



TECHNIK NEST

INNOVATIVE MINDS, NESTING SUCCESS

Name: Muhammad Zuhair

Intern ID: TN/IN02/PY/038

Email ID: mzuhairataalivi143@gmail.com

Task week: 01

Internship Domain: Python language

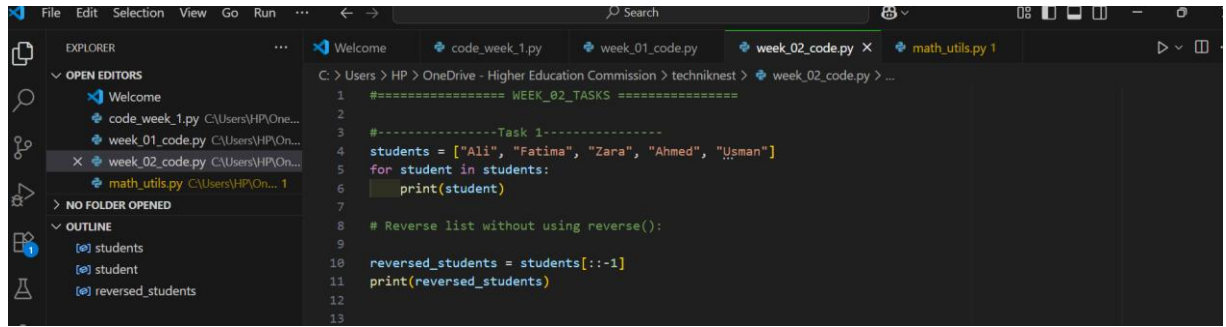
Instructor Name: Mr. Hassan Ali

Tasks

Task 01:

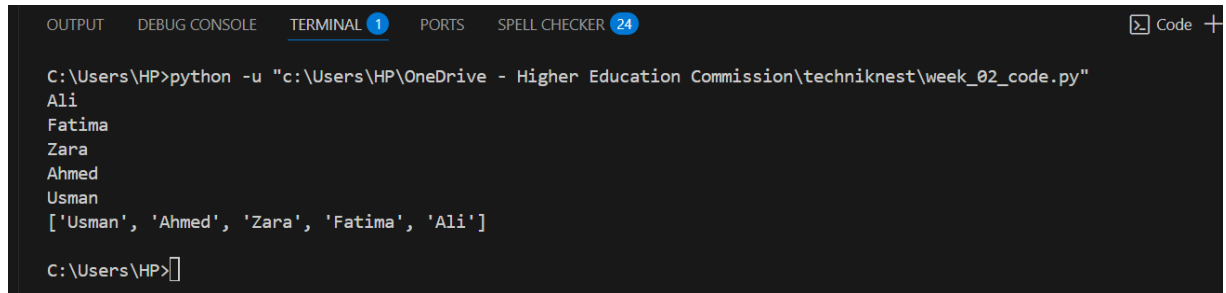
- 1. Store 5 student names & print each.**
- 2. Reverse list without reverse().**

code:



```
1 #===== WEEK_02_TASKS =====
2
3 #-----Task 1-----
4 students = ["Ali", "Fatima", "Zara", "Ahmed", "Usman"]
5 for student in students:
6     print(student)
7
8 # Reverse list without using reverse():
9
10 reversed_students = students[::-1]
11 print(reversed_students)
12
13
```

Output:

