Trisha Lakhani
1BM19CS214.

Doubly Linked List.

```
Struct node {
    struct node * prev;
     int data;
     Struct node * next;
 };
 struct node * head = NULL;

 void add-at_beginning ()
   {    struct node * ptr = NULL;
        ptr = (struct node *) malloc (sizeof (struct node));
        printf ("enter the node data");
        scanf ("%d", &ptr → data);
        ptr → prev = NULL;
        ptr → next = NULL;
         if (head == NULL)
          {   head = ptr;
          }
         else {
              ptr → next = head
              head → prev = ptr
               head = ptr;
             }
    }
void delete_at-specified loc() {
       struct node *ptr, *temp;
       int val;
        printf (" Enter the value after which the node is
                    deleted");
        scanf ("%d", &val);
        temp = head;
        while ( temp → data != val)
          temp = temp→ next ;
        if ( temp → next == NULL)
           { printf ( "\n Can't delete \n");
           }
        else if ( temp → next → next ==NULL)
         {          temp→ next = NULL;
              printf ( "Node deleted");
         }
        else
         {
           ptr = temp → next;
           temp → next = ptr → next ;
            ptr → next → prev = temp ;
             free (ptr);
             printf ("Node Deleted");
         }
      }

void display () {
          if ( head == NULL) {
             { printf ('list is empty");
             }
          else {
             Struct node * temp = head;
               while ( temp != NULL)
             { printf ("%d\t", temp → data);
                temp = temp → next;
               }
             printf ("\n");
        }
      }.
```