OBJECT ORIENTED PROGRAMMING

LAB TASK 3

NAME: HASSAN KHAN

ROLL NO:F24_605

SEMESTER: 2ND

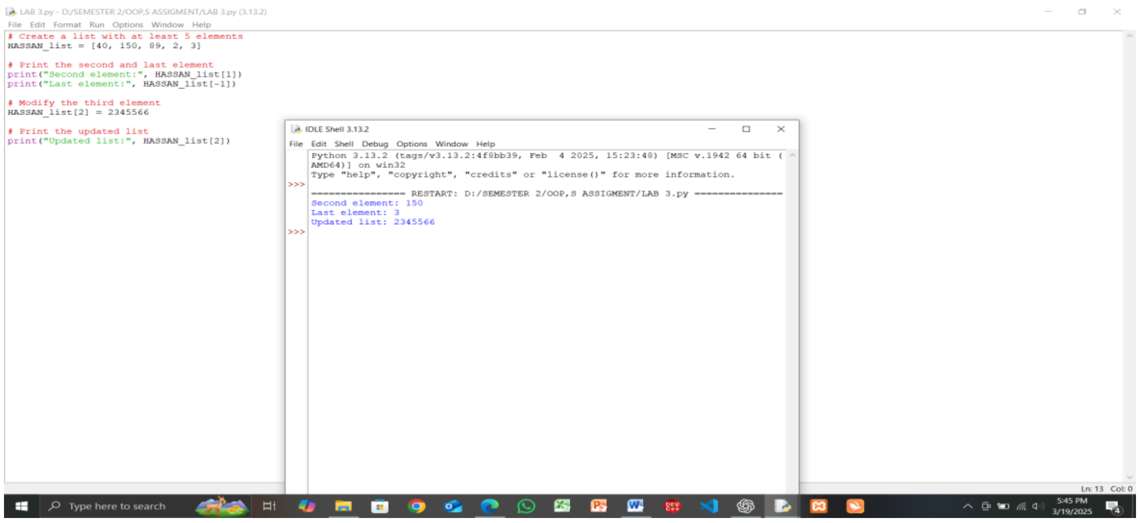
SECTION: C

SUBJECT: Object Oriented Programming

SUBMITTED TO: MR Jamal Abdul Ahad

DEPARTMENT OF COMPUTER SCIENCE ABBOTTABAD UNIVERSITY OF SCIENCE AND TECHNOLOGY

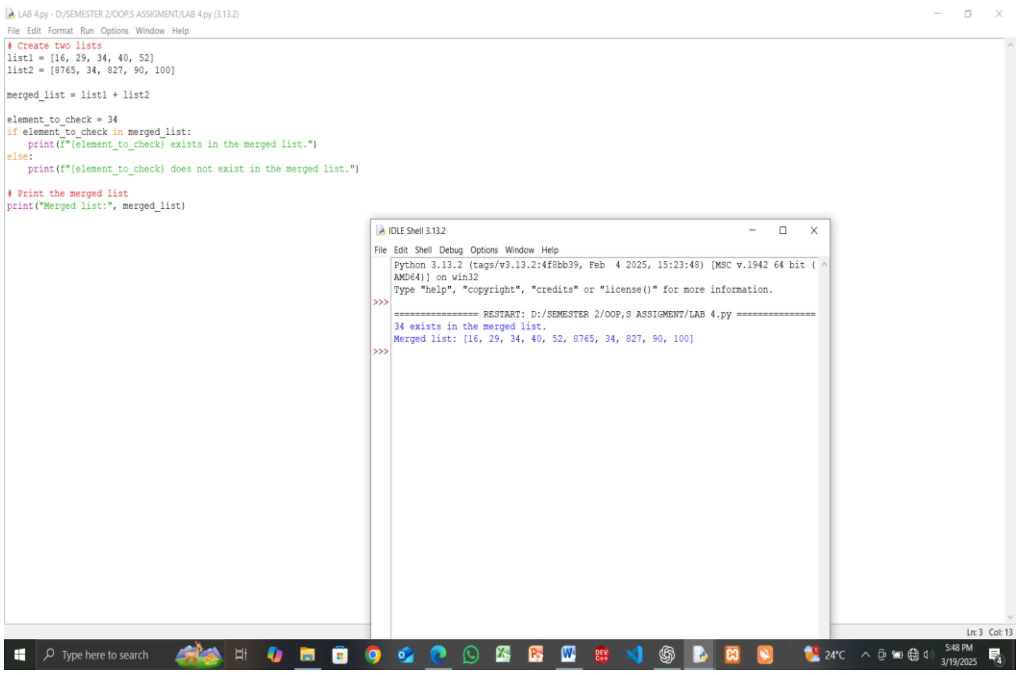
CREATING AND ACCESSING AND UPDATING LIST


