05. WAP to Implement Singly Linked List with following operations a) Create a linked list. b) Deletion of first element, specified element and last element in the list. c) Display the contents of the linked list.

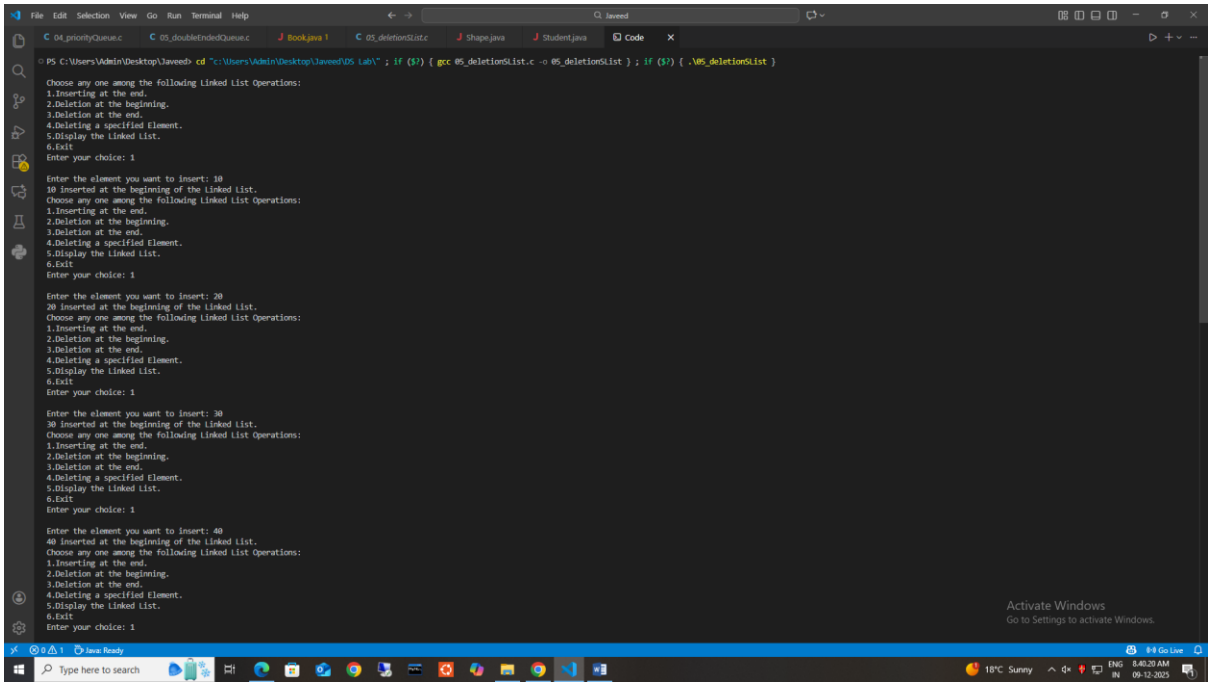
```c
#include <stdio.h>
#include <stdlib.h>
struct node
{
    /* data */
    int info;
    struct node *next;
};
struct node *insertAtEnd(struct node *start, int item)
{
    struct node *p, *temp;
    p = (struct node *)malloc(sizeof(struct node));
    p->info = item;
    p->next = NULL;
    if (start == NULL)
    {
        start = p;
    }
    else
    {
        temp = start;
        while (temp->next != NULL)
            temp = temp->next;
        temp->next = p;
    }
    return start;
}
void displayLinkedList(struct node *start)
{
    if (start == NULL)
    {
        printf("Linked List is Empty.");
        return;
    }
    else
    {
        struct node *temp = start;
        printf("Elements of the Linked list are: ");
        while (temp != NULL)
        {
            printf("%d ", temp->info);
            temp = temp->next;
        }
    }
```

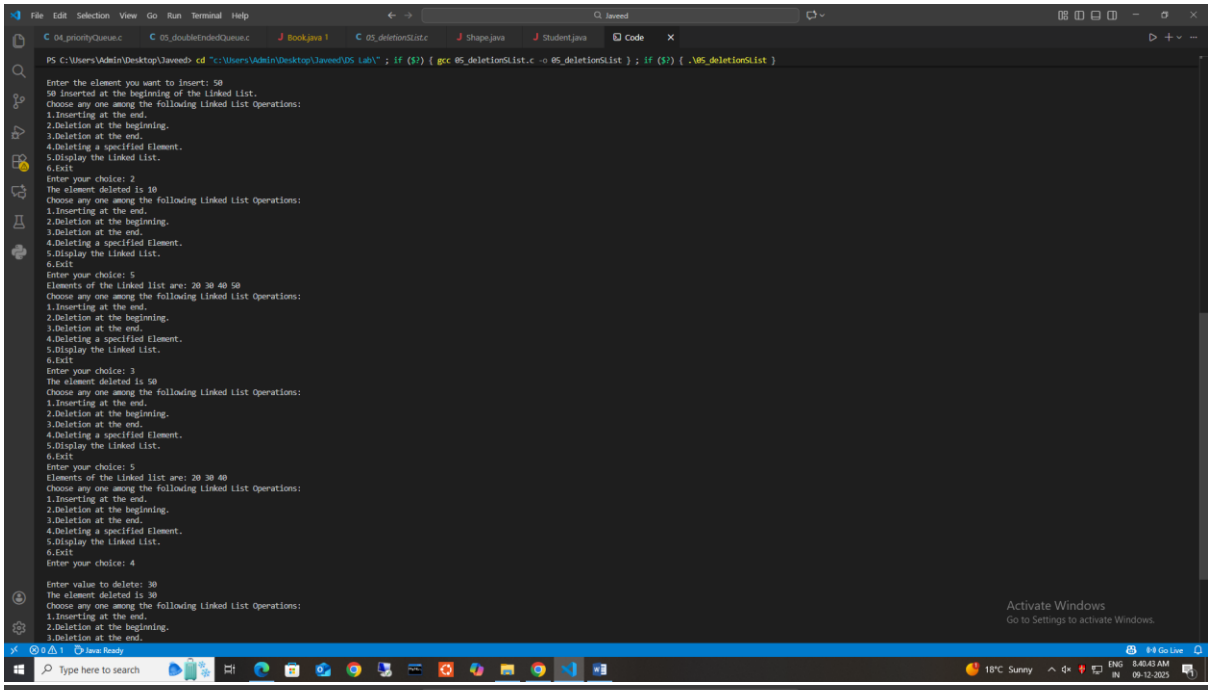
```
}
struct node *deleteFirst(struct node *start)
{
    if (start == NULL)
    {
        printf("Link is empty!\n");
        return start;
    }
    struct node *temp = start;
    start = start->next;
    printf("The element deleted is %d", temp->info);
    free(temp);
