

# NORTH WESTERN UNIVERSITY



---

**Course Title: Data Structures Sessional**

**Course Code: CSE-2104**

Special Thanks to:

Md. Shymon Islam

Lecturer, Department Of CSE

North Western University

Khulna, Bangladesh.

Developed by:

Farah Ulfat Reefa

ID: 20221170010

Hridoy Kumar Bala

ID: 20221177010

Ayan Tasmim Shaity

ID: 20221132010

Department Of CSE

North Western University

Khulna, Bangladesh.

## Table of Contents

1) Introduction.....	1
2) Objectives.....	1
3) Description.....	2
▪ <i>Homepage</i> .....	(...)
▪ <i>Secondary Window</i> .....	(...)
▪ <i>Operation List</i> .....	(...)
▪ <i>User Input</i> .....	(...)
▪ <i>Mouse Action</i> .....	(...)
▪ <i>Output Display</i> .....	(...)
4) Dependencies.....	20

## **Introduction:**

This is a GUI project that aims to bridge the gap between the complexities of working with arrays and linked lists and the ease of use provided by a graphical interface. By utilizing intuitive visual components and interactive features, we have designed an application that simplifies the process of performing common operations on arrays and linked lists.

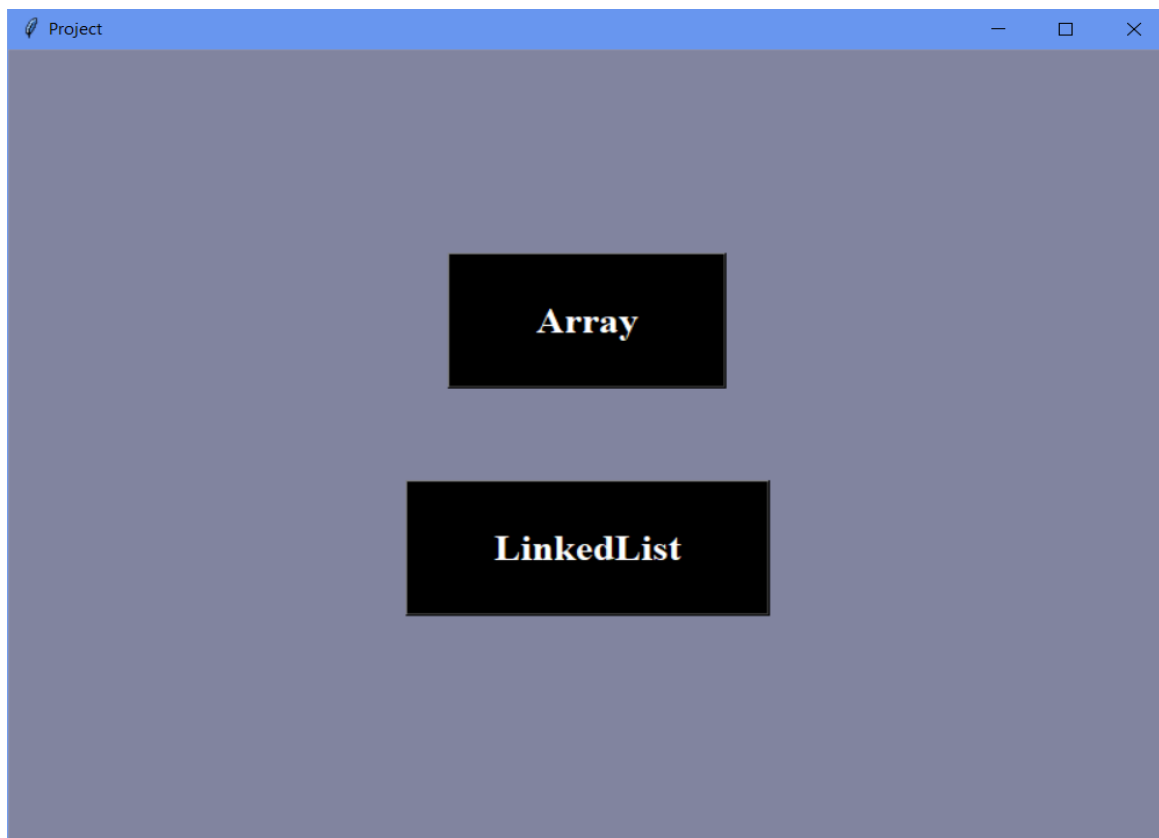
Arrays and linked lists are fundamental data structures used in computer science and programming. Arrays offer a contiguous block of memory for storing elements of the same type, providing efficient access to elements based on their indices. On the other hand, a linked list is a dynamic data structure composed of nodes, where each node contains a value and a reference (or pointer) to the next node in the sequence, allowing dynamic memory allocation and efficient insertion and deletion operations.

## **Objectives:**

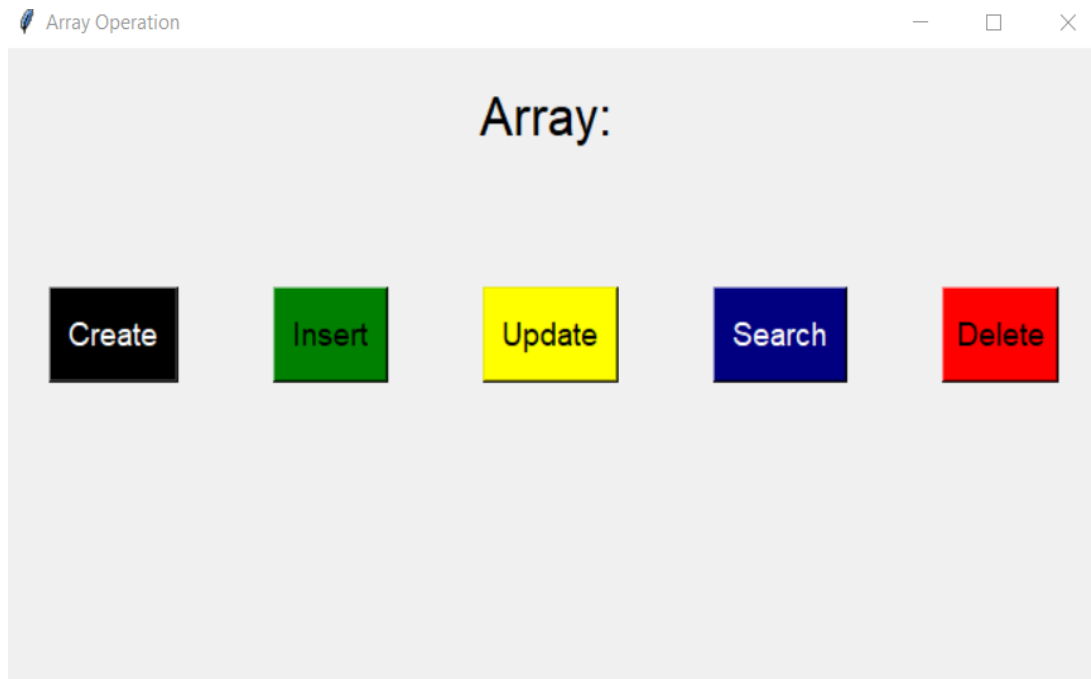
The primary objective of this project is to create a Graphical User Interface (GUI) application that allows users to interact with and perform operations on two different data structures: Arrays and Linked lists. The program utilizes the Tkinter library in python to create the GUI. The program aims to provide a user-friendly interface for users to create, modify, and visualize arrays and linked lists. It demonstrates basic functionalities of these data structures and their respective operations.

## Description:

- Homepage: The main window of the application displays two buttons: "Array" and "Linked List". When the "Array" button is clicked, it opens a secondary window dedicated to array operations. Similarly, when the "Linked List" button is clicked, it opens a window for linked list operations.

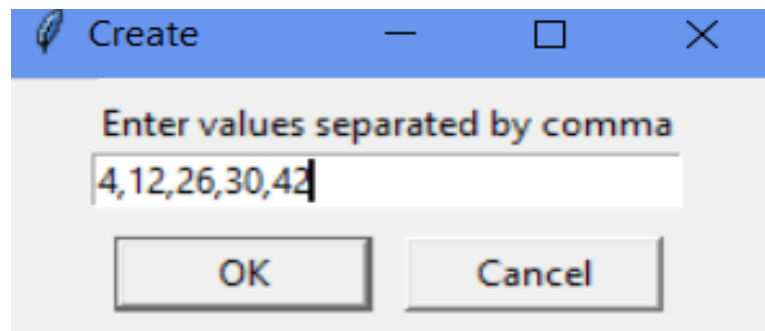


- Array Window: The secondary window for Array contains buttons: “Create”, “Insert”, “Delete”, “Update”, “Search.”

