//Program to implement Queue Using Array – Visakh Bobby – S3R2 – ROLL No : 34

#include<stdio.h>

#include<stdlib.h>

int F=-1,R=-1; //insertion using rear

int N;

int Q[100];

void QInsert(int item)

{

//Insertion from the Rear End

  if(R>=N-1)

    printf("Queue Overflow , no element can be inserted\n");

  else

  {

    R=R+1;

   Q[R] = item;

    printf("Element %d is inserted to Queue\n",item);

  }

  if(F==-1)

    F=0;

}

void QDel()

{

  //Deletion From The Front End.

  if(F==-1)

    printf("Queue Underflow , No Element Is Present \n");

  else

  {

    printf("Element %d is deleted from Queue\n",Q[F]);

    if(F==R) //both the F & R at the same position , only one element is there in the array

    F=R=-1;

    else

    F=F+1;

  }

}

void QPrint()

{

if(F==-1)

printf("Queue Underflow , No element present inside Queue\n");

  else

  {

  printf("The Queue Elements:\n");

  for(int i=F;i<=R;i++)

    printf("%d\t",Q[i]);

  }

}

void main()

{

int Item;//element to be inserted

int ch=1;

printf("Enter the Maximum Number Of Elements\n");

scanf("%d",&N);

while(ch>=1 && ch<=3)

{

printf("Enter your choice:\n");

printf("1. Queue Insert\n");

printf("2. Queue Delete\n");

printf("3. Queue Display\n");

printf("4. Exit\n");

scanf("%d" , &ch);

   switch(ch)

   {

     case 1 : printf("Enter the item to be inserted\n");

     scanf("%d" , &Item);

         QInsert(Item);

         break;

     case 2 : QDel();

     break;

     case 3 : QPrint();

     break;

     case 4 : exit(0);

     default : printf("Invalid choice\n");

   }

}

} //end of void main

**Output:**