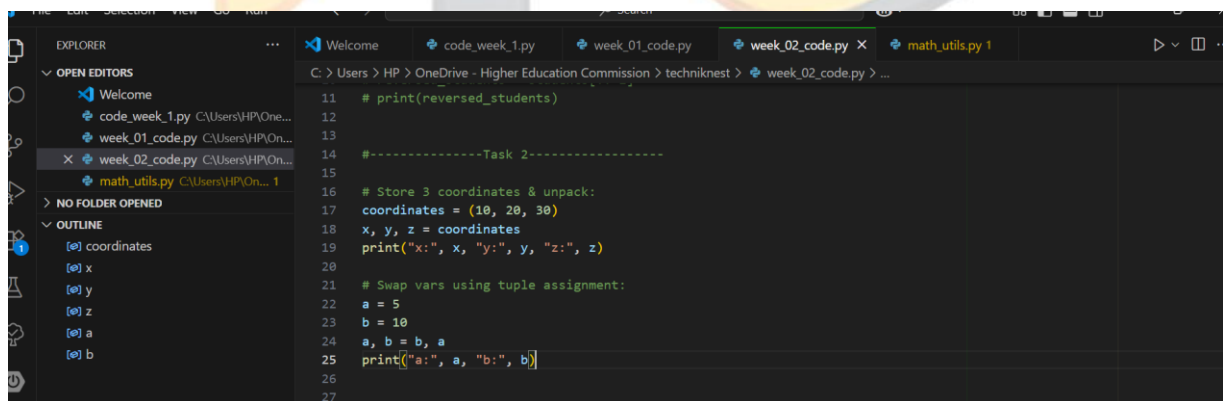


```
OUTPUT  DEBUG CONSOLE  TERMINAL 1  PORTS  SPELL CHECKER 24  Code +
C:\Users\HP>python -u "c:\Users\HP\OneDrive - Higher Education Commission\techniknest\week_02_code.py"
Ali
Fatima
Zara
Ahmed
Usman
['Usman', 'Ahmed', 'Zara', 'Fatima', 'Ali']
C:\Users\HP>
```

Task 02:

1. Store 3 coordinates & unpack.
2. Swap vars using tuple assignment.

Code:



```
11 # print(reversed_students)
12
13
14 #-----Task 2-----
15
16 # Store 3 coordinates & unpack:
17 coordinates = (10, 20, 30)
18 x, y, z = coordinates
19 print("x:", x, "y:", y, "z:", z)
20
21 # Swap vars using tuple assignment:
22 a = 5
23 b = 10
24 a, b = b, a
25 print(("a:", a, "b:", b))
26
27
```

Output:

```
OUTPUT  DEBUG CONSOLE  TERMINAL 1  PORTS  SPELL CHECKER 24  Code + v

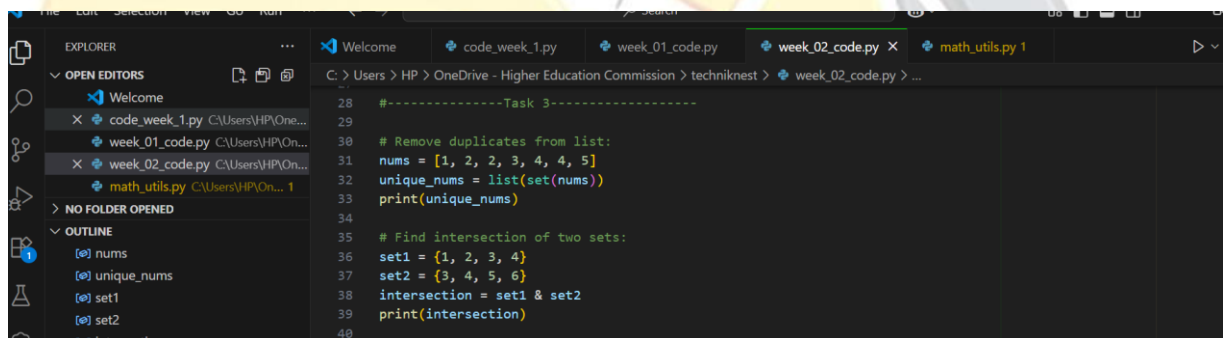
C:\Users\HP>python -u "c:\Users\HP\OneDrive - Higher Education Commission\techniknest\week_02_code.py"
x: 10 y: 20 z: 30
a: 10 b: 5

C:\Users\HP>
```

Task 03:

1. Remove duplicates from list.
2. Find intersection of two sets.

Code:



```
28 #-----Task 3-----
29
30 # Remove duplicates from list:
31 nums = [1, 2, 2, 3, 4, 4, 5]
32 unique_nums = list(set(nums))
33 print(unique_nums)
34
35 # Find intersection of two sets:
36 set1 = {1, 2, 3, 4}
37 set2 = {3, 4, 5, 6}
38 intersection = set1 & set2
39 print(intersection)
40
```

