Zayd Ahmed
1BM21CS254

Program to Perform Creation, Insertion, Deletion & displaying in Doubly Linked list

Input:-

```
//a)Create a doubly linked list.
//b)Insert a new node to the left of the node.
//c)Delete the node based on a specific value
//d)Display the contents of the list

#include <stdio.h>
#include <stdlib.h>

struct node{
    struct node* rlink;
    struct node* llink;
    int data;
};
typedef struct node node;
node *start,*new,*curr;

void create(){
    int ch;
    start=(node *)malloc(sizeof(node));
    printf("enter the element\n");
    scanf("%d",&start->data);
    curr=start;
    curr->llink=NULL;
    printf("do you want to enter more elements(1/0)\n");
    scanf("%d",&ch);
    while(ch==1){
        new=(node *)malloc(sizeof(node));
        printf("enter the element\n");
        scanf("%d",&new->data);
        curr->rlink=new;
        new->llink=curr;
        curr=new;
        printf("do you want to enter more elements(1/0)\n");
```

```
        scanf("%d",&ch);
    }
    curr->rlink=NULL;
}

void insert(){
    if(start==NULL){
        start=(node*)malloc(sizeof(node));
        printf("enter the element\n");
        scanf("%d",&start->data);
        start->llink=NULL;
        start->rlink=NULL;
        curr=start;
        return;
    }
    else{
        new=(node*)malloc(sizeof(node));
        printf("enter the element\n");
        scanf("%d",&new->data);
        new->rlink=start;
        start->llink=new;
        start=new;
        start->llink=NULL;
        return;
    }
}

void delete(int del){
    if(start==NULL){
        printf("empty linked list\n");
        return;
    }
    else{
        node *temp=start;
        while(temp->rlink!=NULL && temp->data!=del){
            temp=temp->rlink;
        }
        if(temp->data==del){
            if(temp->rlink==NULL){
                printf("deleted: %d\n",temp->data);
                temp->llink->rlink=NULL;
                free(temp);
            }
            else if(temp==start){
                printf("deleted: %d\n",temp->data);
                start=start->rlink;
                free(temp);
            }
            else{
                printf("deleted: %d\n",temp->data);
                temp->llink->rlink=temp->rlink;
                temp->rlink->llink=temp->llink;
                free(temp);
                return;
```

```
            }
        }
        else{
            printf("element not found\n");
            return;
        }
    }
}

void display(){
    node *temp;
    temp=start;
    do{
        printf("%d\n",temp->data);
        temp=temp->rlink;
    }while(temp!=NULL);
}

int main(){
    int choice;
    do{
        printf("1.create\n2.insert at left\n3.delete
element\n4.display\n5.exit\n");
        scanf("%d",&choice);
        switch (choice)
        {
        case 1:
            create();
            break;
        case 2:
            insert();
            break;
        case 3:
            int del;
            printf("enter the element to delete\n");
            scanf("%d",&del);
            delete(del);
            break;
        case 4:
            display();
            break;
        case 5:
            exit(0);
        default:
            printf("enter a valid chocie\n");
            break;
        }
    }while(choice!=5);
    return 0;
}
```

Output:-

```
1.create
2.insert at left
3.delete element
4.display
5.exit
1
enter the element
10
do you want to enter more elements(1/0)
1
enter the element
20
do you want to enter more elements(1/0)
1
enter the element
30
do you want to enter more elements(1/0)
0
1.create
2.insert at left
3.delete element
4.display
5.exit
2
enter the element
40
1.create
2.insert at left
3.delete element
4.display
5.exit
4
40
10
20
30
1.create
2.insert at left
3.delete element
4.display
5.exit
3
enter the element to delete
40
deleted: 40
1.create
2.insert at left
3.delete element
4.display
5.exit
4
10
20
30
1.create
2.insert at left
3.delete element
4.display
5.exit
5


...Program finished with exit code 0
Press ENTER to exit console.
```