LAB 3.py - D:/SEMESTER 2/OOP,S ASSIGMENT/LAB 3.py (3.13.2)
File Edit Format Run Options Window Help
IDLE Shell 3.13.2
File Edit Shell Debug Options Window Help
Ln: 13 Col: 0
Type here to search
5:45 PM
3/19/2025

```python
# Create a list with at least 5 elements
HASSAN_list = [40, 150, 89, 2, 3]

# Print the second and last element
print("Second element:", HASSAN_list[1])
print("Last element:", HASSAN_list[-1])

# Modify the third element
HASSAN_list[2] = 2345566

# Print the updated list
print("Updated list:", HASSAN_list[2])
```

```
Python 3.13.2 (tags/v3.13.2:4f8bb39, Feb  4 2025, 15:23:48) [MSC v.1942 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
================ RESTART: D:/SEMESTER 2/OOP,S ASSIGMENT/LAB 3.py ===============
Second element: 150
Last element: 3
Updated list: 2345566
>>>
```

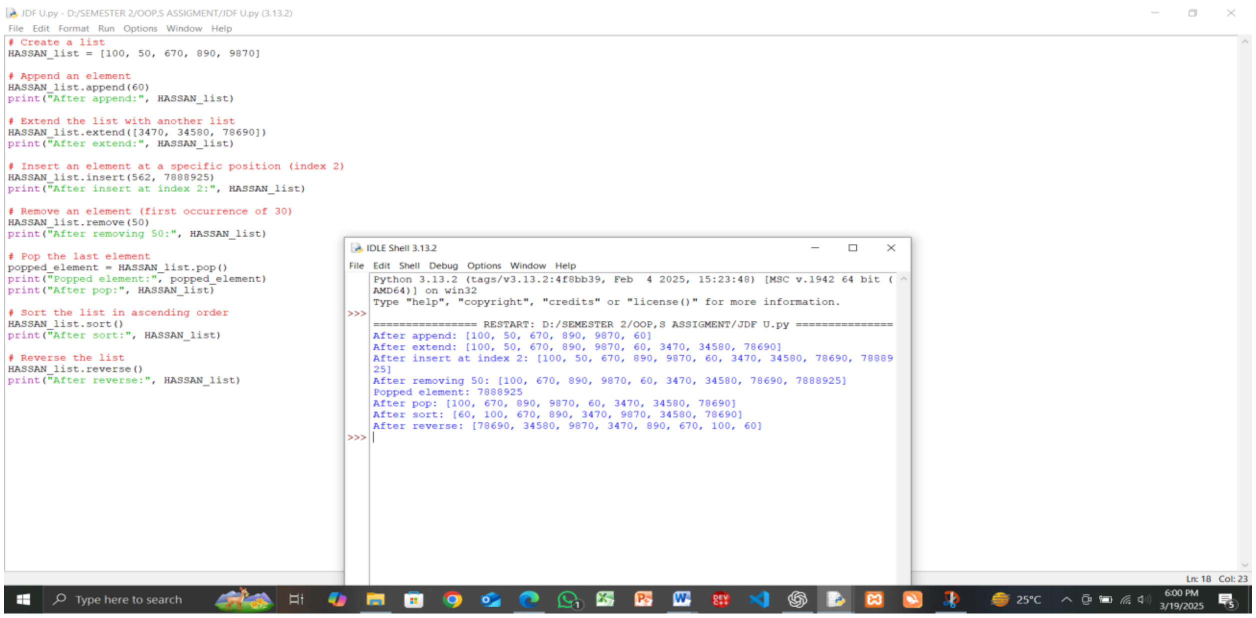
LIST OPERATIONS


LAB 4.py - D:/SEMESTER 2/OOP,S ASSIGMENT/LAB 4.py (3.13.2)
File Edit Format Run Options Window Help
IDLE Shell 3.13.2
File Edit Shell Debug Options Window Help
Ln: 3 Col: 13
Type here to search
24°C
5:48 PM
3/19/2025

```python
# Create two lists
list1 = [16, 29, 34, 40, 52]
list2 = [8765, 34, 827, 90, 100]

merged_list = list1 + list2

element_to_check = 34
if element_to_check in merged_list:
    print(f"{element_to_check} exists in the merged list.")
else:
    print(f"{element_to_check} does not exist in the merged list.")

# Print the merged list
print("Merged list:", merged_list)
```

```
Python 3.13.2 (tags/v3.13.2:4f8bb39, Feb  4 2025, 15:23:48) [MSC v.1942 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
================ RESTART: D:/SEMESTER 2/OOP,S ASSIGMENT/LAB 4.py ===============