Output:

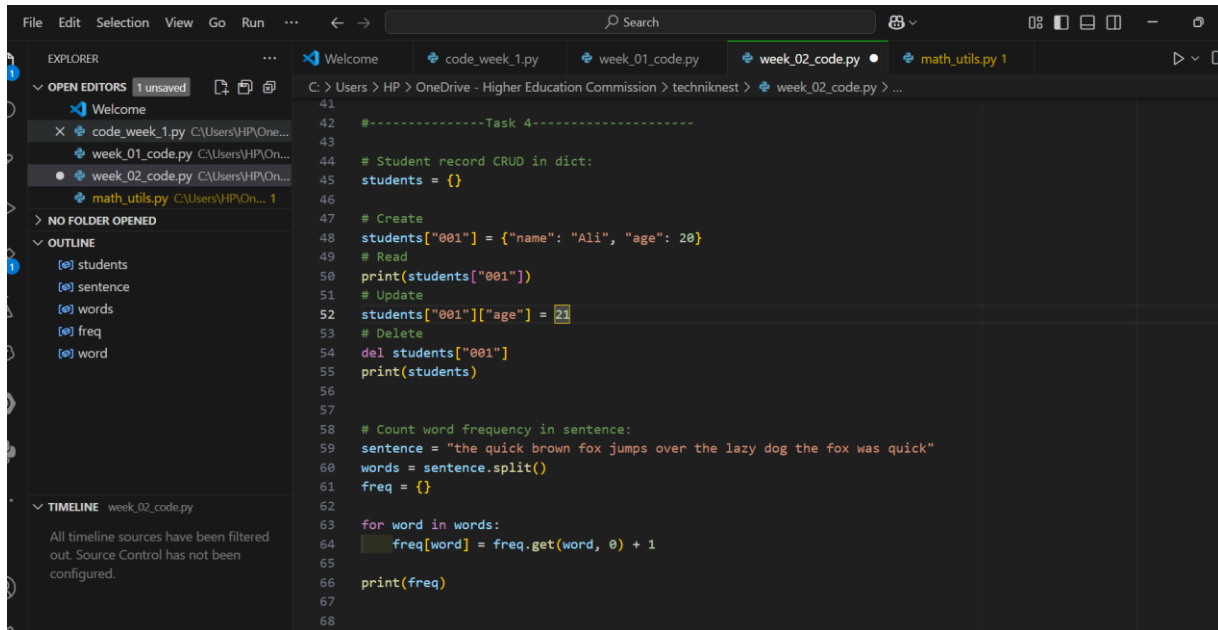
```
C:\Users\HP>python -u "c:\Users\HP\OneDrive - Higher Education Commission\techniknest\week_02_code.py"
[1, 2, 3, 4, 5]
{3, 4}

C:\Users\HP>
```

Task 04:

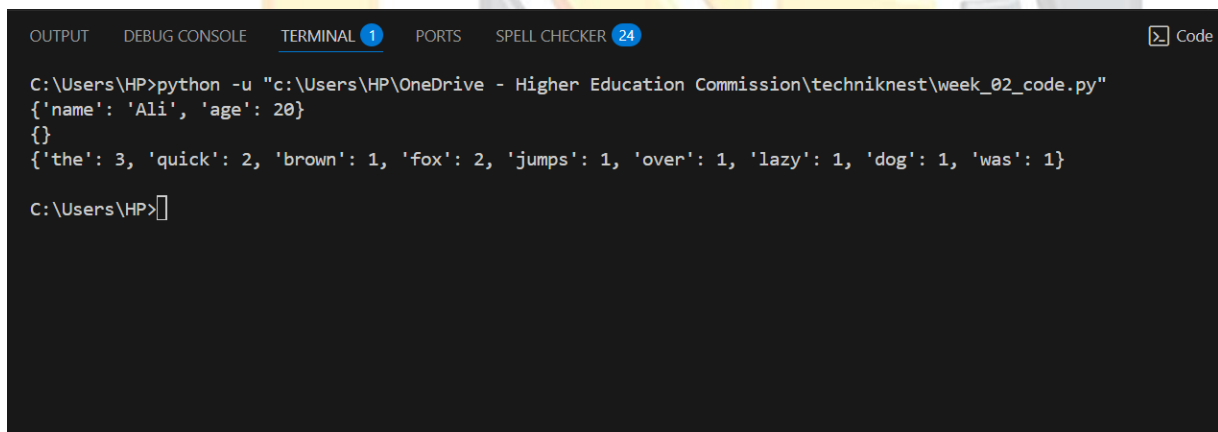
1. Student record CRUD in dict.
2. Count word frequency in sentence.

