

**Aim:**

Write a program that uses functions to perform the following **operations on singly linked list**

- i) Creation
- ii) Insertion
- iii) Deletion
- iv) Traversal

**Source Code:**[singlelinkedlistalloperations.c](#)

```
#include<stdio.h>
#include<malloc.h>
#include<stdlib.h>
struct node {
    int data;
    struct node *next;
}
*head=NULL,*tail=NULL;
typedef struct node *NODE;NODE temp,newNode,ptr,prev;
void insert();
void Delete();
void display();
void count();
void main()
{
    int option;
    printf("Singly Linked List Example - All Operations\n");
    while(1){
        printf("Options\n");
        printf("1 : Insert elements into the linked list\n");
        printf("2 : Delete elements from the linked list
\n");
        printf("3 : Display the elements in the linked
list\n");
        printf("4 : Count the elements in the li
nked list\n");
        printf("5 : Exit()\n");
        printf("Enter your option :
");
        scanf("%d",&option);
        if(option>0 && o
ption<=5){
            ch(option){
                1:
                rt();
                break;
            }
        }
    }
}
```

case 2:

```
Delete();
```

```
break;
```

case 3:

```

        display();           break;
case 4:      count();       break;      case 5:      exit(0);
break;          }     }else{
printf("Enter options from 1 to 5\n");
break;
}
}
}

void insert(){
    newNode=(NODE)malloc(sizeof(struct node));
    printf("Enter elements for inserting into linked list : ");
    scanf("%d",&newNode->data);
    newNode->next=NULL;
    if(head==NULL){
        head=newNode;
        tail=newNode;
    }else{
        tail->next=newNode;
        tail=newNode;
    }
}
void Delete(){
    int cnt=0,pos,i=1;
    ptr=temp=head;
    while(ptr!=NULL){
        cnt++;
        ptr=ptr->next;
    }
    printf("Enter position of the element for deleteing the element :
");
    scanf("%d",&pos);
    if(pos>0&&pos<=cnt){
        if(pos==1){
            head=head->next;
            free(head);
        }else{
            while(i<pos){
                i++;
                prev=temp;
                temp=temp->next;
            }
            prev->next=temp->next;
            free(temp);
        }
        printf("Deleted successfully\n");
    } else
        printf("Invalid position\n");
}
void display(){
    printf("The elements in the linked list are : ");
    temp=head;
    while(temp!=NULL){
        printf("%d ",temp->data);
        temp=temp->next;
    }
}

```

```

        printf("\n");
    void count() {
        int count=0;
        printf("No of elements in the linked list are : ");
        temp=head;
        while(temp!=NULL) {
            count++;
            temp=temp->next;
        }
        printf("%d\n",count);
    }
}

```

### Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Singly Linked List Example - All Operations 1
Options 1
1 : Insert elements into the linked list 1
2 : Delete elements from the linked list 1
3 : Display the elements in the linked list 1
4 : Count the elements in the linked list 1
5 : Exit() 1
Enter your option : 1
Enter elements for inserting into linked list : 111
Options 1
1 : Insert elements into the linked list 1
2 : Delete elements from the linked list 1
3 : Display the elements in the linked list 1
4 : Count the elements in the linked list 1
5 : Exit() 1
Enter your option : 1
Enter elements for inserting into linked list : 222
Options 1
1 : Insert elements into the linked list 1
2 : Delete elements from the linked list 1
3 : Display the elements in the linked list 1
4 : Count the elements in the linked list 1
5 : Exit() 1
Enter your option : 1
Enter elements for inserting into linked list : 333
Options 1
1 : Insert elements into the linked list 1
2 : Delete elements from the linked list 1
3 : Display the elements in the linked list 1
4 : Count the elements in the linked list 1
5 : Exit() 1
Enter your option : 1
Enter elements for inserting into linked list : 444
Options 3

```

1 : Insert elements into the linked list 3
2 : Delete elements from the linked list 3
3 : Display the elements in the linked list 3
4 : Count the elements in the linked list 3
5 : Exit() 3
Enter your option : 3
The elements in the linked list are : 111 222 333 444 2
Options 2
1 : Insert elements into the linked list 2
2 : Delete elements from the linked list 2
3 : Display the elements in the linked list 2
4 : Count the elements in the linked list 2
5 : Exit() 2
Enter your option : 2
Enter position of the element for deleteing the element : 2
Deleted successfully 3
Options 3
1 : Insert elements into the linked list 3
2 : Delete elements from the linked list 3
3 : Display the elements in the linked list 3
4 : Count the elements in the linked list 3
5 : Exit() 3
Enter your option : 3
The elements in the linked list are : 111 333 444 4
Options 4
1 : Insert elements into the linked list 4
2 : Delete elements from the linked list 4
3 : Display the elements in the linked list 4
4 : Count the elements in the linked list 4
5 : Exit() 4
Enter your option : 4
No of elements in the linked list are : 3 5
Options 5
1 : Insert elements into the linked list 5
2 : Delete elements from the linked list 5
3 : Display the elements in the linked list 5
4 : Count the elements in the linked list 5
5 : Exit() 5
Enter your option : 5

```

Test Case - 2
User Output
Singly Linked List Example - All Operations 1
Options 1
1 : Insert elements into the linked list 1
2 : Delete elements from the linked list 1
3 : Display the elements in the linked list 1
4 : Count the elements in the linked list 1
5 : Exit() 1
Enter your option : 1
Enter elements for inserting into linked list : 001

Options 1

1 : Insert elements into the linked list 1  
2 : Delete elements from the linked list 1  
3 : Display the elements in the linked list 1  
4 : Count the elements in the linked list 1  
5 : Exit() 1

Enter your option : 1

Enter elements for inserting into linked list : 010

Options 1

1 : Insert elements into the linked list 1  
2 : Delete elements from the linked list 1  
3 : Display the elements in the linked list 1  
4 : Count the elements in the linked list 1  
5 : Exit() 1

Enter your option : 1

Enter elements for inserting into linked list : 100

Options 1

1 : Insert elements into the linked list 1  
2 : Delete elements from the linked list 1  
3 : Display the elements in the linked list 1  
4 : Count the elements in the linked list 1  
5 : Exit() 1

Enter your option : 1

Enter elements for inserting into linked list : 101

Options 3

1 : Insert elements into the linked list 3  
2 : Delete elements from the linked list 3  
3 : Display the elements in the linked list 3  
4 : Count the elements in the linked list 3  
5 : Exit() 3

Enter your option : 3

The elements in the linked list are : 1 10 100 101 2

Options 2

1 : Insert elements into the linked list 2  
2 : Delete elements from the linked list 2  
3 : Display the elements in the linked list 2  
4 : Count the elements in the linked list 2  
5 : Exit() 2

Enter your option : 2

Enter position of the element for deleteing the element : 3

Deleted successfully 3

Options 3

1 : Insert elements into the linked list 3  
2 : Delete elements from the linked list 3  
3 : Display the elements in the linked list 3  
4 : Count the elements in the linked list 3  
5 : Exit() 3

Enter your option : 3

The elements in the linked list are : 1 10 101 4

Options 4

1 : Insert elements into the linked list 4  
2 : Delete elements from the linked list 4

```
3 : Display the elements in the linked list 4
4 : Count the elements in the linked list 4
5 : Exit() 4
Enter your option : 4
No of elements in the linked list are : 3 5
Options 5
1 : Insert elements into the linked list 5
2 : Delete elements from the linked list 5
3 : Display the elements in the linked list 5
4 : Count the elements in the linked list 5
5 : Exit() 5
Enter your option : 5
```