34 exists in the merged list.
Merged list: [16, 29, 34, 40, 52, 8765, 34, 827, 90, 100]
>>>
```

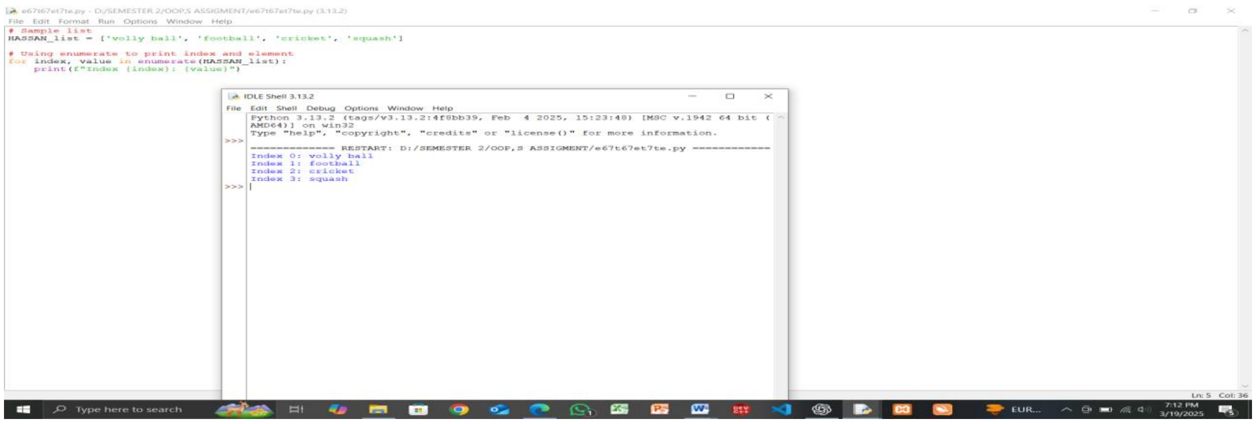
COMMAN LIST METHOD



```
# Create a list
HASSAN_list = [100, 50, 670, 890, 9870]

# Append an element
HASSAN_list.append(60)
print("After append:", HASSAN_list)

# Extend the list with another list
HASSAN_list.extend([3470, 34580, 78690])
print("After extend:", HASSAN_list)

# Insert an element at a specific position (index 2)
HASSAN_list.insert(562, 7888925)
print("After insert at index 2:", HASSAN_list)

# Remove an element (first occurrence of 30)
HASSAN_list.remove(50)
print("After removing 50:", HASSAN_list)

# Pop the last element
popped_element = HASSAN_list.pop()
print("Popped element:", popped_element)
print("After pop:", HASSAN_list)

# Sort the list in ascending order
HASSAN_list.sort()
print("After sort:", HASSAN_list)

# Reverse the list
HASSAN_list.reverse()
print("After reverse:", HASSAN_list)
```

```
Python 3.13.2 (tags/v3.13.2:4f8bb39, Feb  4 2025, 15:23:48) [MSC v.1942 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
================ RESTART: D:/SEMESTER 2/OOP,S ASSIGMENT/JDF U.py ===============
After append: [100, 50, 670, 890, 9870, 60]
After extend: [100, 50, 670, 890, 9870, 60, 3470, 34580, 78690]
After insert at index 2: [100, 50, 670, 890, 9870, 60, 3470, 34580, 78690, 78889
25]
After removing 50: [100, 670, 890, 9870, 60, 3470, 34580, 78690, 7888925]
Popped element: 7888925
After pop: [100, 670, 890, 9870, 60, 3470, 34580, 78690]
After sort: [60, 100, 670, 890, 3470, 9870, 34580, 78690]
After reverse: [78690, 34580, 9870, 3470, 890, 670, 100, 60]
>>>
```