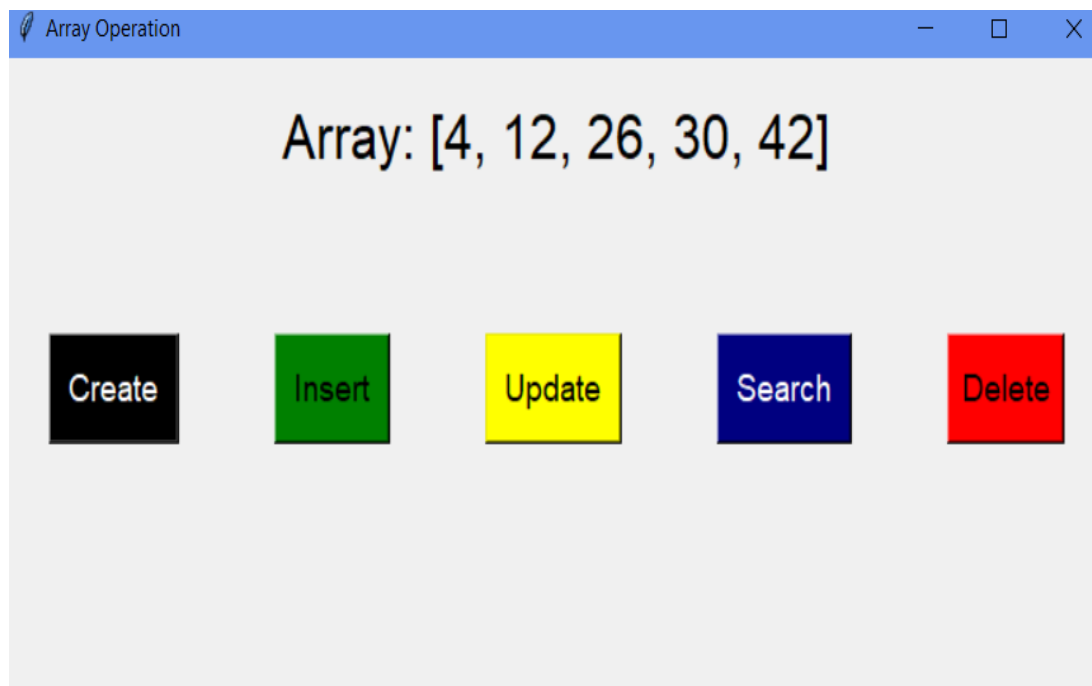


- Array Operation List: Creation, Insertion, Search, Update, Deletion.
- Create: Opens a dialog box where the user can enter values separated by commas to create an array list.
  - Insert: Allows the user to choose an insert option (by index, at the end, or at the beginning) and performs the corresponding insertion operation on the array.
  - Update: Allows the user to update a value in the array by specifying the index and providing a new value.
  - Search : Allows the user to enter a value to search for in the array and displays a message box indicating whether the value was found or not.
  - Delete: Allows the user to choose a delete option (by index, at the end, or at the beginning) and performs the corresponding deletion operation on the array.

- User Input: Opens a dialog box where the user can enter values.

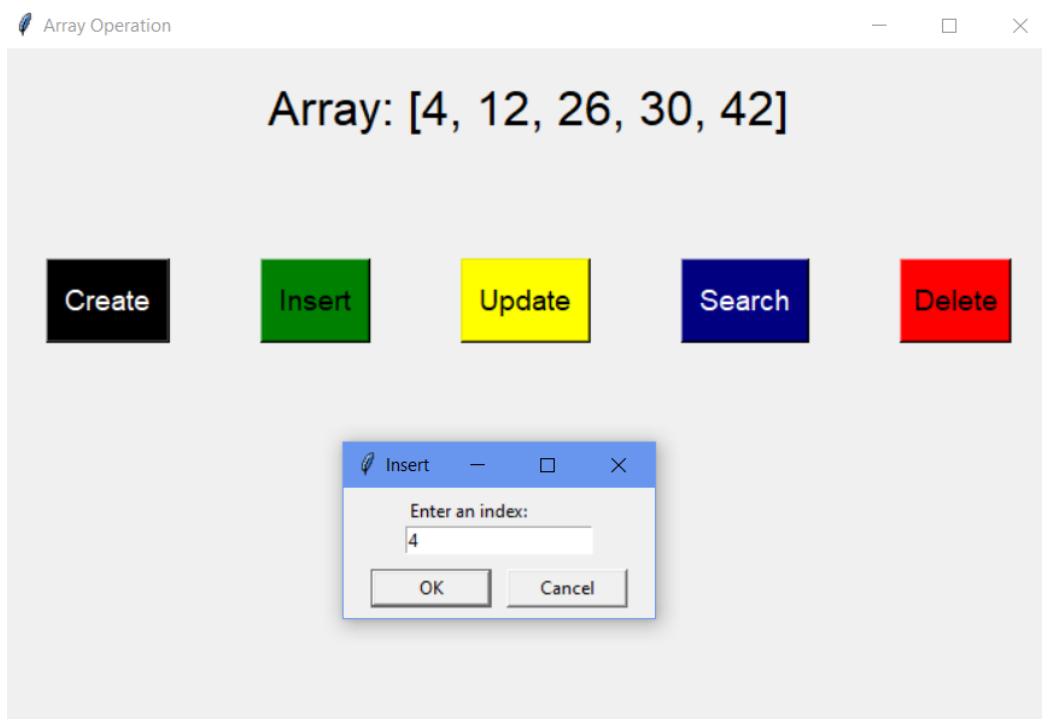
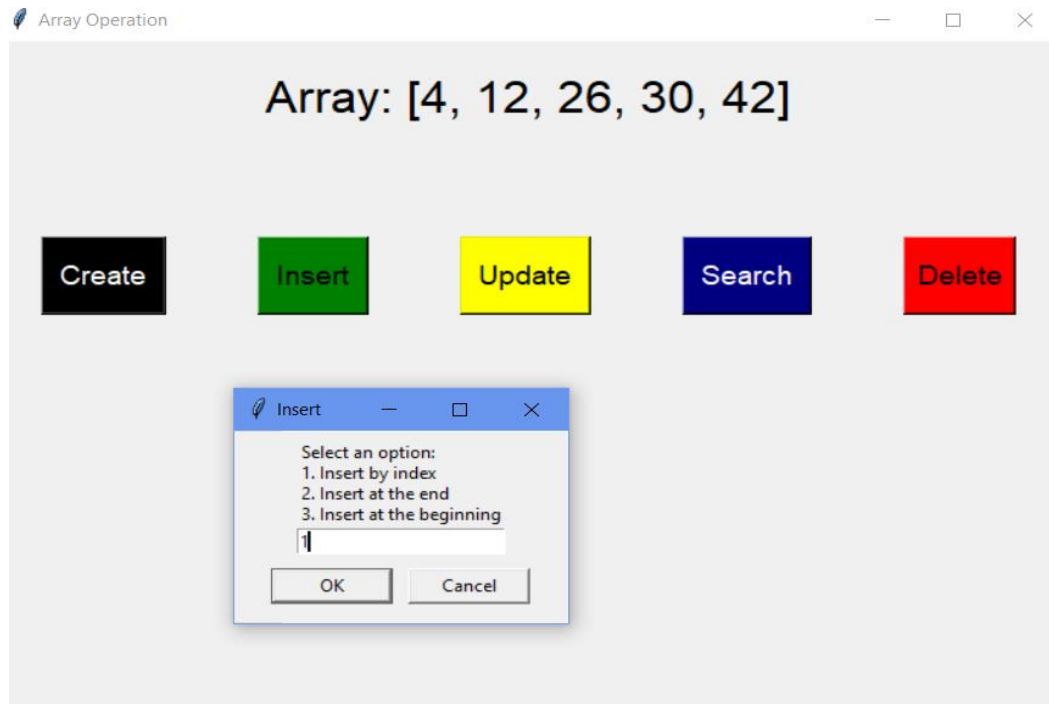


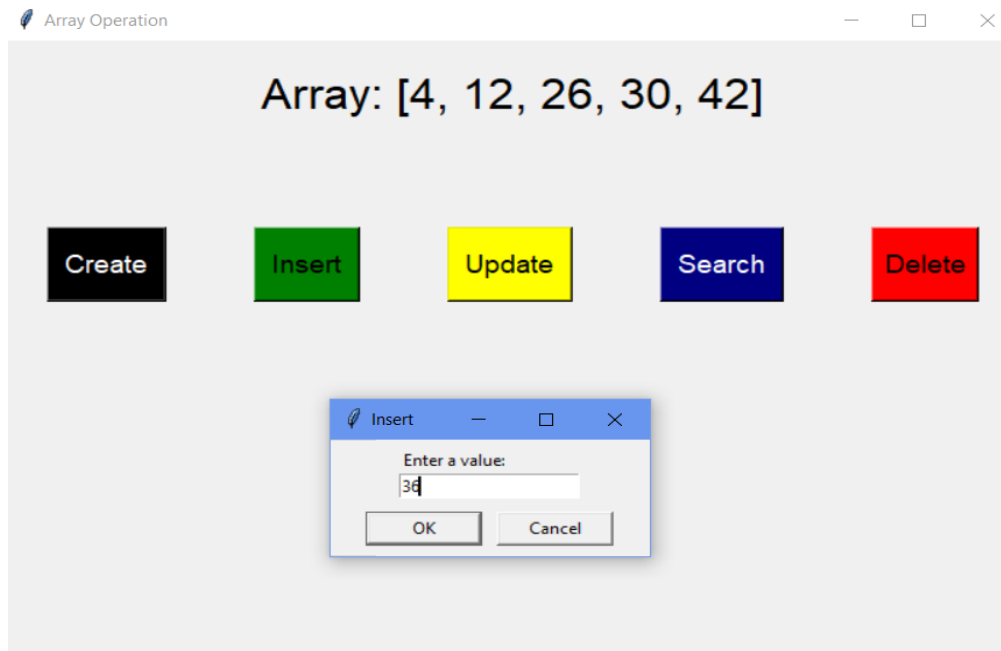
- Array Output Display: The created array list is shown in the display window.



