FrameView SDK 1.3.8513.32290073
76. NVIDIA SHIELD Streaming
77. NVIDIA GPX Common OSS binaries (POCO, OpenSSL, libprotobuf)
78. NVIDIA Install Application
79. NVIDIA Backend
80. NVIDIA Container
81. NVIDIA TelemetryApi helper for NvContainer
82. NVIDIA LocalSystem Container
83. NVIDIA Message Bus for NvContainer
84. NVIDIA NVAPI Monitor plugin for NvContainer
85. NVIDIA NetworkService Container
86. NVIDIA Session Container
87. NVIDIA User Container
88. NVIDIA NvModuleTracker
```

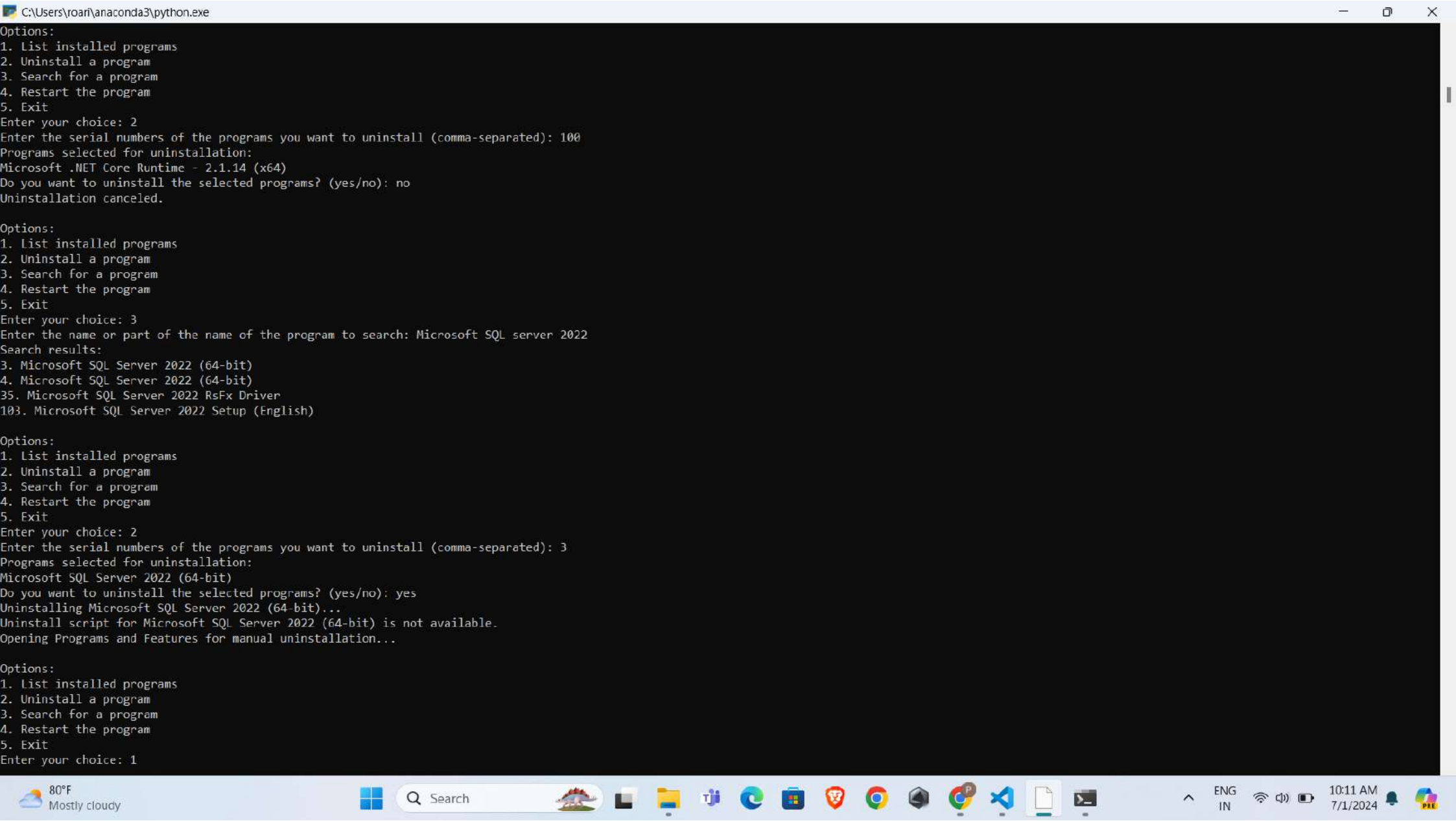
C:\Users\roari\anaconda3\python.exe

```
88. NVIDIA NvModuleTracker
89. NVIDIA NodeJS
90. NVIDIA Platform Controllers and Framework
91. NVIDIA Watchdog Plugin for NvContainer
92. NVIDIA Telemetry Client
93. NVIDIA Virtual Host Controller
94. Nvidia Share
95. NVIDIA ShadowPlay 3.27.0.120
96. NVIDIA SHIELD Wireless Controller Driver
97. NVIDIA Update Core
98. NVIDIA Virtual Audio 4.49.0.0
99. ASUS Mouse HAL
100. Virtual Pet
101. Microsoft .NET Core Runtime - 2.1.14 (x64)
102. ASUS Mouse Extern HAL
103. SQL Server 2022 XEvent
104. Microsoft SQL Server 2022 Setup (English)
105. ASUS MB Peripheral Products
106. SQL Server 2022 Database Engine Services
107. AURA lighting effect add-on x64
108. Microsoft Update Health Tools
109. Microsoft SQL Server 2017 T-SQL Language Service
110. EViews 12 Student Version (64-bit)
111. ASUS Aura SDK
112. Redis on Windows
113. SQL Server 2022 Database Engine Shared
114. SQL Server 2022 DMF
115. SQL Server 2022 Connection Info
116. Microsoft .NET Core Host - 2.1.14 (x64)
117. Quick Share from Google
118. Python 3.12.4 Add to Path (64-bit)
119. Python 3.12.4 pip Bootstrap (64-bit)
120. Microsoft Windows Desktop Runtime - 6.0.31 (x64)
121. MySQL Workbench 8.0 CE
122. Erlang OTP 23.3.1 (11.2)
123. Google Chrome
124. EViews 12 Student Version (64-bit)
125. Microsoft Edge
126. Microsoft Edge Update
127. Microsoft Edge WebView2 Runtime
128. RStudio
129. TallyPrime
130. Toolkit
131. GameSDK Service
132. Microsoft Visual C++ 2013 Redistributable (x64) - 12.0.40664
133. Microsoft Visual C++ 2013 Redistributable (x64) - 12.0.30501
134. ASUS AURA Display Component
135. ASUS MB Peripheral Products
136. Microsoft Windows Desktop Runtime - 6.0.31 (x64)
```

80°F
Mostly cloudy
Search
ENG
IN
10:10 AM
7/1/2024


C:\Users\roari\anaconda3\python.exe
136. Microsoft Windows Desktop Runtime - 6.0.31 (x64)