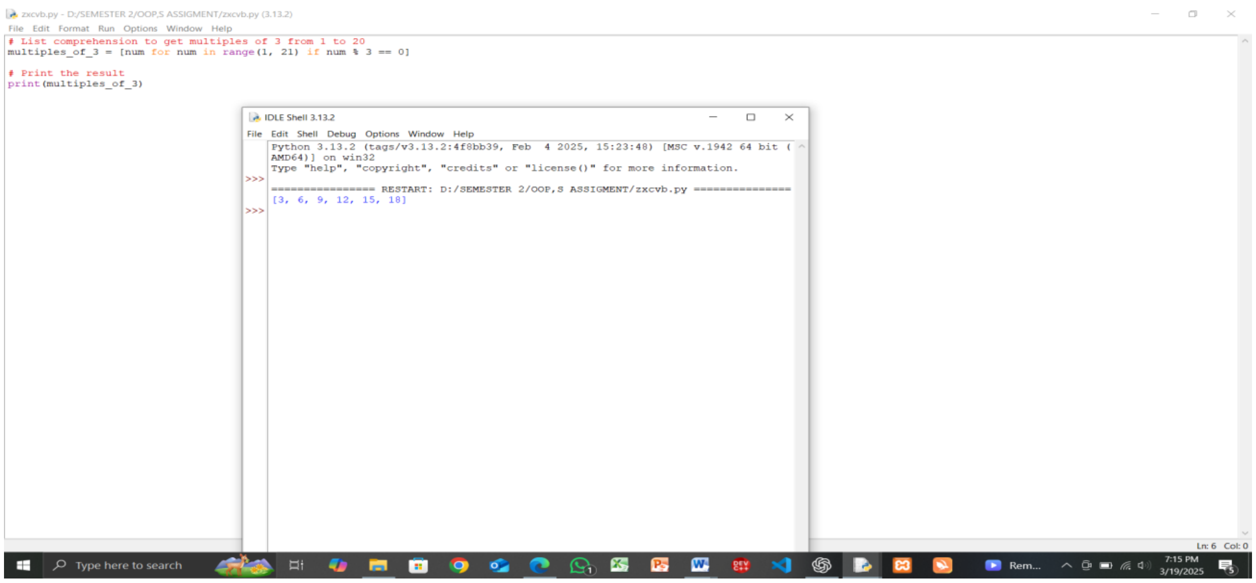
Iterating over list



```
# Sample list
HASSAN_list = ['volly ball', 'football', 'cricket', 'squash']

# Using enumerate to print index and element
for index, value in enumerate(HASSAN_list):
    print(f"Index {index}: {value}")
```

```
Python 3.13.2 (tags/v3.13.2:4f8bb39, Feb  4 2025, 15:23:48) [MSC v.1942 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
============= RESTART: D:/SEMESTER 2/OOP,S ASSIGMENT/e67t67et7te.py ============
Index 0: volly ball
Index 1: football
Index 2: cricket
Index 3: squash
>>>
```