➤ Array Insertion:

After inserting a value 36 at index 4.

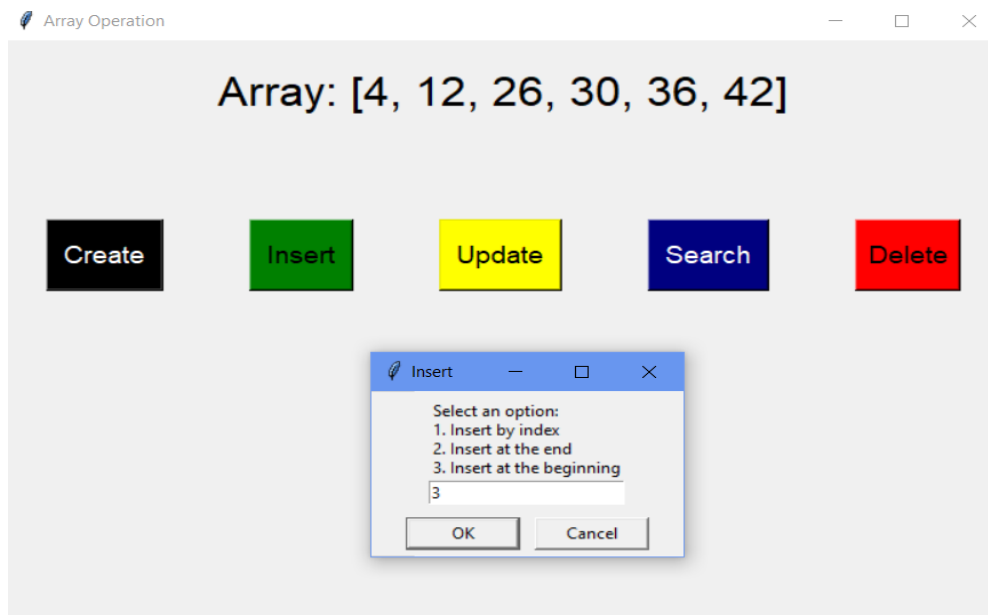




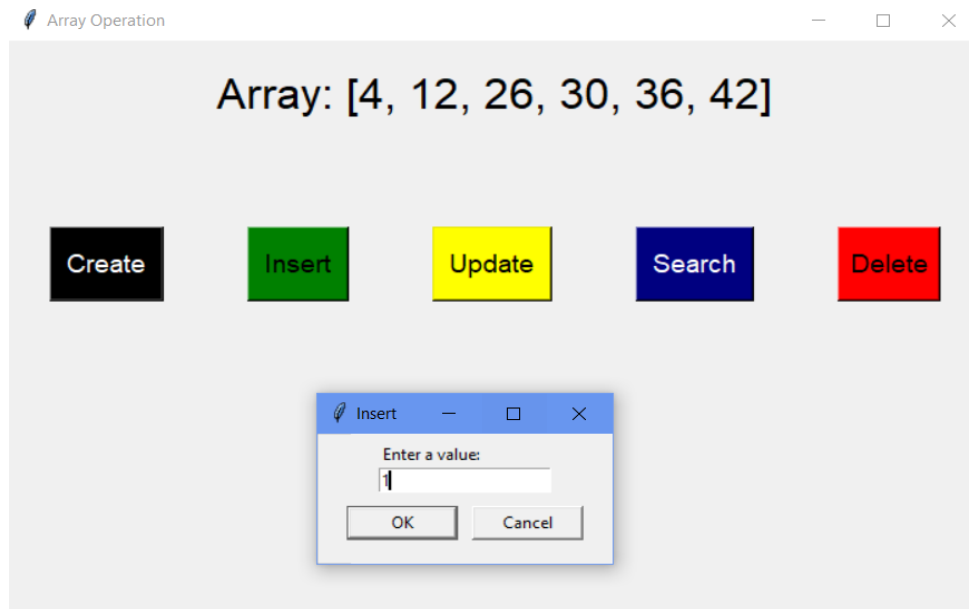
The results after insertion at index :

Array: [4, 12, 26, 30, 36, 42]

After inserting value 1 as the first element of the list.



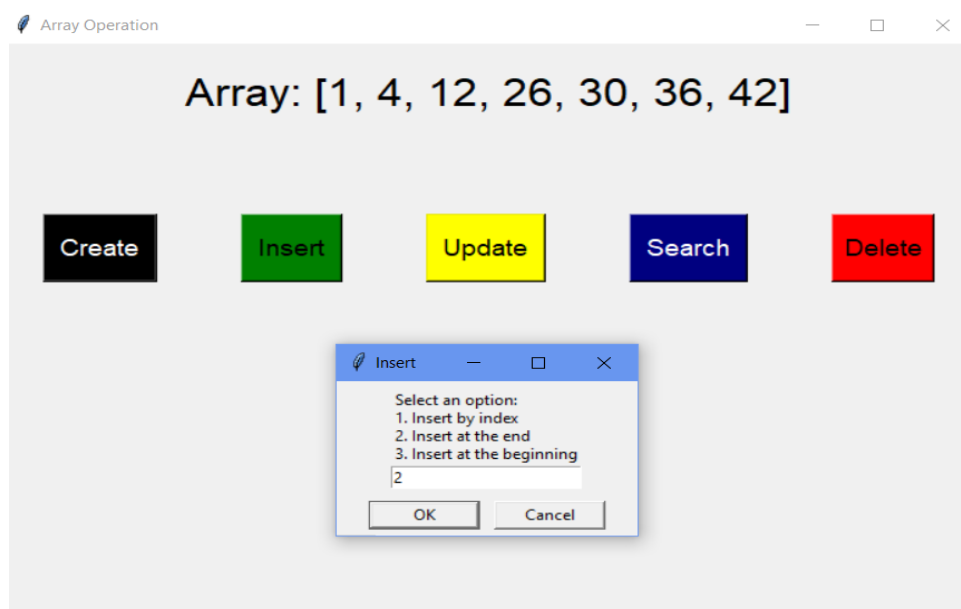


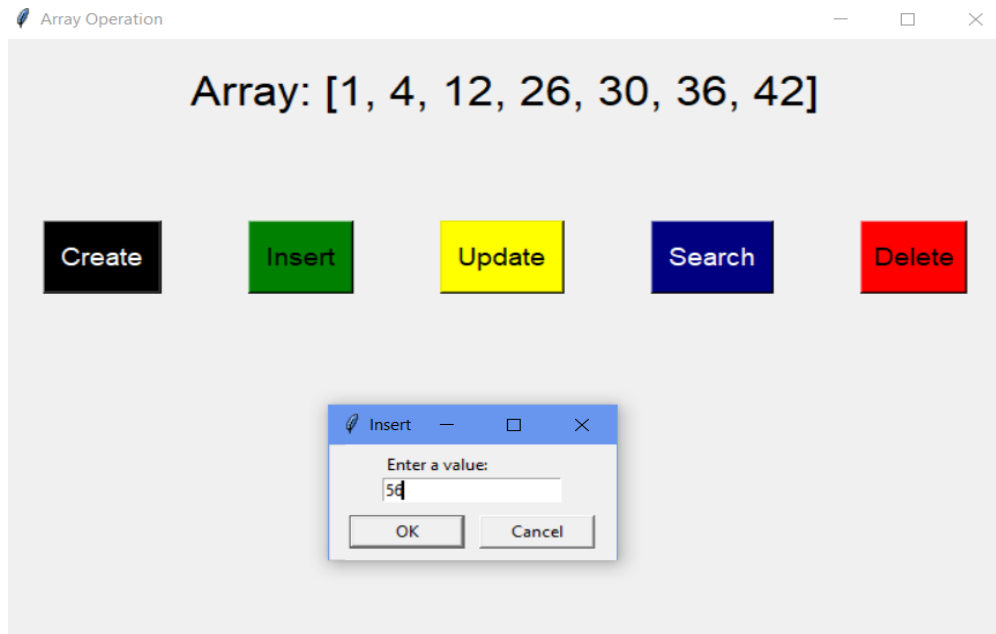


The results after inserting a value at the beginning of the list :

**Array: [1, 4, 12, 26, 30, 36, 42]**

After inserting value 56 as the last element of the list.