137. Microsoft .NET Core 2.1.14 - Windows Server Hosting
138. AURA lighting effect add-on
139. Bing Wallpaper
140. Microsoft Bing Service
141. ASUS Framework Service
142. WebAdvisor by McAfee
143. Python Launcher
144. AURA Service
145. ASUS Mouse Extern HAL
146. Java Auto Updater
147. Microsoft Visual C++ 2015-2022 Redistributable (x86) - 14.30.30704
148. Microsoft .NET Framework 4.8 Developer Pack
149. AURA Service
150. Microsoft Visual C++ 2015-2022 Redistributable (x64) - 14.30.30704
151. ASUS Mouse HAL
152. GameSDK Service
153. ASUS Keyboard HAL
154. Microsoft Visual C++ 2013 x86 Minimum Runtime - 12.0.40664
155. ASUS Smart Display Control
156. MySQL Documents 8.0
157. ASUS Ambient HAL
158. MySQL Connector NET 8.1.0
159. Microsoft .NET Framework 4.8 SDK
160. MySQL Examples and Samples 8.0
161. Microsoft Visual C++ 2008 Redistributable - x86 9.0.30729.6161
162. Microsoft Visual C++ 2013 Redistributable (x86) - 12.0.40664
163. Microsoft .NET Framework 4.8 Targeting Pack (ENU)
164. ClickOnce Bootstrapper Package for Microsoft .NET Framework 4.8 on Visual Studio 2017
165. ASUS Update Helper
166. AI Noise Cancelation Audio Software
167. ASUS AURA Headset Component
168. Microsoft .NET Framework 4.8 Targeting Pack
169. Microsoft Visual C++ 2022 X86 Additional Runtime - 14.30.30704
170. ASUS Aac_GmAcc HAL
171. Microsoft .NET Core Runtime - 2.1.14 (x64)
172. Microsoft Visual C++ 2013 x86 Additional Runtime - 12.0.40664
173. MakeMsi
174. ASUS Aac_NBDT HAL
175. Microsoft Visual Studio Tools for Applications 2019 x86 Hosting Support
176. MySQL Installer - Community
177. Microsoft Visual Studio Tools for Applications 2019
178. Microsoft Visual C++ 2022 X86 Minimum Runtime - 14.30.30704
179. Aura Wallpaper HTML
180. Browser for SQL Server 2022
Options:
1. List installed programs
2. Uninstall a program
80°F
Mostly cloudy
Search
ENG
IN
10:10 AM
7/1/2024


C:\Users\roari\anaconda3\python.exe
Options:
1. List installed programs
2. Uninstall a program
3. Search for a program
4. Restart the program
5. Exit
Enter your choice: 3
Enter the name or part of the name of the program to search: Postgresql
Search results:
7. PostgreSQL 16
Options:
1. List installed programs
2. Uninstall a program
3. Search for a program
4. Restart the program
5. Exit
Enter your choice: 2
Enter the serial numbers of the programs you want to uninstall (comma-separated): 7
Programs selected for uninstallation:
