//Program to create, display, delete (linked lists)

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

  void create();
  void display();
  void deletefront();
  void delete_last();
  void delete_at_node(int);

 Struct node
 {
     int data;
     struct node *next;
 };
 struct node *head = NULL;
 int main()
 {
      int choice, ele;
      do
      {
       printf("1.Create\n 2.Display\n 3.Delete first\n 4. Delete from end\n
                5.Delete particularelement \n 6. Exit \n");
       printf(" enter your choice\n");
       scanf("%d", &choice);
       switch(choice)
       {
          case 1 : create();
                   break;
          case 2 : display();
                   break;
           case 3: delete_front();
                   break;
            case4: delete_last();
                   break;
            case 5: printf("enter element to be deleted \n");
                    scanf("%d",&ele);
                    delete_at_node(ele);
                    break;
            default : exit(0);
       }
     } while(choice ==1 ||choice==2|| choice==3||choice==4||choice==5);
   return 0;
 }
```

```
void create()
{
    struct node *newnode, *temp;
    int item;
    newnode = (struct node*)malloc (sizeof (structnode));
    printf(" enter the data ");
    scanf(" %d", &item);
    newnode ->data=item;
    if (head==NULL)
      {
          newnode ->next= NULL;
          head=newnode;
          printf("Nodecreated\n");
      }
      else
        {
          temp=head;
          while (temp->next != NULL)
            {
              temp = temp->next;
            }
          temp -> next =newnode;
          newnode ->next=NULL;
        }
}
void delete_front()
{
    if (head==NULL)
      {
        printf("empty list\n");
      }
    else
        head =head->next;
}
void delete_last()
{
    struct node *temp;
    if (head==NULL)
          printf("list empty\n");
    else {
          temp=head;
          while(temp->next -> next != NULL)
              {
                temp =temp->next;
              }
          temp->next = NULL;
         }
}
```

```
void  delete_at_node(int ele)
{
    Struct node *temp, *del = NULL;
     if (head = NULL)
       { printf(" list empty\n");
         return;
       }
     temp= head;
      if (head->data = = ele)
          {
             head = head->next;
             return;
          }
       while (temp->next != NULL)
       {
          if(temp-> next-> data== ele)
           {
              del = temp -> next;
               if (del->next = =NULL)
                  temp->next=NULL;
               else
                  temp->next= del->next;
           }
          else
            temp = temp->next;
       }
      if( del ==NULL)
      { printf(" Element not found \n");
         return;
      }
}


void display()
{
     struct node *ptr = NULL;
      ptr= head;
      if (ptr ==NULL)
          printf("List empty \n");
       else {
           while (ptr != NULL)
            {
              printf(" %d ", ptr->data);
                ptr= ptr-> next;
            }
       }
       printf("\n");
}
```