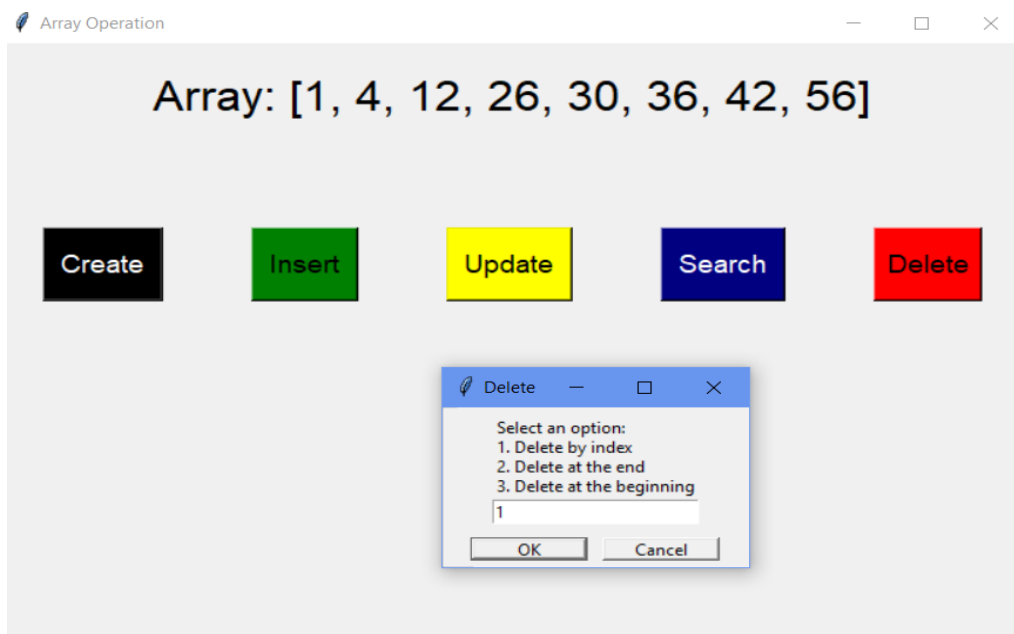


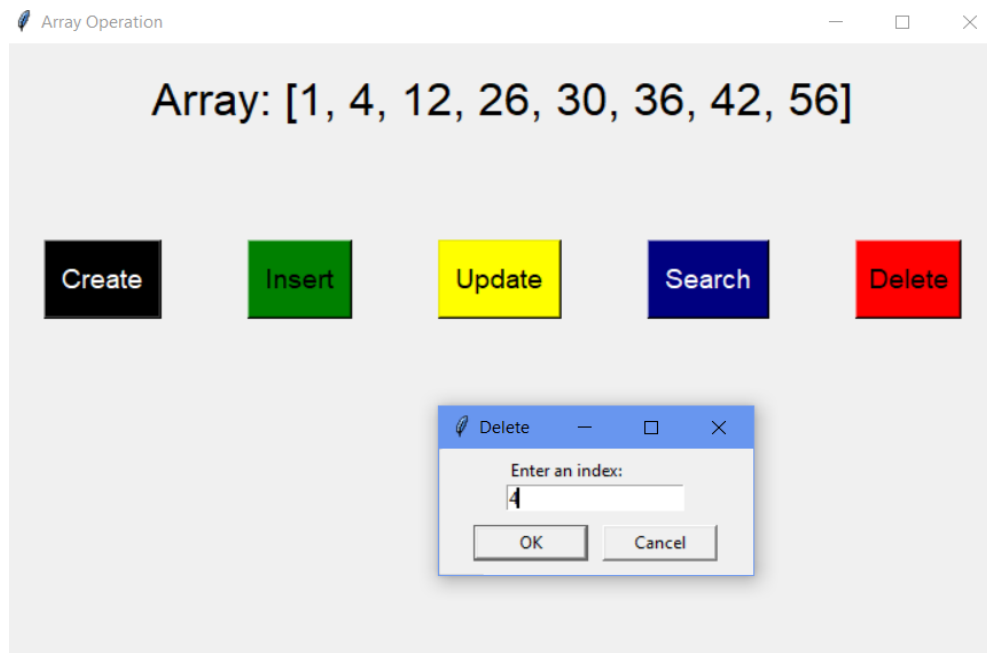


The results after inserting a value at the end of the list :

Array: [1, 4, 12, 26, 30, 36, 42, 56]

➤ Array Deletion: After deleting a value 36 at index 4.

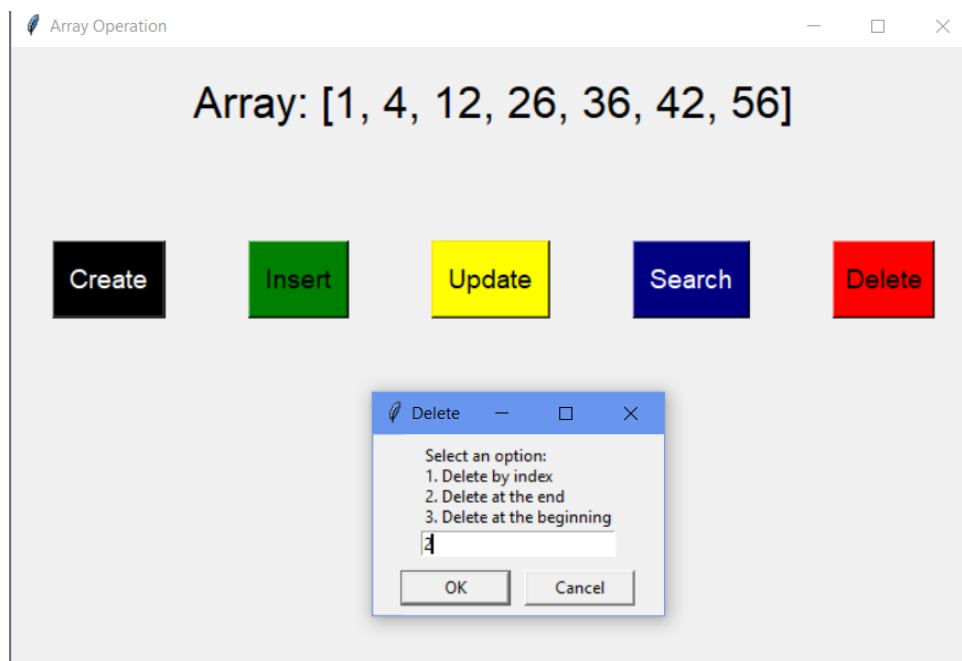


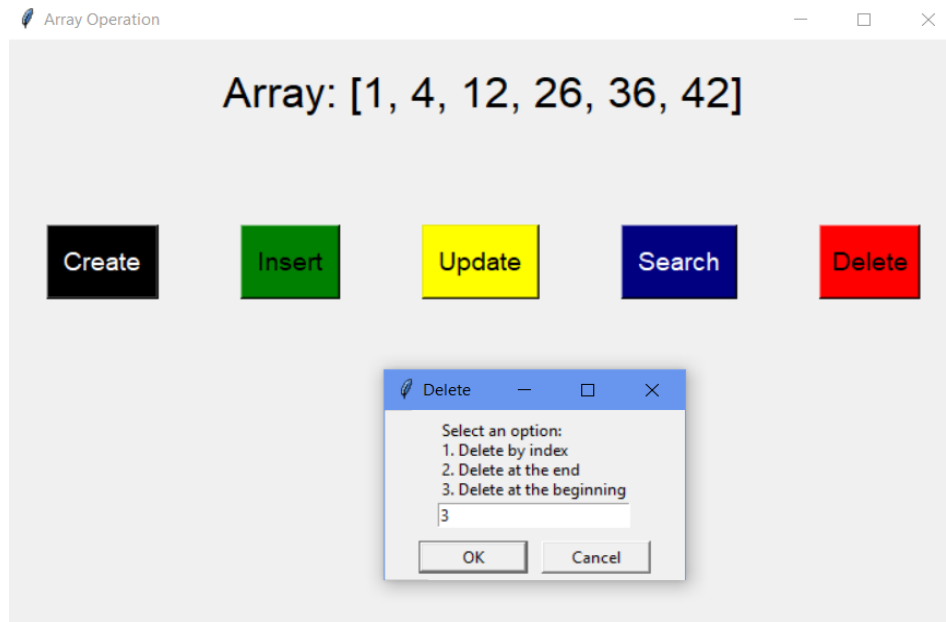


The results after deleting at index :

**Array: [1, 4, 12, 26, 36, 42, 56]**

After deleting the last element 56 from the list.



