

Aim:

To implement basic operations on singly linked list.

Algorithm:

1. Define structure with **data** and **next**.
2. For insertion → create node and adjust links.
3. For deletion → traverse and free node.
4. For display → traverse and print all nodes.

Code:

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

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

```
struct Node {
```

```
    int data;
```

```
    struct Node* next;
```

```
};
```

```
struct Node* head=NULL;
```

```
void insert(int val) {
```

```
    struct Node* newNode=malloc(sizeof(struct Node));
```

```
    newNode->data=val; newNode->next=head;
```

```
    head=newNode;
```

```
}
```

```
void delete() {  
  
    if (head==NULL) { printf("List empty\n"); return; }  
  
    struct Node* temp=head;  
  
    head=head->next;  
  
    free(temp);  
  
}
```

```
void display() {  
  
    struct Node* temp=head;  
  
    while (temp!=NULL) {  
  
        printf("%d -> ", temp->data);  
  
        temp=temp->next;  
  
    }  
  
    printf("NULL\n");  
  
}
```

```
int main() {  
  
    int choice, val;  
  
    while (1) {  
  
        printf("\n1.Insert 2.Delete 3.Display 4.Exit\nChoice: ");  
  
        scanf("%d",&choice);  
  
        if (choice==1) {  
  
            printf("Enter value: ");  
  
            scanf("%d",&val);
```

```
        insert(val);  
    } else if (choice==2) delete();  
    else if (choice==3) display();  
    else break;  
}  
return 0;  
}
```

Input & Output:

1.Insert 2.Delete 3.Display 4.Exit

Choice: 1

Enter value: 10

Choice: 1

Enter value: 20

Choice: 3

20 -> 10 -> NULL

Choice: 2

Choice: 3

10 -> NULL

Result:

Linked list insertion, deletion, and display operations implemented successfully.