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

typedef struct queue {
        int data;
        struct queue * ptr;
}queue;

queue *front = NULL, *rear = NULL;

queue * createnode() {
        queue * newnode = (queue *)malloc(sizeof(queue));
        printf("Enter the data element : ");
        scanf("%d" , &newnode->data);
        newnode->ptr = NULL;
        return newnode;
}

void insert() {
        if(front == NULL) {
                front = createnode();
                rear = front;
        }
        else {
                queue * newnode = createnode();
                rear->ptr = newnode;
                rear = newnode;
        }
        printf("%d successfully inserted.\n" , rear->data);
}

void delete() {
        if (front == NULL)
                printf("Queue is empty.\n");
        else {
                printf("%d successfully deleted.\n" , front->data);
                queue * temp = front;
                front = front->ptr;
                free(temp);
        }
}

void display() {
        if (front == NULL)
                printf("Queue is empty.\n");
        else {
                queue * strt = front;
                printf("Elements of queue are : ");
                while(strt != NULL) {
                        printf("%d ", strt->data);
                        strt = strt->ptr;
                }
                printf("\n");
        }
}
```

```
void main() {
        int choice , flag = 1 ;
        printf("Queue using Singly Linked List.\n");
        printf("Enter the operation to be performed.\n");
        while(flag) {
                printf("1.Insert \t 2.Delete \t 3.Display \t 4.Exit  :
");
                scanf("%d" , &choice);
                switch(choice) {
                        case 1 : insert();
                                 break;

                        case 2 : delete();
                                 break;

                        case 3 : display();
                                 break;

                        case 4 : flag = 0 ;
                                 break;

                        default : printf("Invalid input.\n");
                                  break;
                }
        }
}
```