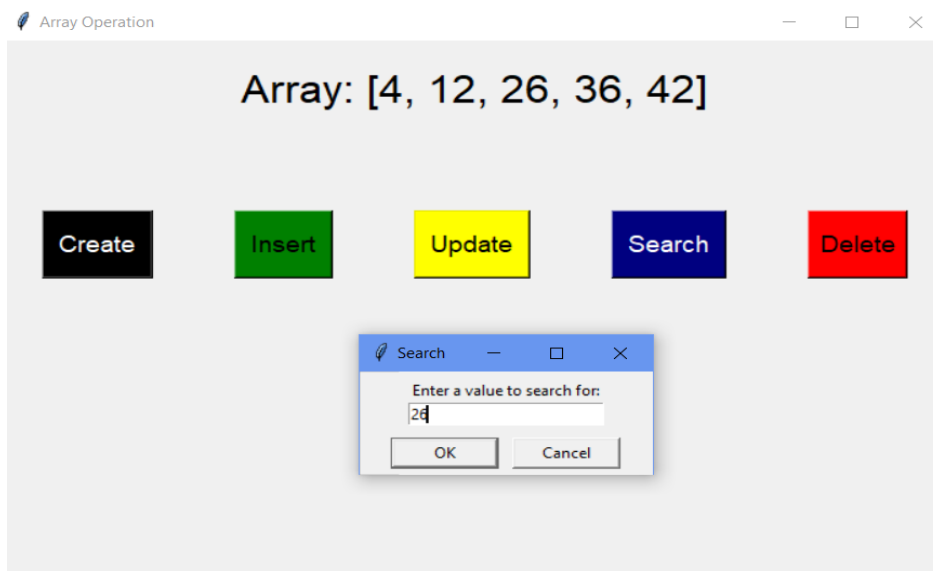


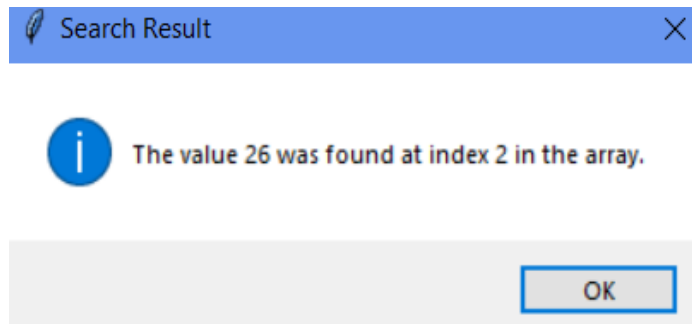
The results after deleting both the last and the first element from the list.

Array: [4, 12, 26, 36, 42]

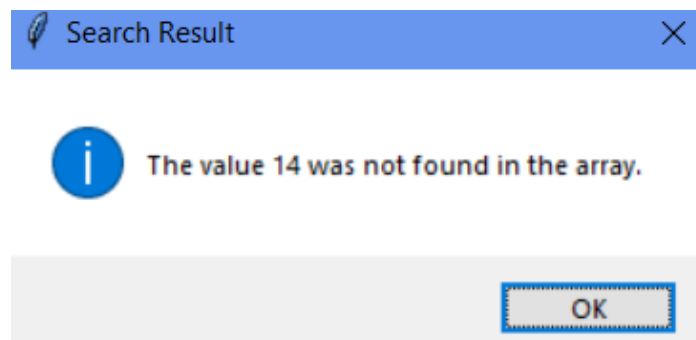
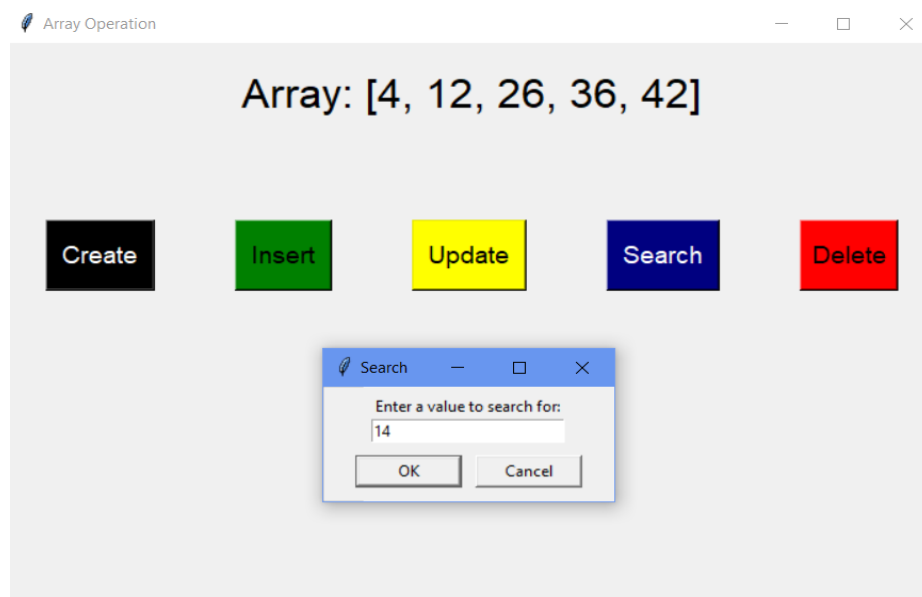
➤ Array Search:

Searching value 26, as it is in the array we get its position.

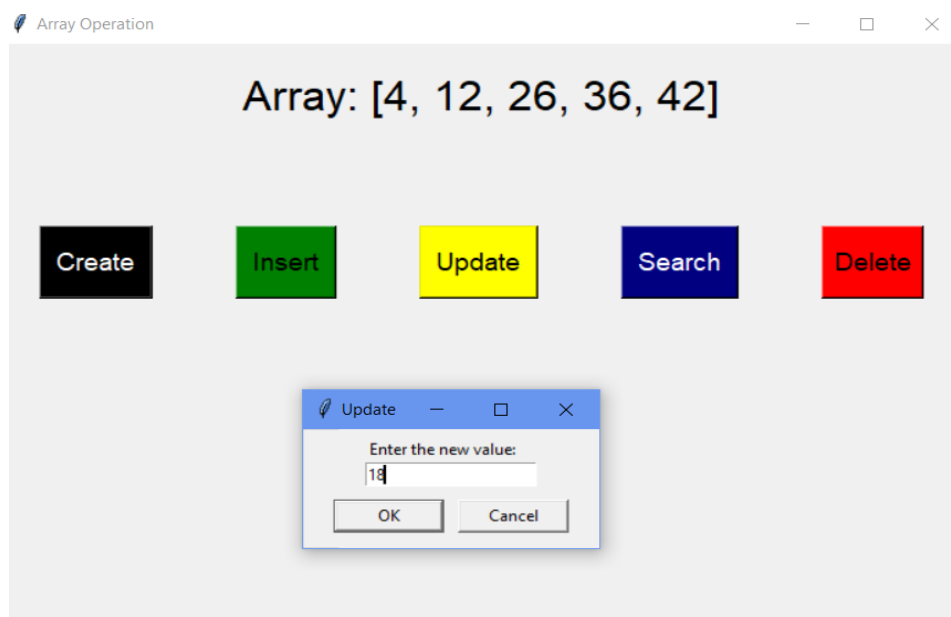
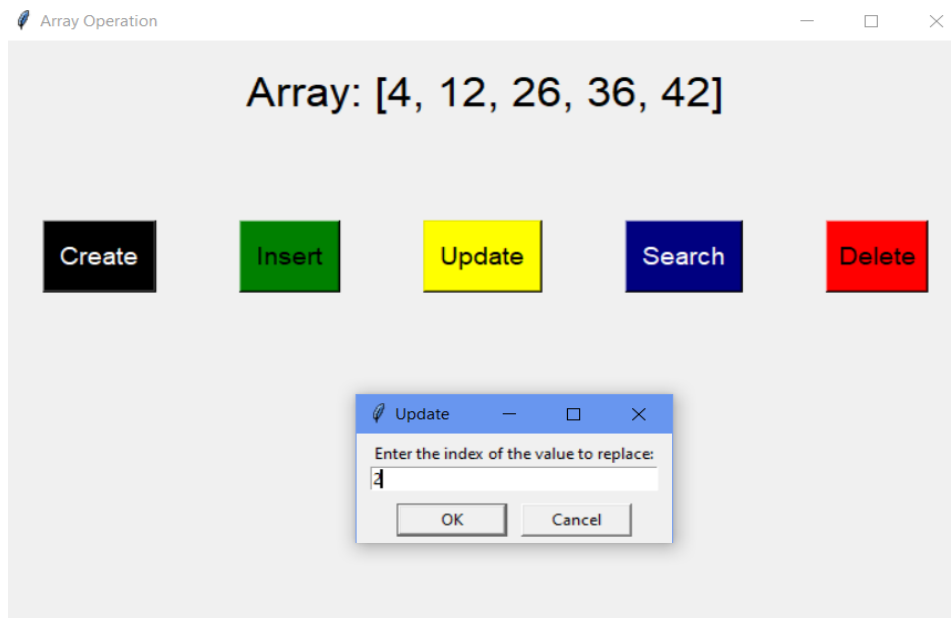




Searching value 14, it shows a message that it is not in the array.