code:



```
41
42 #-----Task 4-----
43
44 # Student record CRUD in dict:
45 students = {}
46
47 # Create
48 students["001"] = {"name": "Ali", "age": 20}
49 # Read
50 print(students["001"])
51 # Update
52 students["001"]["age"] = 21
53 # Delete
54 del students["001"]
55 print(students)
56
57
58 # Count word frequency in sentence:
59 sentence = "the quick brown fox jumps over the lazy dog the fox was quick"
60 words = sentence.split()
61 freq = {}
62
63 for word in words:
64     freq[word] = freq.get(word, 0) + 1
65
66 print(freq)
67
68
```

Output:



```
OUTPUT  DEBUG CONSOLE  TERMINAL 1  PORTS  SPELL CHECKER 24  Code

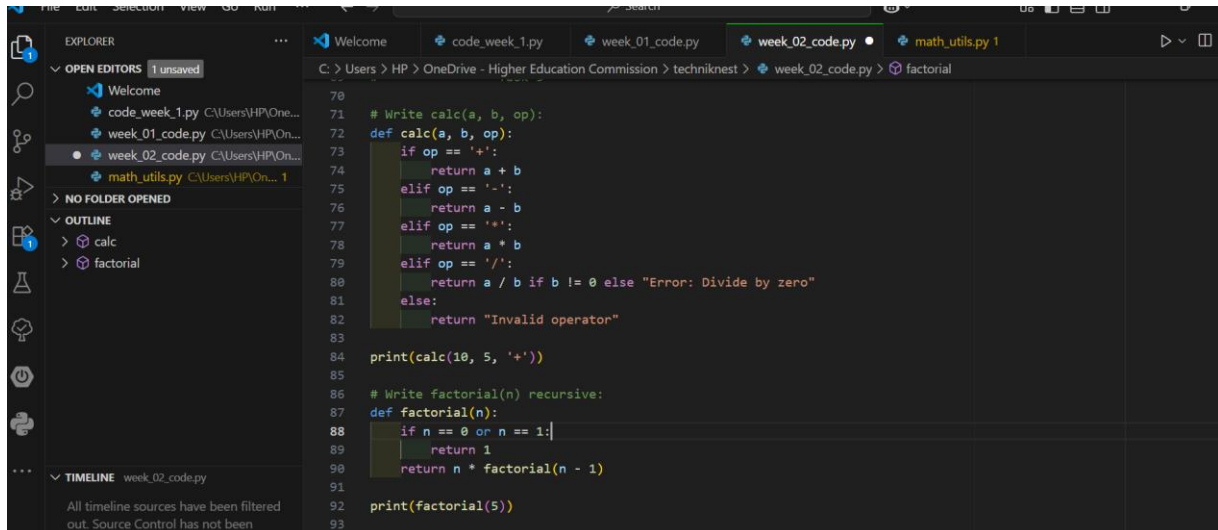
C:\Users\HP>python -u "c:\Users\HP\OneDrive - Higher Education Commission\techniknest\week_02_code.py"
{'name': 'Ali', 'age': 20}
{}
{'the': 3, 'quick': 2, 'brown': 1, 'fox': 2, 'jumps': 1, 'over': 1, 'lazy': 1, 'dog': 1, 'was': 1}

C:\Users\HP>
```

Task 05:

