Disha.N
1BM19CS051

// Circular queue.

```
#include <stdio.h>
#include <stdlib.h>
# define N 3
int circular_queue[N];
int front = -1, rear = -1;
void insert()
{
    int value;
    if (front == (rear+1) % N)
    printf("The queue is full, insertion is not possible\n");
    else
    {
        if (front == -1)
        front = 0;
        printf("enter the value to be inserted\n");
        scanf("%d", &value);
        rear = (rear+1) % N;
        circular_queue[rear] = value;
    }
}
void delete()
{
    int elem;
    if (front == -1 && rear == -1)
    {
        printf("Queue underflow\n");
        return;
    }
    else if (front == rear)
    {
        elem = circular_queue[front];
        front = (front+1) % N; rear = -1;
    }
    else {
        elem = circular_queue[front];
        front = (front+1) % N;
    }
    printf("Deleted %d\n", elem);
}
```

```
void display()
{
    int i;
    if (front==-1)
    printf("Queue is empty\n");
    else
     {
        printf("Queue is:\n");
        for (i=front; i!=rear; i=(i+1)%N)
        printf("%d\t", circular queue[i]);
        printf("%d", circular queue[i]);
        printf("\n");
      }
}

int main ()
{
    int choice;
    do
      {
        printf("Enter your choice\n");
        printf("1. Insert\n 2. Delete\n 3. Display 4. Exit\n");
        scanf("%d", &choice);
        switch (choice)
        {
            case 1: insert();
                    break;
             case 2: delete();
                     break;
             case 3: display();
                     break;
             case 4: exit(0);
             default : printf("Incorrect choice\n");
        }
    }(while (choice!=4);
    return 0;
}
```