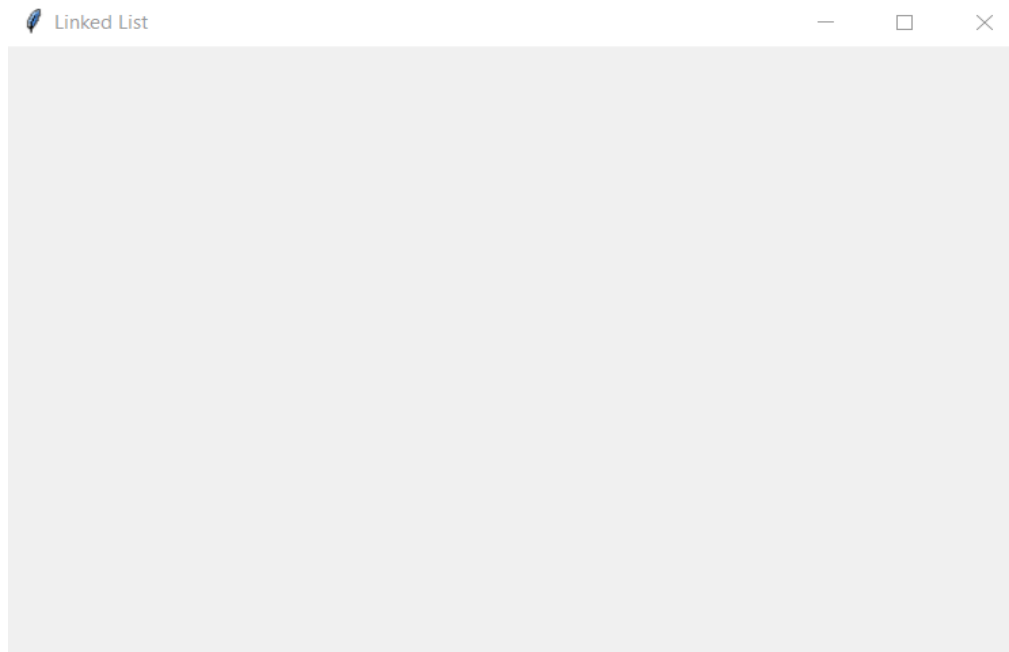
- Array Update: Replaces the value 26 with 18.



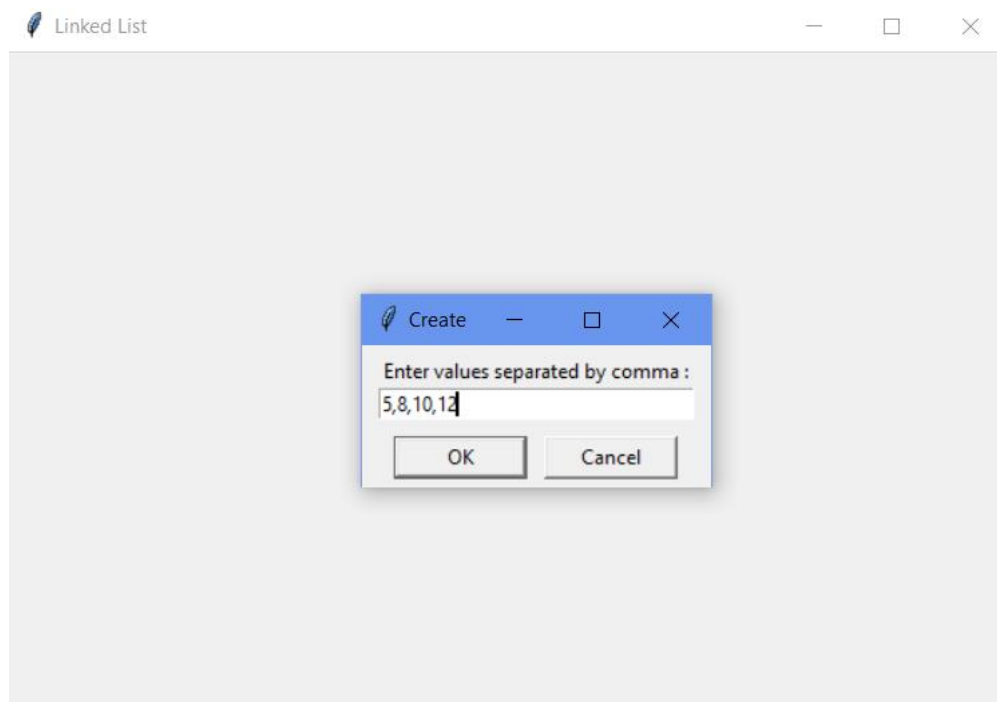
The results after updating the list :

Array: [4, 12, 18, 36, 42]

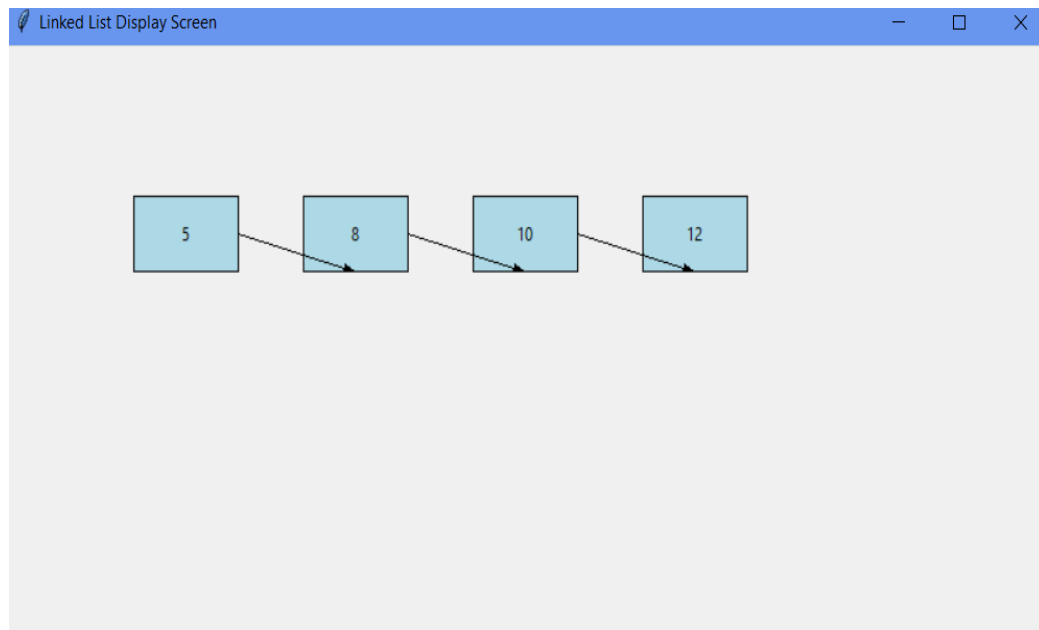
- **Linked List Window:** The secondary window for Linked List is a blank page.



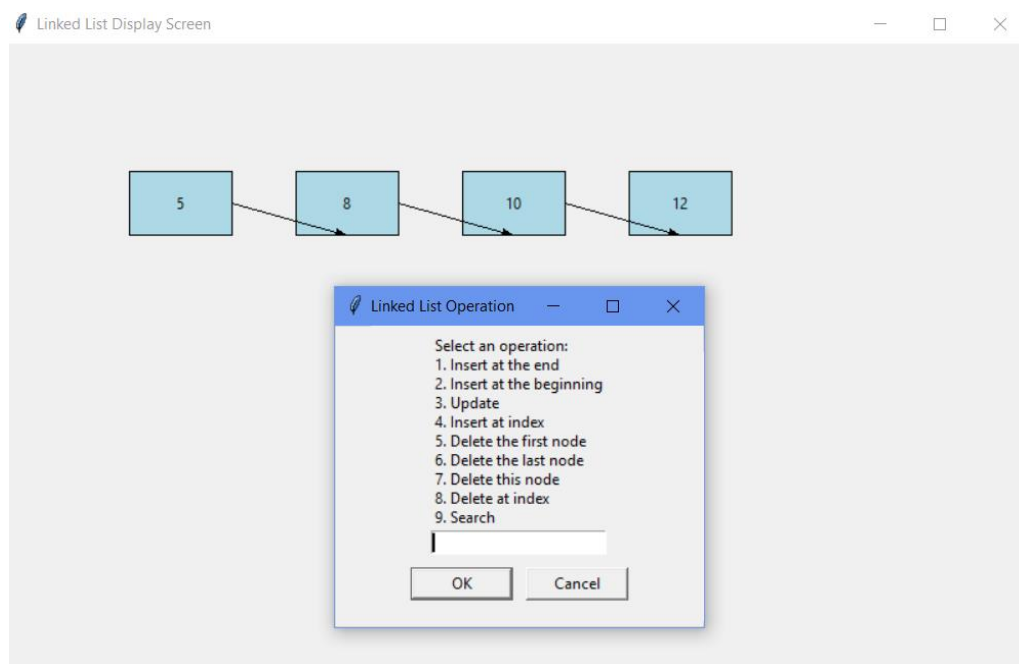
- **Mouse Action:** Clicking anywhere in the window prompts the user to enter values to create a linked list.



- *Linked List Output Display*: The linked list is displayed on a canvas, with each node represented by a rectangle containing its respective value.