    return start;
}

struct node *deleteSpecific(struct node *start, int target)
{
    if (start == NULL)
    {
        printf("List is empty!\n");
        return start;
    }
    struct node *temp = start, *prev = NULL;
    if (temp != NULL && temp->info == target)
    {
        start = start->next;
        printf("The element deleted is %d", temp->info);
        free(temp);
        return start;
    }
    while (temp != NULL && temp->info != target)
    {
        prev = temp;
        temp = temp->next;
    }
    if (temp == NULL)
    {
        printf("Element not found");
    }
    prev->next = temp->next;
    printf("The element deleted is %d", temp->info);
    free(temp);
    return start;
}

struct node *deleteLast(struct node *start)
{
    if (start == NULL)
```

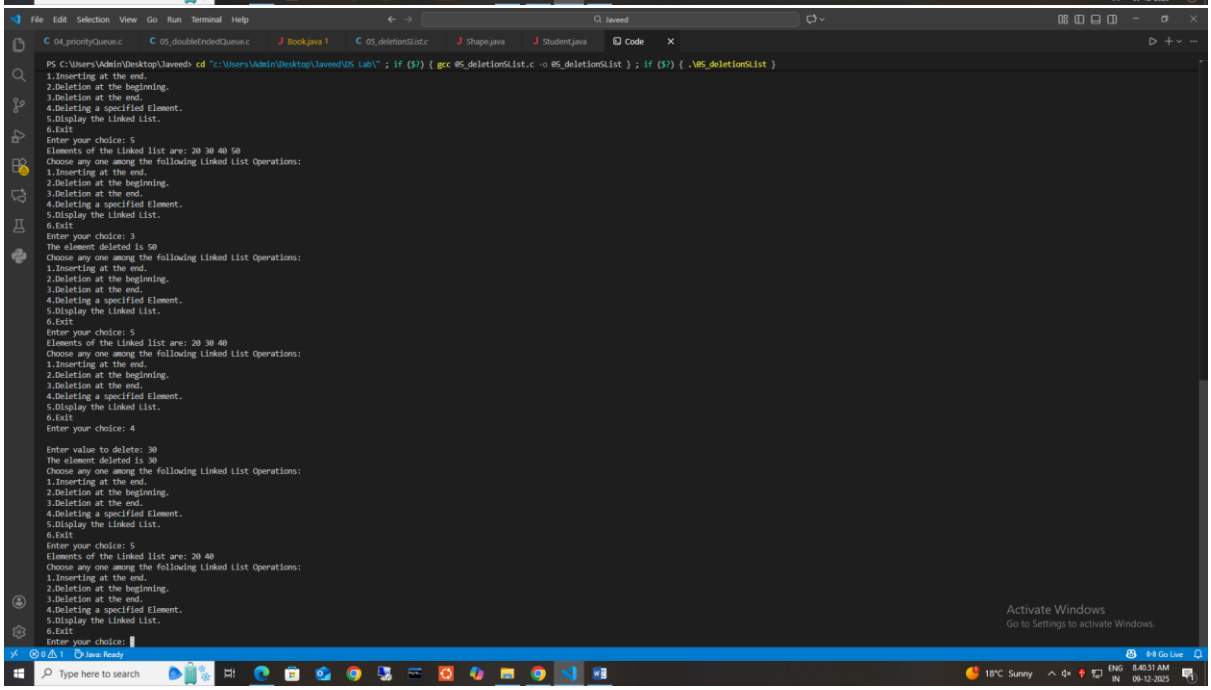
```
    {
        printf("List is empty!\n");
        return start;
    }
    if (start->next == NULL)
    {
        free(start);
        start = NULL;
        return start;
    }
    struct node *temp = start, *prev = NULL;
    while (temp->next != NULL)
    {
        prev = temp;
        temp = temp->next;
    }
    prev->next = NULL;
    printf("The element deleted is %d", temp->info);
    free(temp);
    return start;
}
int main()
{
    struct node *start = NULL;
    int choice, value;
    while (1)
    {
        printf("\nChoose any one among the following Linked List Operations:
");