1. Write `calc(a,b,op)`.
2. Write `factorial(n)` recursive.

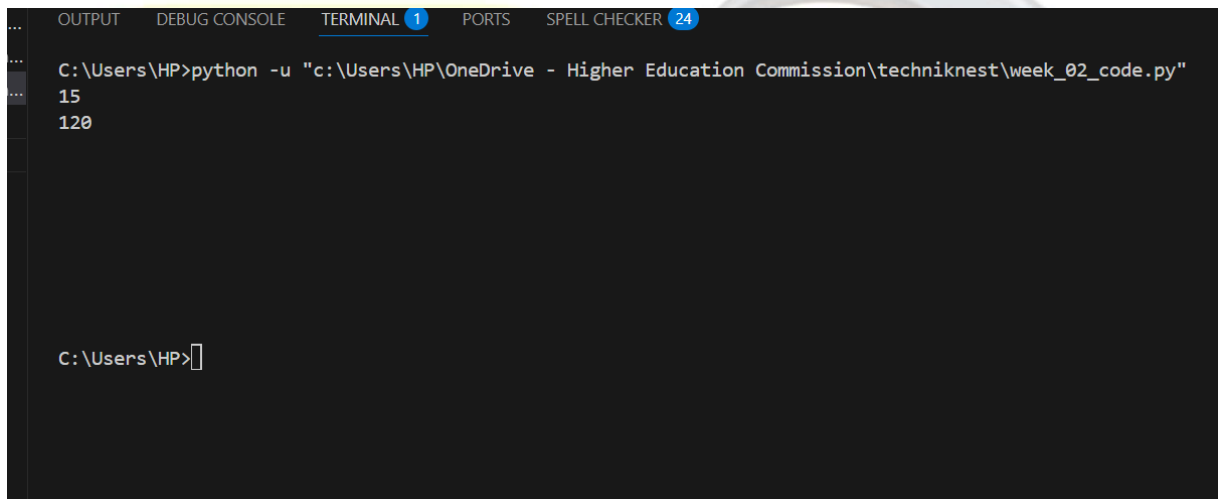
Code:



The screenshot shows a code editor with the following Python code in `week_02_code.py`:

```
70
71 # Write calc(a, b, op):
72 def calc(a, b, op):
73     if op == '+':
74         return a + b
75     elif op == '-':
76         return a - b
77     elif op == '*':
78         return a * b
79     elif op == '/':
80         return a / b if b != 0 else "Error: Divide by zero"
81     else:
82         return "Invalid operator"
83
84 print(calc(10, 5, '+'))
85
86 # Write factorial(n) recursive:
87 def factorial(n):
88     if n == 0 or n == 1:
89         return 1
90     return n * factorial(n - 1)
91
92 print(factorial(5))
93
```

Output:



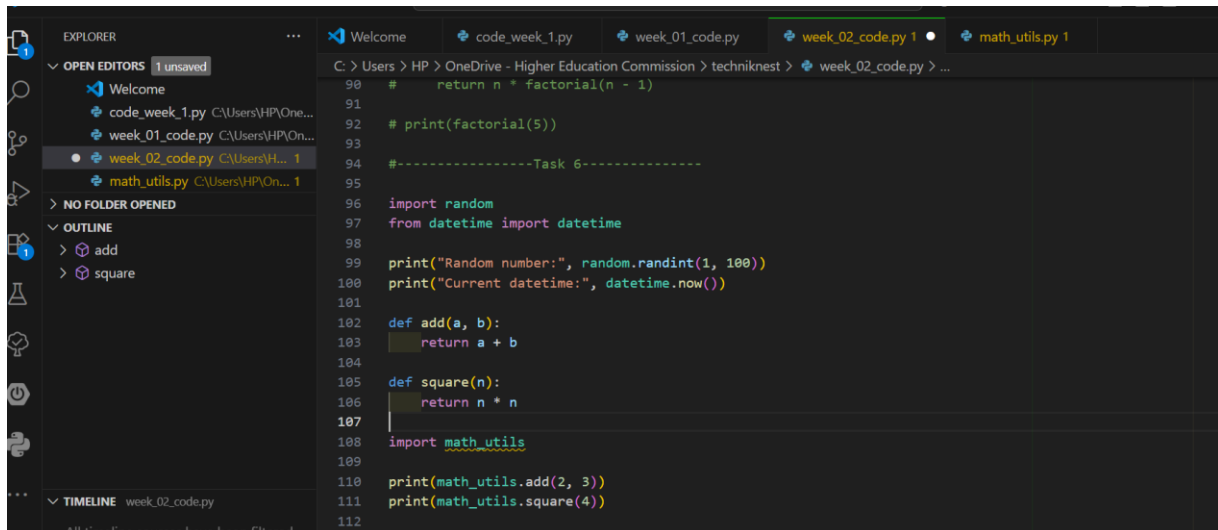
The screenshot shows a terminal window with the following output:

```
C:\Users\HP>python -u "c:\Users\HP\OneDrive - Higher Education Commission\techniknest\week_02_code.py"
15
120

C:\Users\HP>
```