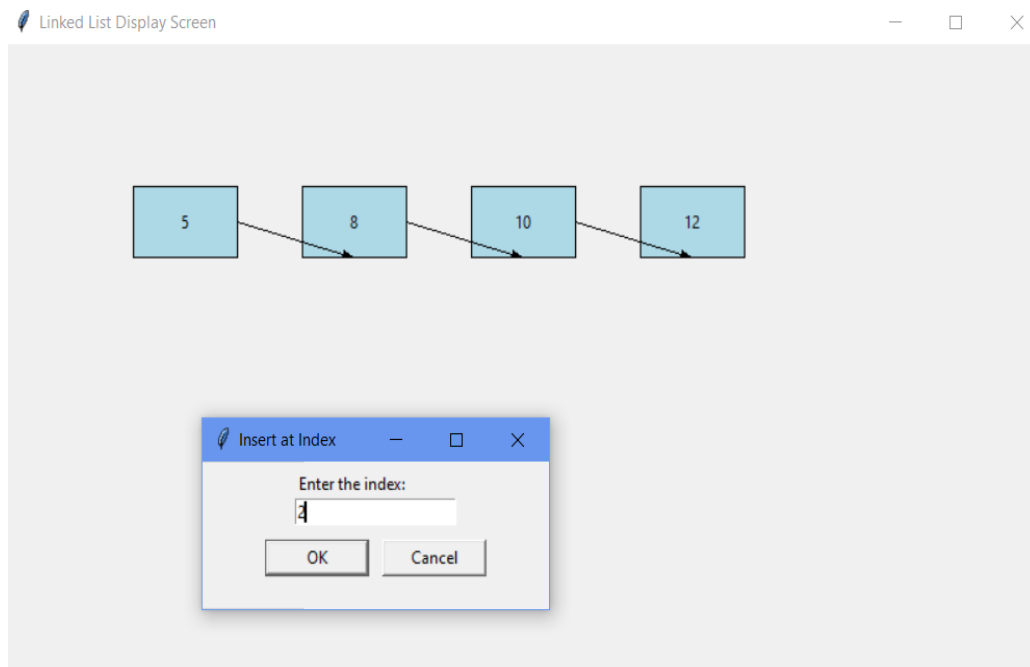
- *Linked list Operations*: By clicking any one of the nodes, it displays a dialog box with various operations that can be performed on the linked list.

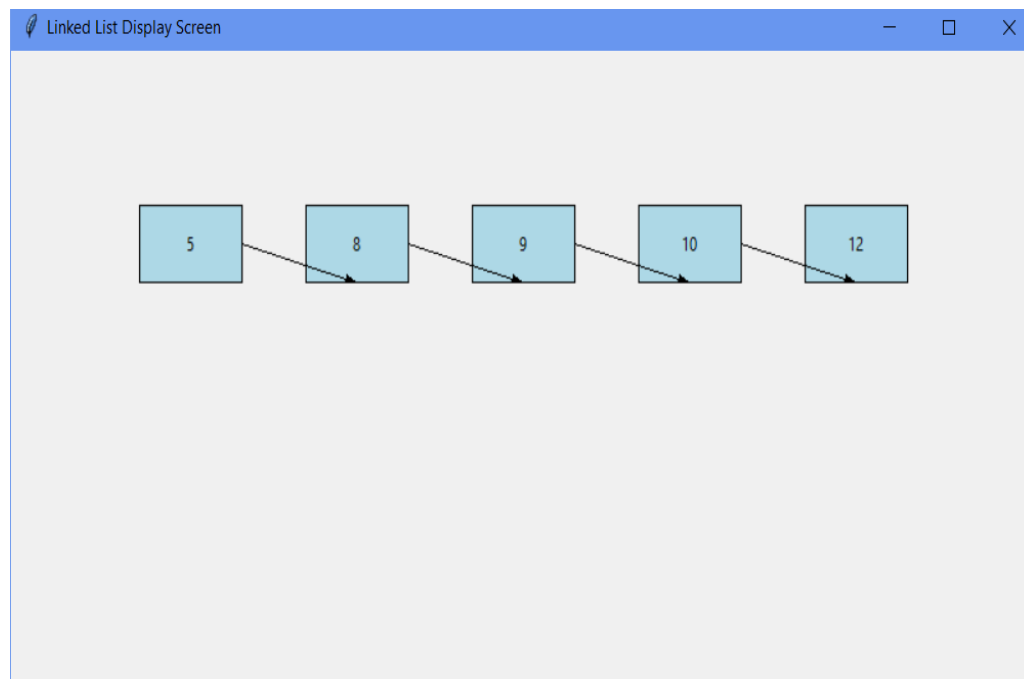
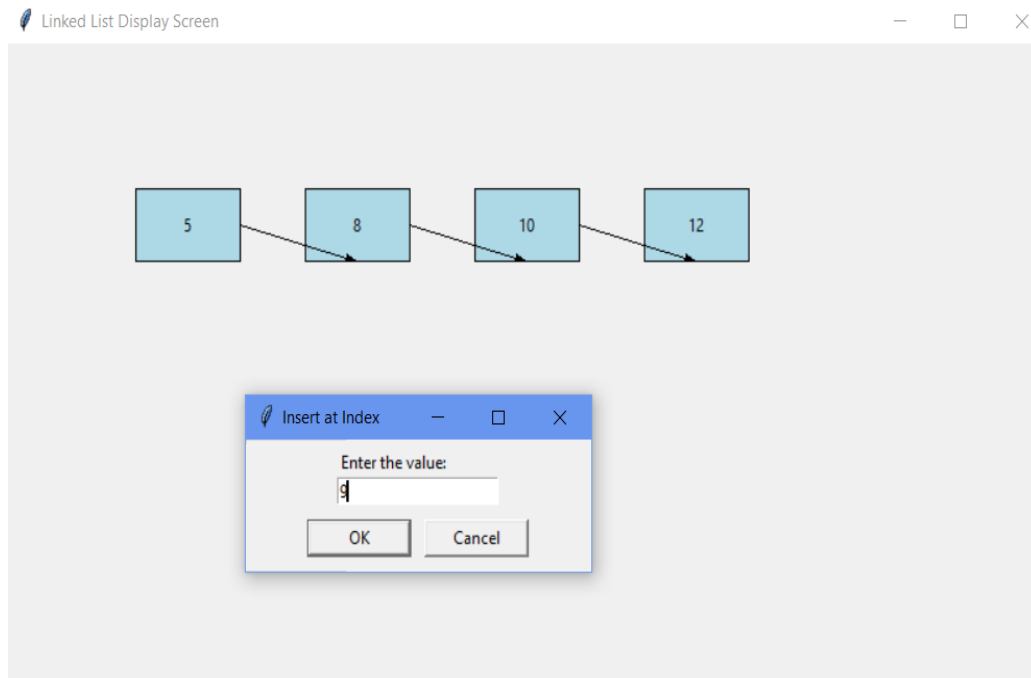




- Insert at the end, beginning, or index: Allows the user for a value and inserts it into the linked list at the specified position.
- Update: Allows the user to update the value of the clicked node.
- Delete the first node, last node, or clicked node: Performs the corresponding deletion operation on the linked list.
- Delete at index: Allows the user for an index and deletes the node at that position.
- Search: Allows the user for a value and displays a message box indicating whether the value was found in the linked list and its position.

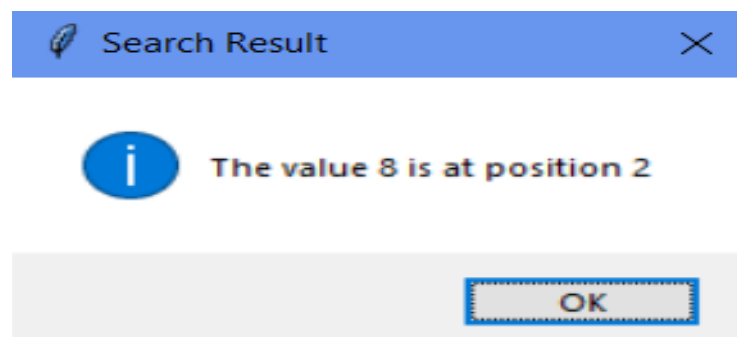
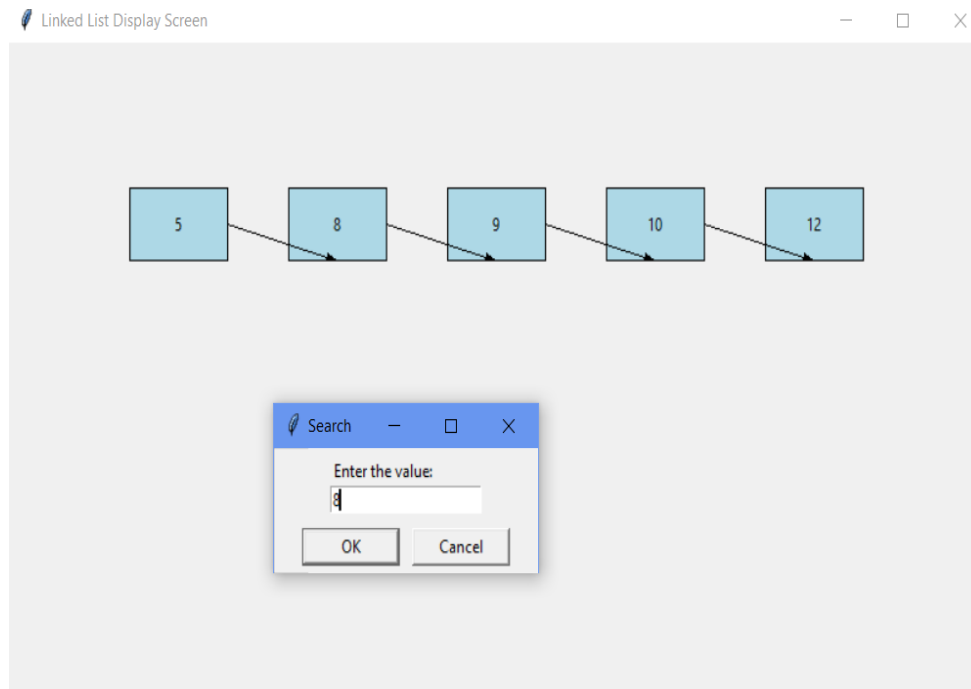
➤ *Linked list Insertion:* Inserting value 9 at index 2.



