def list\_operations():

# Initialize an empty list

my\_list = []

print("Initial list:", my\_list)

# 1. Append: Add an element to the end of the list

element = input("Enter an element to append to the list: ")

my\_list.append(element)

print("List after append:", my\_list)

# 2. Extend: Add multiple elements to the end of the list

elements = input("Enter multiple elements (comma-separated) to extend the list: ").split(',')

my\_list.extend(elements)

print("List after extend:", my\_list)

# 3. Insert: Add an element at a specific position

position = int(input("Enter the position to insert an element: "))

element = input("Enter the element to insert: ")

my\_list.insert(position, element)

print("List after insert:", my\_list)

# 4. Remove: Remove the first occurrence of a specific element

element = input("Enter the element to remove: ")

if element in my\_list:

my\_list.remove(element)

print("List after remove:", my\_list)

else:

print(f"Element '{element}' not found in the list.")

# 5. Pop: Remove and return the element at a specific position

position = int(input("Enter the position to pop an element: "))

if position < len(my\_list):

popped\_element = my\_list.pop(position)

print(f"Popped element: {popped\_element}")

print("List after pop:", my\_list)

else:

print("Invalid position.")

# 6. Clear: Remove all elements from the list

my\_list.clear()

print("List after clear:", my\_list)

# Re-initialize the list for further operations

my\_list = [1, 2, 2, 3, 4, 5, 2, 6, 7, 8, 2, 9, 10]

print("Re-initialized list:", my\_list)

# 7. Index: Find the first occurrence of a specific element

element = int(input("Enter the element to find the index of: "))

if element in my\_list:

index = my\_list.index(element)

print(f"First occurrence of element '{element}' is at index: {index}")

else:

print(f"Element '{element}' not found in the list.")

# 8. Count: Count the occurrences of a specific element

element = int(input("Enter the element to count: "))

count = my\_list.count(element)

print(f"Element '{element}' occurs {count} time(s) in the list.")

# 9. Sort: Sort the list in ascending order

my\_list.sort()

print("List after sort:", my\_list)

# 10. Reverse: Reverse the elements of the list

my\_list.reverse()

print("List after reverse:", my\_list)

# Run the list operations function

list\_operations()