        printf("\n1.Inserting at the end.\n2.Deletion at the
beginning.\n3.Deletion at the end.\n4.Deleting a specified Element.\n5.Display
the Linked List.\n6.Exit");
        printf("\nEnter your choice: ");
        scanf("%d", &choice);
        if (choice == 1)
        {
            printf("\nEnter the element you want to insert: ");
            scanf("%d", &value);
            start = insertAtEnd(start, value);
            printf("%d inserted at the beginning of the Linked List.", value);
        }
        else if (choice == 2)
        {
            start = deleteFirst(start);
        }
        else if (choice == 3)
        {
            start = deleteLast(start);
```

```
        }
        else if (choice == 4)
        {
            printf("\nEnter value to delete: ");
            scanf("%d", &value);
            start = deleteSpecific(start, value);
        }
        else if (choice == 5)
        {
            displayLinkedList(start);
        }
        else if (choice == 6)
        {
            printf("\nExiting the program!!");
            return 0;
        }
        else
        {
            printf("\nEnter a valid choice!");
        }
    }
    return 0;
}
```

OUTPUT:




PS C:\Users\Admin\Desktop\Javeed> cd "c:\Users\Admin\Desktop\Javeed\DS Lab\" ; if ($?) { gcc 05_deletionSList.c -o 05_deletionSList } ; if ($?) { .\05_deletionSList }
Enter the element you want to insert: 50
50 inserted at the beginning of the Linked List.
