

Debugger

+

https://www.onlinegdb.com/edit/ryq1Rb4vD110%...🔒📌

▶ Run

⌂ Debug

■ Stop

↗ Share

💾 Save

{ } Beautify

⬇

Language C

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#define QUE_SIZE 3
int item, front=0, rear=-1, q[10];

void insertrear()
{
    if(rear==QUE_SIZE-1)
    {
        printf("Queue overflow\n");
        return;
    }
    rear=rear+1;
    q[rear]=item;
}

int deletefront(){
    if(front>rear)
    {
        front=0;
        rear=-1;
        return -1;
    }
    return q[front++];
}

void displayQ()
{
    int i;
    if(front>rear){
        printf("Queue is empty");
        return;
    }
}
```

input

Where to search

🔍

📁

🌐

📁

📧3

⚡

⚙️

🦊

?

^

🔌

📶

🔊

ENG

e Debugger X

+

https://www.onlinegdb.com/edit/ryq1Rb4vD

...

▶ Run

⏮ Debug

■ Stop

↻ Share

💾 Save

{ } Beautify

⬇ Download

Language C

```
if(front>rear){
    printf("Queue is empty");
    return;
}
else
printf("contents of queue\n");
for(i=front;i<=rear;i++){
    printf("%d\n",q[i]);
}

main()

int choice;
for(;;){
    printf("\n 1:insert rear\n 2: delete front\n 3:display\n 4:exit\n");
    printf("Enter the choice\n");
    scanf("%d",&choice);
    switch(choice){
        case 1:printf("Enter the item to be inserted\n");
        scanf("%d",&item);
        insertrear();
        break;
        case 2:item=deletefront();
        if(item==-1)
            printf("queue is empty\n");
        else
            printf("item deleted=%d\n",item);
        break;
        case 3:displayQ();
        break;
        default:exit(0);
    }
}
```

input

where to search

ENG

rear  
e front  
7

choice

item to be inserted

rear  
e front  
7

choice

item to be inserted

rear  
e front  
7

choice

item to be inserted

rear



