Run | Debug | Stop | Share | Save | { } Beautify | Language C

main.c

```c
#include <stdio.h>
#include <stdlib.h>
#define size 3

int front=-1;
int rear=-1;


void insert(int,int []);
int delete(int []);
void display(int []);
int main()
{
    int ch,queue[size];;
    int item;
    do{
        printf("\n\n 1. Insert to Queue: ");
        printf("\n 2. delete from the Queue: ");
        printf("\n 3. Display Queue\n ");
        printf("\n 4. Exit\n");
        printf("Enter the option :");
        scanf("%d",&ch);
        switch(ch)
        {
            case 1:  printf("Enter the element\n");
                     scanf("%d",&item);
                     insert(item,queue);
                     break;
            case 2: item=delete(queue);
```

Search the web and Windows

21:48
02-01-2021

Run Debug Stop Share Save { } Beautify Language C

main.c

```
            case 2: item=delete(queue);
                    if(item==-9999)
                      printf("Queue undreflow\n");
                    else
                    printf("Removed element: %d",item);
                    break;
            case 3: display(queue);
                    break;
            case 4: exit(0);
        }
    } while (ch!=4);
    return 0;
}

void insert(int ele,int queue[])
{
    if ((rear+1)%size==front)
       printf("Queue overflow\n");
    else
    {
      rear=(rear+1)%size;
      queue[rear]=ele;
      if ((rear+1)%size==front)
       printf("Queue is full\n");
      if(front ==-1)
          front=0;
    }
}
int delete(int queue[])
```

Search the web and Windows

21:48
02-01-2021

Run Debug Stop Share Save { } Beautify Language C

main.c

```c
int delete(int queue[])
{
    int item;
    if(front == -1)
        return -9999;
    else
    {
        item=queue[front];

        if(front==rear)
        {
            front=-1;
            rear=-1;
        }
        else{front=(front+1)%size;}
        return item;
    }

}

void display(int queue[])
{
    int i;
    if((front==-1)&& (rear==-1))
        printf("Queue is empty\n");
    else
    { if(front<=rear)
       for(i=front;i<=rear;i++)
           printf("%d", queue[i]);
```

Search the web and Windows

21:49
02-01-2021

Run | Debug | Stop | Share | Save | { } Beautify | Language C

main.c

```
        item=queue[front];

        if(front==rear)
        {
            front=-1;
            rear=-1;
        }
        else{front=(front+1)%size;}
        return item;
    }

}

void display(int queue[])
{
    int i;
    if((front==-1)&& (rear==-1))
        printf("Queue is empty\n");
    else
    { if(front<=rear)
       for(i=front;i<=rear;i++)
           printf("%d", queue[i]);
      else
      { for(i=front;i<size;i++)
           printf("%d", queue[i]);
        for(i=0;i<=rear;i++)
           printf("%d", queue[i]);
      }
    }
}
```

input

```
 1. Insert to Queue:
 2. delete from the Queue:
 3. Display Queue
 4. Exit
Enter the option :1
Enter the element
10


 1. Insert to Queue:
 2. delete from the Queue:
 3. Display Queue
 4. Exit
Enter the option :1
Enter the element
20


 1. Insert to Queue:
 2. delete from the Queue:
 3. Display Queue
 4. Exit
Enter the option :1
Enter the element
30
Queue is full
```

input

```
30
Queue is full


 1. Insert to Queue:
 2. delete from the Queue:
 3. Display Queue
 4. Exit
Enter the option :3
102030

 1. Insert to Queue:
 2. delete from the Queue:
 3. Display Queue
 4. Exit
Enter the option :2
Removed element: 10

 1. Insert to Queue:
 2. delete from the Queue:
 3. Display Queue
 4. Exit
Enter the option :2
Removed element: 20

 1. Insert to Queue:
 2. delete from the Queue:
 3. Display Queue
```

input

```
 1. Insert to Queue:
 2. delete from the Queue:
 3. Display Queue
 4. Exit
Enter the option :2
Removed element: 20

 1. Insert to Queue:
 2. delete from the Queue:
 3. Display Queue
 4. Exit
Enter the option :2
Removed element: 30

 1. Insert to Queue:
 2. delete from the Queue:
 3. Display Queue
 4. Exit
Enter the option :2
Queue undreflow


 1. Insert to Queue:
 2. delete from the Queue:
 3. Display Queue
 4. Exit
Enter the option :3
```

input

```
Enter the option :2
Removed element: 30

 1. Insert to Queue:
 2. delete from the Queue:
 3. Display Queue
 4. Exit
Enter the option :2
Queue undreflow


 1. Insert to Queue:
 2. delete from the Queue:
 3. Display Queue
 4. Exit
Enter the option :3
Queue is empty


 1. Insert to Queue:
 2. delete from the Queue:
 3. Display Queue
 4. Exit
Enter the option :4


...Program finished with exit code 0
Press ENTER to exit console.
```