```
//SINGLY LINKED LIST
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>

struct node
{
        int data;
        struct node *next;
};
struct node *start = NULL;
void create_ll();
void display();
void insert_beg();
void insert_end();
void insert_before();
void insert_after();
void delete_beg();
void delete_end();
void delete_node();
void main()
{
        int option;
        do
        {


                printf("\n\n *****MAIN MENU *****");
                printf("\n 1: Create a list");
                printf("\n 2: Display the list");
                printf("\n 3: Add a node at the beginning");
                printf("\n 4: Add a node at the end");
                printf("\n 5: Add a node before a given node");
                printf("\n 6: Add a node after a given node");
                printf("\n 7: Delete a node from the beginning");
                printf("\n 8: Delete a node from the end");
                printf("\n 9: Delete a given node");
                printf("\n 10: EXIT");
                printf("\n\n Enter your option : ");
                scanf("%d", &option);
                switch(option)
                {
```

```
            case 1:
                create_ll();
                printf("\n LINKED LIST CREATED");
                break;
            case 2:
                display();
                break;
            case 3:
                insert_beg();
                break;
            case 4:
                insert_end();
                break;
            case 5:
                insert_before();
                break;
            case 6:
                insert_after();
                break;
            case 7:
                delete_beg();
                break;
            case 8:
                delete_end();
                break;
            case 9:
                delete_node();
                break;


        }
    }while(option !=10);

}

void create_ll()
{
    struct node *new_node, *ptr;
    int num;
    printf("\n Enter -1 to end");
    printf("\n Enter the data : ");
    scanf("%d", &num);
    while(num!=-1)
    {
        new_node = (struct node *)malloc(sizeof(struct node));
        new_node -> data=num;
        if(start==NULL)
        {
            new_node -> next = NULL;
            start = new_node;
        }
        else
        {
            ptr=start;
            while(ptr->next!=NULL)
            {
                ptr=ptr->next;
            }
            ptr->next = new_node;
            new_node->next=NULL;
        }
        printf("\n Enter the data : ");
        scanf("%d", &num);
    }

}
void display()
{
    struct node *ptr;
    ptr = start;
    while(ptr != NULL)
    {
        printf("\t %d", ptr -> data);
        ptr = ptr -> next;
    }

}
void insert_beg()
{
    struct node *new_node;
    int num;
    printf("\n Enter the data : ");
    scanf("%d", &num);
    new_node = (struct node *)malloc(sizeof(struct node));
```

```
        new_node -> data = num;
        new_node -> next = start;
        start = new_node;

}
void insert_end()
{
        struct node *ptr, *new_node;
        int num;
        printf("\n Enter the data : ");
        scanf("%d", &num);
        new_node = (struct node *)malloc(sizeof(struct node));
        new_node -> data = num;
        new_node -> next = NULL;
        ptr = start;
        while(ptr -> next != NULL)
                ptr = ptr -> next;
        ptr -> next = new_node;

}
void insert_before()
{
        struct node *new_node, *ptr, *preptr;
        int num, val;
        printf("\n Enter the data : ");
        scanf("%d", &num);
        printf("\n Enter the value before which the data has to be inserted : ");
        scanf("%d", &val);
        new_node = (struct node *)malloc(sizeof(struct node));
        new_node -> data = num;
        ptr = start;
        while(ptr -> data != val)
        {
                preptr = ptr;
                ptr = ptr -> next;
        }
        preptr -> next = new_node;
        new_node -> next = ptr;

}
void insert_after()
{
        struct node *new_node, *ptr, *preptr;
        int num, val;
        printf("\n Enter the data : ");
        scanf("%d", &num);
        printf("\n Enter the value after which the data has to be inserted : ");
        scanf("%d", &val);
        new_node = (struct node *)malloc(sizeof(struct node));
        new_node -> data = num;
        ptr = start;
        preptr = ptr;
        while(preptr -> data != val)
        {
                preptr = ptr;
                ptr = ptr -> next;
        }
        preptr -> next=new_node;
        new_node -> next = ptr;

}
void delete_beg()
{
        struct node *ptr;
        ptr = start;
        start = start -> next;
        free(ptr);

}
void delete_end()
{
        struct node *ptr, *preptr;
        ptr = start;
        while(ptr -> next != NULL)
        {
                preptr = ptr;
                ptr = ptr -> next;
        }
        preptr -> next = NULL;
        free(ptr);

}
void delete_node()
{
        struct node *ptr, *preptr;
        int val;
        printf("\n Enter the value of the node which has to be deleted : ");
        scanf("%d", &val);
        ptr = start;
        if(ptr -> data == val)
        {
                delete_beg();

        }
        else
        {
                while(ptr -> data != val)
                {
                        preptr = ptr;
                        ptr = ptr -> next;
                }
                preptr -> next = ptr -> next;
                free(ptr);
        }


}
```