Linked list

DHRUVA.N
1BM19CS04

```
void creat()
{
  struct node *newnode, *temp;
  int item;
  newnode = (struct node *) malloc (sizeof (struct node));
  printf ("Enter the data:");
  scanf ("%d", &item);
  newnode -> data = item;
  if (head == NULL)
  {
    newnode -> next = NULL;
    head = newnode;
    printf ("Node created \n");
  }
  else
  {
    temp = head;
    while (temp -> next != NULL)
    {
      temp = temp -> next;
    }
    temp -> next = newnode;
    newnode -> next = NULL;
```

DHRUVA.M
1BM19CS041

```
        printf ("Node created \n");
    }
}
void display ()
{
    struct node * ptr = NULL;
    ptr = head;
    if (ptr == NULL)
    {
        printf ("Nothing to print \n");
    }
    else
    {
        while (ptr != NULL)
        {
            printf ("%d", ptr -> data);
            ptr = ptr -> next;
        } printf ("\n");
    } printf ("-----\n");
}
void delfirst (int ele)
{
    struct node * temp, * del = NULL;
```

```
if (head == NULL)
{
    printf("Empty list, Can't delete\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 in list\n");
    return;
}
```

1BM19CS049
DHRUVA.M

```
void insert_front()
{
    struct node *newnode;
    int ele;
    printf("Enter the element : ");
    scanf("%d", &ele);
    newnode = (struct node *) malloc(sizeof(struct node));
    newnode -> data = ele;
    newnode -> next = head;
    head = newnode;
}
```