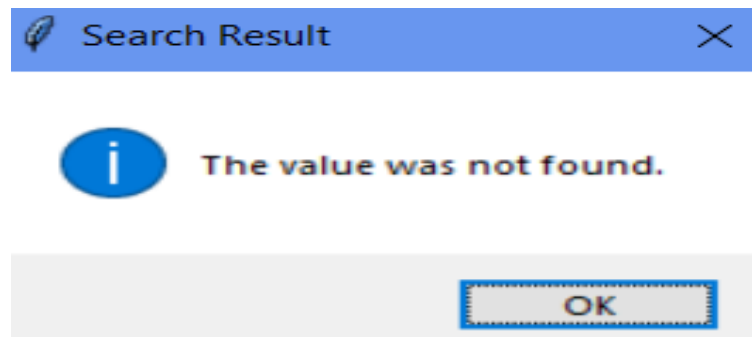
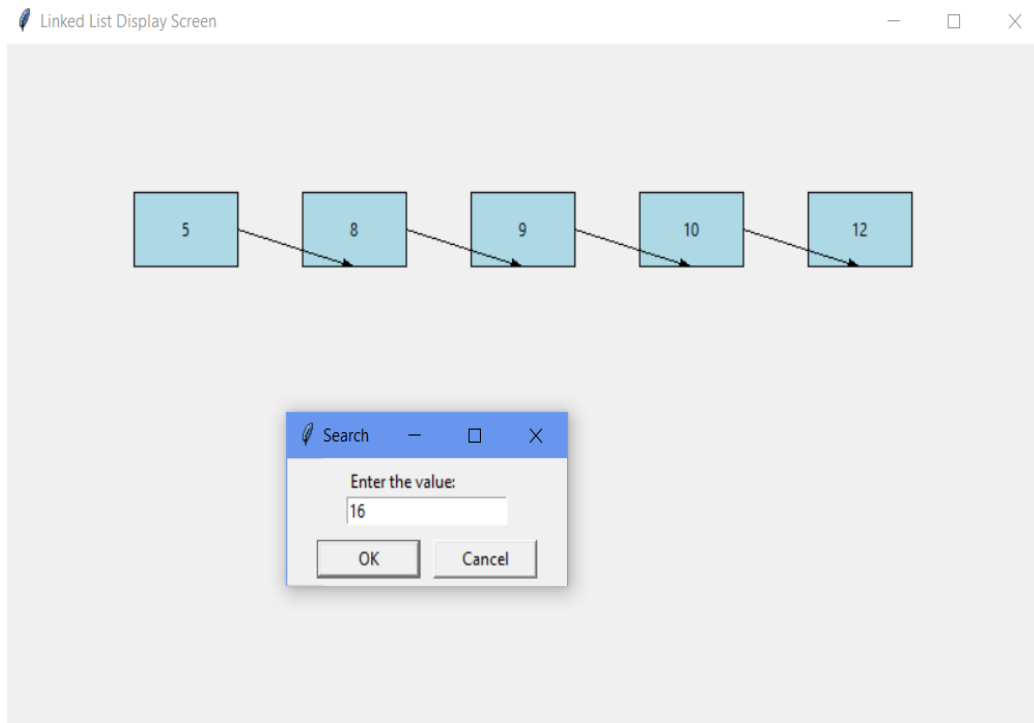


➤ Linked list Search:

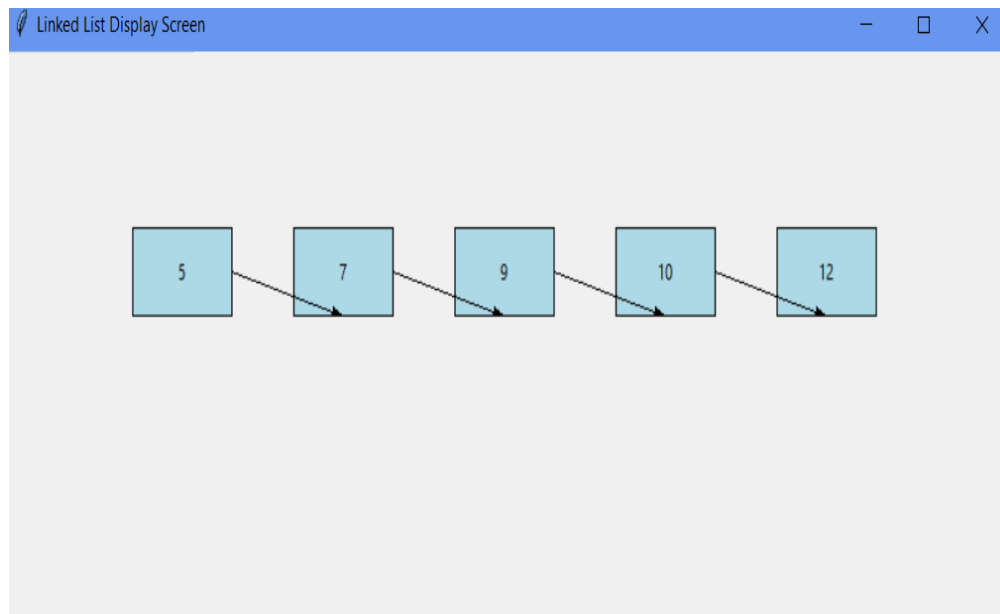
Searching the value 8, we get its position as it exists in the linked list.



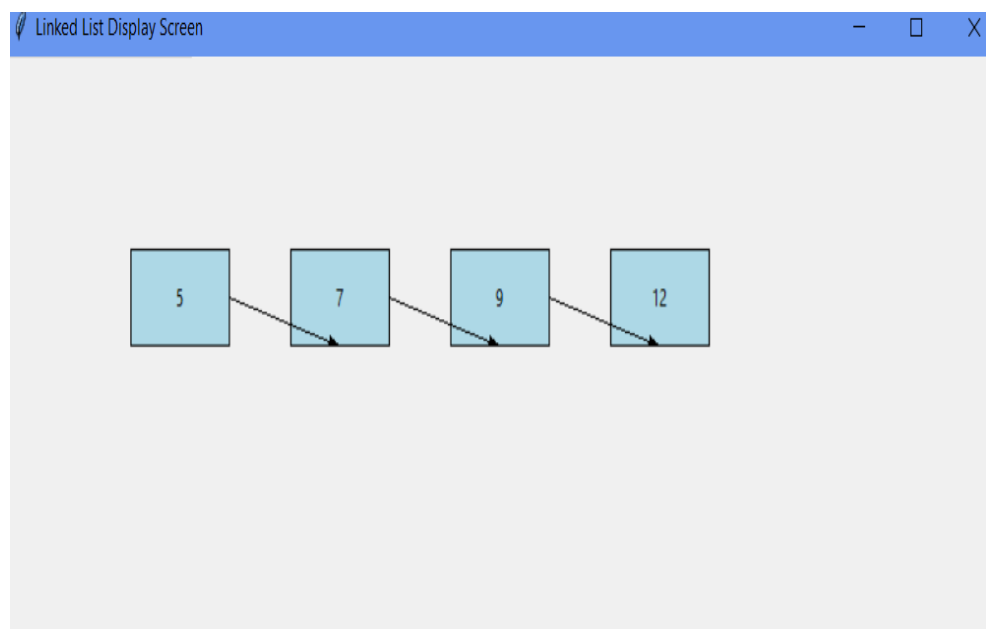
Searching the value 16, shows a message that it is not in the linked list.



- *Linked list Update:* After replacing the value 8 with 7.



- *Linked list Deletion:* After deleting the value 10 at index 3.



## **Dependencies :**

- JetBrainsIDE: Jet Brains IDEs are designed to enhance and streamline the software development process by providing a comprehensive set of features and tools for coding, debugging, testing, and version control. To built this project we used PyCharm, an IDE developed by Jet Brains specifically tailored for Python development, including support for Tkinter. PyCharm provides a built-in package manager called "PyCharm Package Manager" or "pip" and many more other packages. Installing all the necessary packages and libraries required for the program it gets successful results.

**25 June, 2023**

**North Western University**

**Khulna, Bangladesh.**