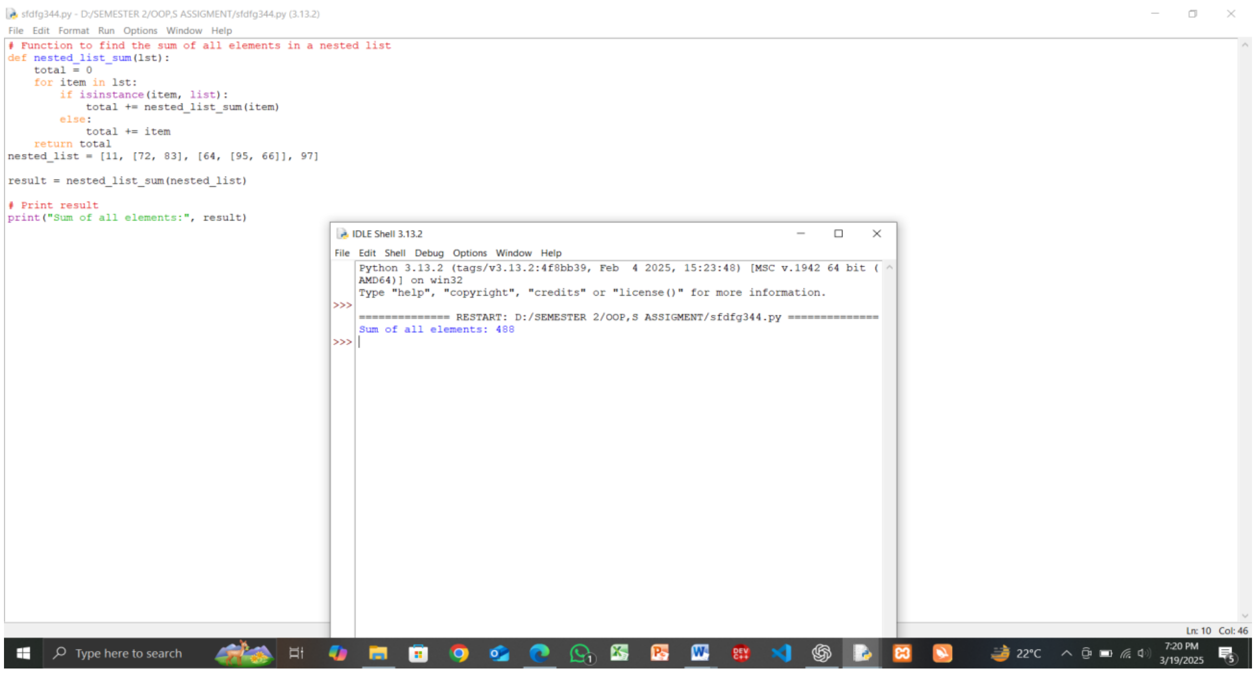
LIST COMPREHENSIONS



```
# List comprehension to get multiples of 3 from 1 to 20
multiples_of_3 = [num for num in range(1, 21) if num % 3 == 0]

# Print the result
print(multiples_of_3)
```

```
Python 3.13.2 (tags/v3.13.2:4f8bb39, Feb  4 2025, 15:23:48) [MSC v.1942 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
================ RESTART: D:/SEMESTER 2/OOP,S ASSIGMENT/zxcvb.py ===============
[3, 6, 9, 12, 15, 18]
>>>
```

