

## Singly Linked List using Front insertion/deletion and End insertion/deletion

```
#include<stdio.h>
#include<stdlib.h>
#include<malloc.h>
void create(),display(),insert_front(),insert_end(),delete_front(),delete_end();
struct node{
    int data;
    struct node *link;
};
struct node *first=NULL, *last=NULL, *cur, *next,*prev;

void create(){
    int i=0,n;
    printf("Enter the no.of nodes you want in linked list:\n");
    scanf("%d",&n);
    while(i<n){
        cur=(struct node*)malloc(sizeof(struct node));
        printf("Enter the data:\n");
        scanf("%d",&cur->data);
        if(i==0){
            cur->link=NULL;
            first=cur;
        }
        else{
            cur->link=first;
            first=cur;
        }
        i++;
    }
}

void insert_front(){
```

```

    cur=(struct node*)malloc(sizeof(struct node));
    printf("Enter the data:\n");
    scanf("%d",&cur->data);
    if(first==NULL){
        cur->link=NULL;
        first=cur;
    }
    else{
        cur->link=first;
        first=cur;
    }
}

void insert_end(){
    cur=(struct node*)malloc(sizeof(struct node));
    printf("Enter the data:\n");
    scanf("%d",&cur->data);
    if(first==NULL){
        cur->link=NULL;
        first=cur;
    }
    else{
        cur->link=NULL;
        next=first;
        while(next->link!=NULL)
            next=next->link;
        next->link=cur;
    }
}

void delete_front(){
    if(first!=NULL){
        cur=first;
        first=first->link;
        printf("Node deleted contains: %d\n",cur->data);
    }
}

```

```

        free(cur);
    }
    else{
        printf("Void Deletion\n");
    }
}

void delete_end(){
    if(first!=NULL){
        next=first;
        prev=first;
        while(next->link!=NULL){
            prev=next;
            next=next->link;
        }
        prev->link=NULL;
        printf("Node deleted contains: %d\n",next->data);
        free(next);
    }
}

void display(){
    cur=first;
    printf("\n");
    while(cur!=NULL){
        printf("%d\n",cur->data);
        cur=cur->link;
    }
}

void main(){
    int ch;
    printf("\t\tSINGLY LINKED LIST\t\t");
    do{
        printf("\n1.Create\n2.Display\n3.Insert front\n4.Insert
end\n5.Delete front\n6.Delete end\n7.Exit\n");
    }
}

```

```
printf("\n\nEnter your choice:\n");
scanf("%d",&ch);
switch(ch){
    case 1: create(); break;
    case 2: display(); break;
    case 3: insert_front(); break;
    case 4: insert_end(); break;
    case 5: delete_front(); break;
    case 6: delete_end(); break;
    case 7: exit(0);
    default: printf("\nInvalid Choice\n");
}
}while(1);
}
```