```
// Singly linear linked list program...(S.L.L.L)....


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

struct Node
{
   int ele;
   struct Node *next;
};

struct Node *first;

int insert_node()
{
   struct Node *nn,*temp;
   int ch2,sele;

   nn=(struct Node*)malloc(sizeof(struct Node));

   printf("Enter Element for New Node :\n");
   scanf("%d",&nn->ele);

   if(first==NULL)
   {
      nn->next=NULL;
      first=nn;
      printf("list created\n");
   }
   else
   {
      printf("At which position you want to add New Node :\n");

      printf("1 -At FIRST POSITION\n");
      printf("2 -At LAST POSITION\n");
      printf("3 -At SPECIFIC POSITION\n");

      printf("Provide your choice :\n");
      scanf("%d",&ch2);

      switch(ch2)
```

```
{
   case 1 :// inserting NEW NODE at FIRST POSITION.

          nn->next=first;
          first=nn;
          printf("NEW NODE placed at FIRST POSITION");
          break;

   case 2 :// inserting NEW NODE at LAST POSITION.

          nn->next=NULL;
          temp=first;
          while(temp->next !=NULL)
          {
             temp=temp->next;
          }//At the end TEMP points to LAST NODE.

          temp->next=nn;
          printf("NEW NODE placed at LAST POSITION\n");

          break;

   case 3 :// inserting NEW NODE at SPECIFIC POSITION.

         printf("Enter Element of that Node\n");
         scanf("%d",&sele);

         temp=first;
         while(temp->ele !=sele && temp !=NULL)
         {
           temp=temp->next;
         }//At the end TEMP points to Selected Node.

         if(temp==NULL)
         {
           printf("No such Node found, NEW NODE Dropped\n");
         }
         else
         {
           nn->next=temp->next;
           temp->next=nn;
           printf("New Node placed at Specific Position\n");
         }
```

```
                break;




        }
    }

}

int remove_node()
{
    struct Node *temp,*temp2;
    int ch3,sele;

    if(first==NULL)
    {
        printf("List UNDERFLOW \n");
    }
    else
    {
        if(first->next==NULL)
        {
            temp=first;
            first=NULL;
            free(temp);
            printf("There was only 1 node, it is now Removed\n");
        }
        else
        {
            printf("which Node you want to remove :\n");
            printf("1 -FIRST NODE\n");
            printf("2 -LAST NODE\n");
            printf("3 -SPECIFIC NODE\n");

            printf("Provide your choice :\n");
            scanf("%d",&ch3);

            switch(ch3)
            {
                case 1 :// removing FIRST NODE.

                        temp=first;
```

```
        first=first->next;
        free(temp);
        printf("First Node Removed\n");
        break;

case 2 :// removing LAST NODE.

      temp=first;
      while(temp->next !=NULL)
      {
        temp=temp->next;
      }//At the end TEMP points to last node.

      temp2=first;
      while(temp2->next !=temp)
      {
       temp2=temp2->next;
      } //At the end TEMP2 points to second last node.

      temp2->next=NULL;
      free(temp);
      printf("Last Node Removed\n");
      break;

case 3 :// removing SPECIFIC NODE.

        printf("Enter Element of that Node,which you want to remove\n");
        scanf("%d",&sele);

        temp=first;
        while(temp->ele !=sele && temp!=NULL)
        {
           temp=temp->next;
        }//At the end TEMP points to Selected Node.

        if(temp==NULL)
        {
           printf("NO such Node Found\n");
        }
        else
        {
           if(first==temp)
           {
```

```
                    first=first->next;
                    free(temp);
                }
                else
                {
                    temp2=first;
                    while(temp2->next !=temp)
                    {
                      temp2=temp2->next;
                    }//At the end TEMP2 points to Prevoius Node of Selected Node.

                     temp2->next=temp->next;
                     free(temp);
                     printf("Specific Node Removed\n");
                }
            }
            break;


        }
    }
  }
}

int display_node()
{
   int ele;
   struct Node *temp;
   if(first==NULL)
   {
      printf("list is not created,Nothing to display\n");
   }
   else
   {
      printf("list contains\n");
      temp=first;
      while(temp!=NULL)
      {
         printf("%d\n",temp->ele);
         temp=temp->next;
      }
   }
}
```

```
int main()
{
   system("cls");
   int ch;

   first=NULL;
   while(1)
   {
      getch();
      system("cls");

     printf("\n which operation do you want to perform\n");
     printf("1 -insert new node\n");
     printf("2 -remove node\n");
     printf("3 -display node\n");
     printf("4 -exit\n");

     printf("Provide your choice: \n");
     scanf("%d",&ch);

      switch(ch)
      {
          case 1 :// inserting new node operation.

                insert_node();
                break;

          case 2 :// removing node operation.

                remove_node();
                break;

          case 3 :// displaying node operation.

                display_node();
                break;

          case 4 :// exit.

                exit(0);

     }
```

```
    }
    getch();
    return 0;


}
```