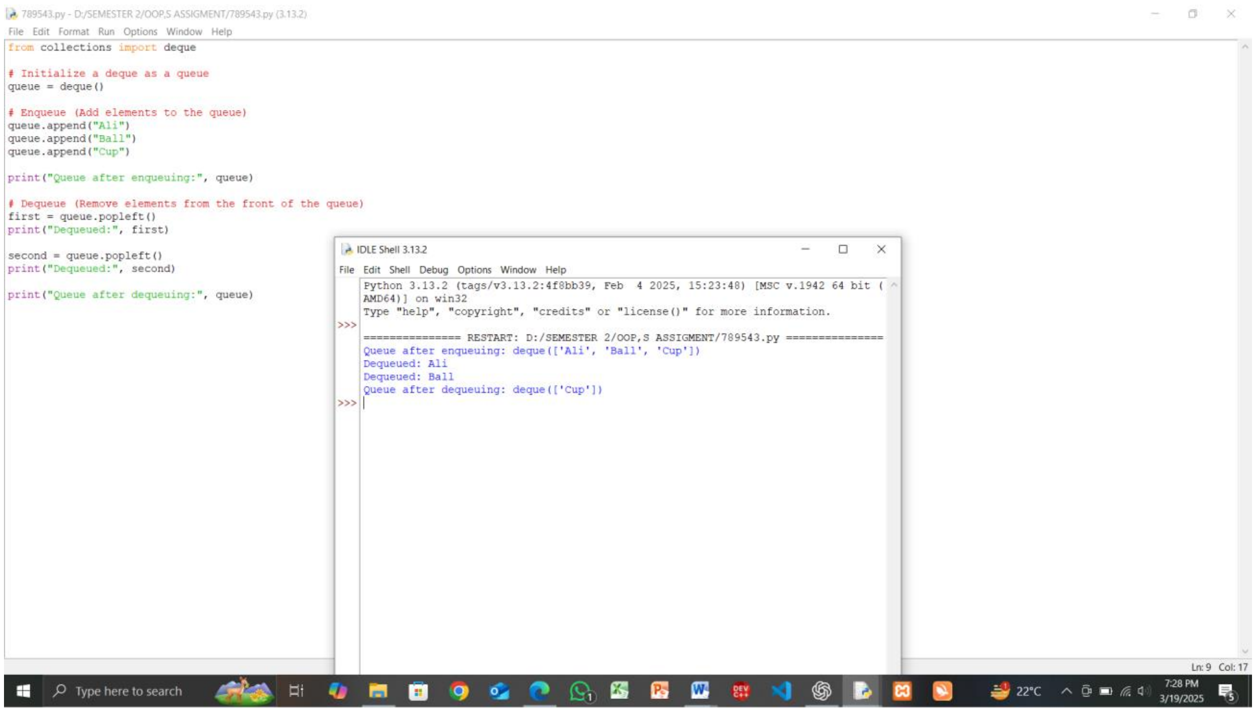
WORKING WITH NESTED LISTS



```
# Function to find the sum of all elements in a nested list
def nested_list_sum(lst):
    total = 0
    for item in lst:
        if isinstance(item, list):
            total += nested_list_sum(item)
        else:
            total += item
    return total
nested_list = [11, [72, 83], [64, [95, 66]], 97]

result = nested_list_sum(nested_list)

# Print result
print("Sum of all elements:", result)
```

```
Python 3.13.2 (tags/v3.13.2:4f8bb39, Feb  4 2025, 15:23:48) [MSC v.1942 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
============== RESTART: D:/SEMESTER 2/OOP,S ASSIGMENT/sfdfg344.py ==============
Sum of all elements: 488
>>>
```