PostgreSQL 16
Do you want to uninstall the selected programs? (yes/no): yes
Uninstalling PostgreSQL 16 ...
Uninstaller command found: "C:\Program Files\PostgreSQL\16\uninstall-postgresql.exe"
Successfully uninstalled the program.
Options:
1. List installed programs
2. Uninstall a program
3. Search for a program
4. Restart the program
5. Exit
Enter your choice: 1
Installed programs:
1. Microsoft Office Home and Student 2021 - en-us
2. McAfee
3. Microsoft SQL Server 2022 (64-bit)
4. Microsoft SQL Server 2022 (64-bit)
5. Microsoft OneDrive
6. Microsoft OneNote - en-us
7. Proton VPN
8. psqlODBC 13.02.0000
9. R for Windows 4.3.2
10. WinRAR 7.00 (64-bit)
11. XAMPP
12. ASUS Ambient HAL
13. Microsoft Visual C++ 2013 x64 Additional Runtime - 12.0.40664
14. ARMOURY CRATE Service
80°F
Mostly cloudy
Search
ENG
IN
10:10 AM
7/1/2024


C:\Users\roari\anaconda3\python.exe
Options:
1. List installed programs
2. Uninstall a program
3. Search for a program
4. Restart the program
5. Exit
Enter your choice: 2
Enter the serial numbers of the programs you want to uninstall (comma-separated): 100
Programs selected for uninstallation:
Microsoft .NET Core Runtime - 2.1.14 (x64)
Do you want to uninstall the selected programs? (yes/no): no
Uninstallation canceled.
Options:
1. List installed programs
2. Uninstall a program
3. Search for a program
4. Restart the program
5. Exit
Enter your choice: 3
Enter the name or part of the name of the program to search: Microsoft SQL server 2022
Search results:
3. Microsoft SQL Server 2022 (64-bit)
4. Microsoft SQL Server 2022 (64-bit)
35. Microsoft SQL Server 2022 RsFx Driver
103. Microsoft SQL Server 2022 Setup (English)
Options:
1. List installed programs
2. Uninstall a program
3. Search for a program
4. Restart the program
5. Exit
Enter your choice: 2
Enter the serial numbers of the programs you want to uninstall (comma-separated): 3
Programs selected for uninstallation:
Microsoft SQL Server 2022 (64-bit)
Do you want to uninstall the selected programs? (yes/no): yes
Uninstalling Microsoft SQL Server 2022 (64-bit)...
Uninstall script for Microsoft SQL Server 2022 (64-bit) is not available.
Opening Programs and Features for manual uninstallation...
Options:
1. List installed programs
2. Uninstall a program
3. Search for a program
4. Restart the program
5. Exit
Enter your choice: 1
80°F
Mostly cloudy
Search
ENG
IN
10:11 AM
7/1/2024