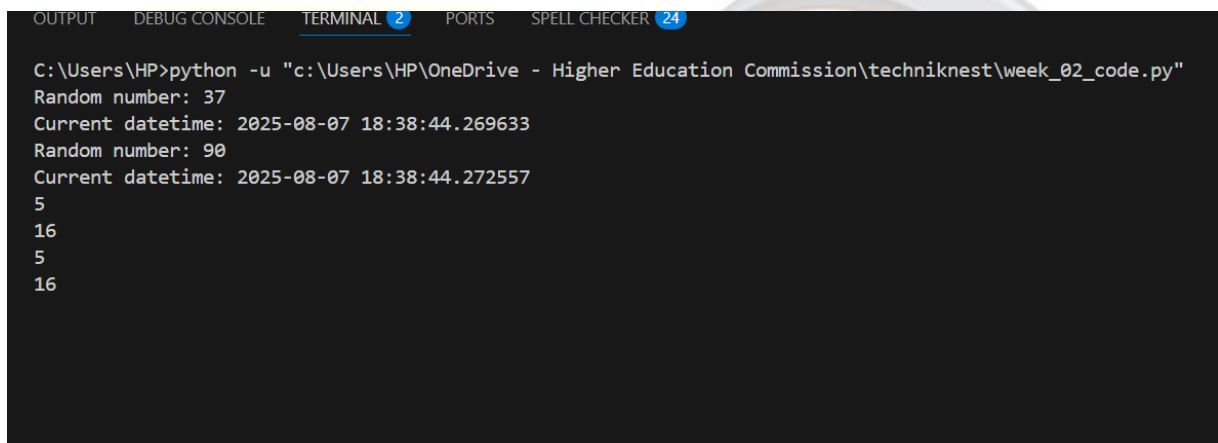
Task 06:

1. Use random & datetime in script.
 2. Create `math_utils` module & import.
- code:



```
90 # return n * factorial(n - 1)
91
92 # print(factorial(5))
93
94 #-----Task 6-----
95
96 import random
97 from datetime import datetime
98
99 print("Random number:", random.randint(1, 100))
100 print("Current datetime:", datetime.now())
101
102 def add(a, b):
103     return a + b
104
105 def square(n):
106     return n * n
107
108 import math_utils
109
110 print(math_utils.add(2, 3))
111 print(math_utils.square(4))
112
```

Output:



```
C:\Users\HP>python -u "c:\Users\HP\OneDrive - Higher Education Commission\techniknest\week_02_code.py"
Random number: 37
Current datetime: 2025-08-07 18:38:44.269633
Random number: 90
Current datetime: 2025-08-07 18:38:44.272557
5
16
5
16
```

Task 07:

1. Safe int input loop.
 2. File open with error message.
- code:

```
111 # print(math_utils.square(4))
112
113 #-----Task 7-----
114
115
116 # Safe int input loop:
117 # Keeps asking until the user provides a valid integer.
118
119 while True:
120     user_input = input("Enter an integer: ")
121     try:
122         number = int(user_input)
123         print("You entered:", number)
124         break
125     except ValueError:
126         print("Invalid input. Please enter a valid integer.")
127
128 # File open with error message:
129 # Attempts to open a file and handles the error if the file is missing.
130 filename = "data.txt"
131
132 try:
133     with open(filename, 'r') as file:
134         content = file.read()
135         print("File content:\n", content)
136 except FileNotFoundError:
137     print(f"Error: The file '{filename}' does not exist.")
138
139
```