Choose any one among the following Linked List Operations:
1.Inserting at the end.
2.Deletion at the beginning.
3.Deletion at the end.
4.Deleting a specified Element.
5.Display the Linked List.
6.Exit
Enter your choice: 2
The element deleted is 10
Choose any one among the following Linked List Operations:
1.Inserting at the end.
2.Deletion at the beginning.
3.Deletion at the end.
4.Deleting a specified Element.
5.Display the Linked List.
6.Exit
Enter your choice: 5
Elements of the Linked list are: 20 30 40 50
Choose any one among the following Linked List Operations:
1.Inserting at the end.
2.Deletion at the beginning.
3.Deletion at the end.
4.Deleting a specified Element.
5.Display the Linked List.
6.Exit
Enter your choice: 3
The element deleted is 50
Choose any one among the following Linked List Operations:
1.Inserting at the end.
2.Deletion at the beginning.
3.Deletion at the end.
4.Deleting a specified Element.
5.Display the Linked List.
6.Exit
Enter your choice: 5
Elements of the Linked list are: 20 30 40
Choose any one among the following Linked List Operations:
1.Inserting at the end.
2.Deletion at the beginning.
3.Deletion at the end.
4.Deleting a specified Element.
5.Display the Linked List.
6.Exit
Enter your choice: 4
Enter value to delete: 30
The element deleted is 30
Choose any one among the following Linked List Operations:
1.Inserting at the end.
2.Deletion at the beginning.
3.Deletion at the end.
Activate Windows
Go to Settings to activate Windows.


PS C:\Users\Admin\Desktop\Javeed> cd "c:\Users\Admin\Desktop\Javeed\DS Lab\" ; if ($?) { gcc 05_deletionSList.c -o 05_deletionSList } ; if ($?) { .\05_deletionSList }
1.Inserting at the end.
2.Deletion at the beginning.
3.Deletion at the end.
4.Deleting a specified Element.
5.Display the Linked List.
6.Exit
Enter your choice: 5
Elements of the Linked list are: 20 30 40 50
Choose any one among the following Linked List Operations:
1.Inserting at the end.
2.Deletion at the beginning.
3.Deletion at the end.
4.Deleting a specified Element.
5.Display the Linked List.
6.Exit
Enter your choice: 3
The element deleted is 50
Choose any one among the following Linked List Operations:
1.Inserting at the end.
2.Deletion at the beginning.
3.Deletion at the end.
4.Deleting a specified Element.
5.Display the Linked List.
6.Exit
Enter your choice: 5
Elements of the Linked list are: 20 30 40
Choose any one among the following Linked List Operations:
1.Inserting at the end.
2.Deletion at the beginning.
3.Deletion at the end.
4.Deleting a specified Element.
5.Display the Linked List.
6.Exit
Enter your choice: 4
Enter value to delete: 30
The element deleted is 30
Choose any one among the following Linked List Operations:
1.Inserting at the end.
2.Deletion at the beginning.
3.Deletion at the end.
4.Deleting a specified Element.
5.Display the Linked List.
6.Exit
Enter your choice: 5
Elements of the Linked list are: 20 40
Choose any one among the following Linked List Operations:
1.Inserting at the end.
2.Deletion at the beginning.
3.Deletion at the end.
4.Deleting a specified Element.
5.Display the Linked List.
6.Exit
Enter your choice:
Activate Windows
Go to Settings to activate Windows.