LIST PERFORMANCE AND OPTIMAZATION



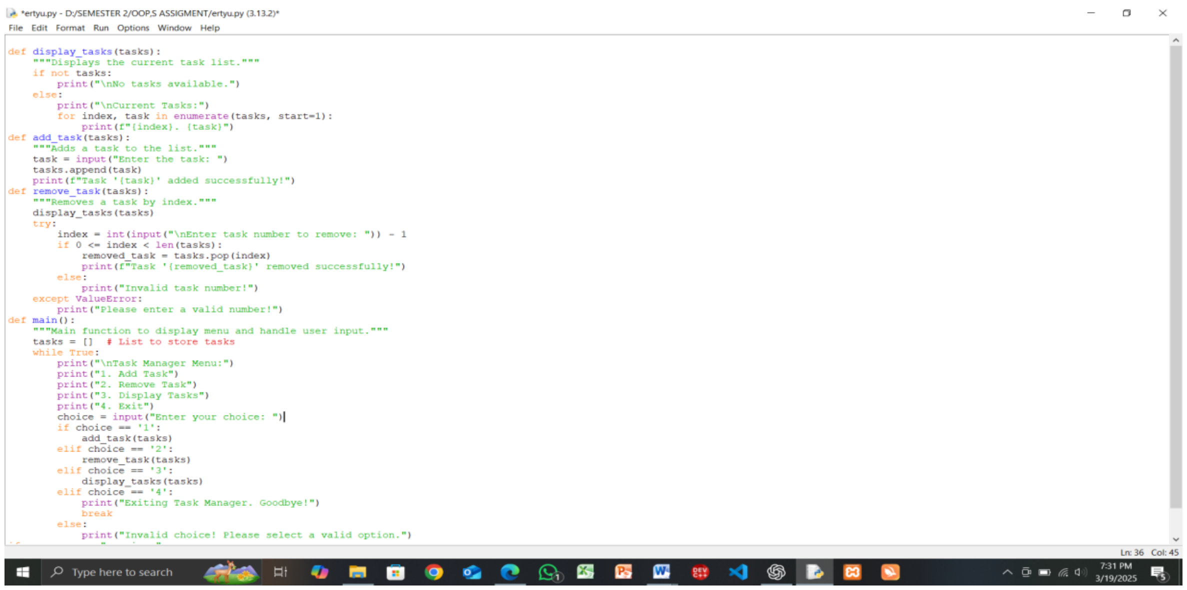
789543.py - D:/SEMESTER 2/OOP,S ASSIGMENT/789543.py (3.13.2)

```python
from collections import deque

# Initialize a deque as a queue
queue = deque()

# Enqueue (Add elements to the queue)
queue.append("Ali")
queue.append("Ball")
queue.append("Cup")

print("Queue after enqueuing:", queue)

# Dequeue (Remove elements from the front of the queue)
first = queue.popleft()
print("Dequeued:", first)

second = queue.popleft()
print("Dequeued:", second)

print("Queue after dequeuing:", queue)
```