Output:

```
C:\Users\HP>python -u "c:\Users\HP\OneDrive - Higher Education Commission\techniknest\week_02_code.py"
Enter an integer: 2
You entered: 2
Error: The file 'data.txt' does not exist.

C:\Users\HP>
```

Task 08:

Phonebook App: CRUD contacts dict <-> JSON file storage.
code:

```
C:\Users\HP\OneDrive - Higher Education Commission > techniknest > week_02_code.py > main

140 #-----Task 8-----
141
142 import json
143
144 FILENAME = "phonebook.json"
145
146 # Load contacts from file
147 def load_phonebook():
148     try:
149         with open(FILENAME, "r") as file:
150             return json.load(file)
151     except:
152         return {} # Return empty phonebook if file doesn't exist or is empty
153
154 # Save contacts to file
155 def save_phonebook(phonebook):
156     with open(FILENAME, "w") as file:
157         json.dump(phonebook, file, indent=4)
158
159 # Add a contact
160 def add_contact(phonebook):
161     name = input("Enter name: ")
162     number = input("Enter phone number: ")
163     phonebook[name] = number
164     print("Contact added.")
165
166 # View all contacts
167 def view_contacts(phonebook):
```

```
169     if not phonebook:
170         print("No contacts found.")
171     else:
172         for name, number in phonebook.items():
173             print(f"{name}: {number}")
174
175 # Update a contact
176 def update_contact(phonebook):
177     name = input("Enter name to update: ")
178     if name in phonebook:
179         number = input("Enter new phone number: ")
180         phonebook[name] = number
181         print("Contact updated.")
182     else:
183         print("Contact not found.")
184
185 # Delete a contact
186 def delete_contact(phonebook):
187     name = input("Enter name to delete: ")
188     if name in phonebook:
189         del phonebook[name]
190         print("Contact deleted.")
191     else:
192         print("Contact not found.")
193
194 # Main menu
195 def main():
196     phonebook = load_phonebook()
```



```

197
198     while True:
199         print("\nPhonebook Menu")
200         print("1. Add Contact")
201         print("2. View Contacts")
202         print("3. Update Contact")
203         print("4. Delete Contact")
204         print("5. Exit")
205
206         choice = input("Choose an option: ")
207
208         if choice == "1":
209             add_contact(phonebook)
210         elif choice == "2":
211             view_contacts(phonebook)
212         elif choice == "3":
213             update_contact(phonebook)
214         elif choice == "4":
215             delete_contact(phonebook)
216         elif choice == "5":
217             save_phonebook(phonebook)
218             print("Phonebook saved. Goodbye!")
219             break
220         else:
221             print("Invalid choice. Try again.")
222
223     # Run the app
224     if __name__ == "__main__":

```

```

225         main()
226
227

```

Output:

```

OUTPUT  DEBUG CONSOLE  TERMINAL 1  PORTS  SPELL CHECKER 24  Code + v

C:\Users\HP>python -u "c:\Users\HP\OneDrive - Higher Education Commission\techniknest\week_02_code.py"

Phonebook Menu
1. Add Contact
2. View Contacts
3. Update Contact
4. Delete Contact
5. Exit
Choose an option: 1
Enter name: Muhammad Zuhair
Enter phone number: 03495982155
Contact added.

Phonebook Menu
1. Add Contact
2. View Contacts
3. Update Contact
4. Delete Contact
5. Exit
Choose an option: 1
Enter name: Ali
Enter phone number: 03335781310
Contact added.

```

```
Phonebook Menu
1. Add Contact
2. View Contacts
3. Update Contact
4. Delete Contact
5. Exit
Choose an option: 4
Enter name to delete: Ali
Contact deleted.
```

```
Phonebook Menu
1. Add Contact
2. View Contacts
3. Update Contact
4. Delete Contact
5. Exit
Choose an option: 5
Phonebook saved. Goodbye!
```

```
C:\Users\HP>
```