IDLE Shell 3.13.2

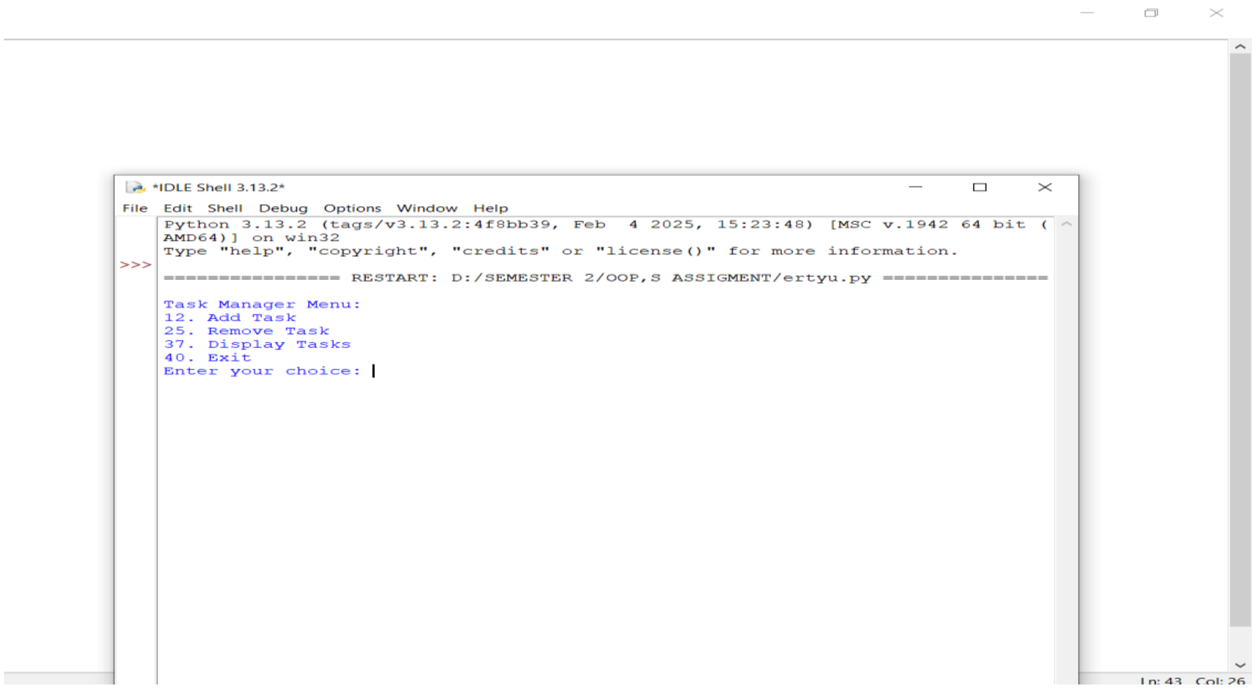
```
Python 3.13.2 (tags/v3.13.2:4f8bb39, Feb  4 2025, 15:23:48) [MSC v.1942 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
=============== RESTART: D:/SEMESTER 2/OOP,S ASSIGMENT/789543.py ===============
Queue after enqueuing: deque(['Ali', 'Ball', 'Cup'])
Dequeued: Ali
Dequeued: Ball
Queue after dequeuing: deque(['Cup'])
>>>
```

REAL WORLD APPLICATION OF LIST



ertyu.py - D:/SEMESTER 2/OOP,S ASSIGMENT/ertyu.py (3.13.2)

```python
def display_tasks(tasks):
    """Displays the current task list."""
    if not tasks:
        print("\nNo tasks available.")
    else:
        print("\nCurrent Tasks:")
        for index, task in enumerate(tasks, start=1):
            print(f"{index}. {task}")
def add_task(tasks):
    """Adds a task to the list."""
    task = input("Enter the task: ")
    tasks.append(task)
    print(f"Task '{task}' added successfully!")
def remove_task(tasks):
    """Removes a task by index."""
    display_tasks(tasks)
    try:
        index = int(input("\nEnter task number to remove: ")) - 1
        if 0 <= index < len(tasks):
            removed_task = tasks.pop(index)
            print(f"Task '{removed_task}' removed successfully!")
        else:
            print("Invalid task number!")
    except ValueError:
        print("Please enter a valid number!")
def main():
    """Main function to display menu and handle user input."""
    tasks = []  # List to store tasks
    while True:
        print("\nTask Manager Menu:")
        print("1. Add Task")
        print("2. Remove Task")
        print("3. Display Tasks")
        print("4. Exit")
        choice = input("Enter your choice: ")
        if choice == '1':
            add_task(tasks)
        elif choice == '2':
            remove_task(tasks)
        elif choice == '3':
            display_tasks(tasks)
        elif choice == '4':
            print("Exiting Task Manager. Goodbye!")
            break
        else:
            print("Invalid choice! Please select a valid option.")
```



```
Python 3.13.2 (tags/v3.13.2:4f8bb39, Feb  4 2025, 15:23:48) [MSC v.1942 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
================ RESTART: D:/SEMESTER 2/OOP,S ASSIGMENT/ertyu.py ===============

Task Manager Menu:
12. Add Task
25. Remove Task
37. Display Tasks
40. Exit
Enter your choice: 
```