C: > Users > Dell > Desktop > Go > C exp2dsa.c > display()

```c
// Implementation of Queue using Array
#include<stdio.h>
int Q[100], FRONT = -1, REAR = -1, i, n, x, choice;
void insert();
void delete();
void display();

void main() {
    printf("Welcome to Implementation of Queue using Array !");
    printf("\n Enter the size of Queue (Maximum size = 100): ");
    scanf("%d",&n);
    do {
        printf("\n Queue Operation available: ");
        printf("\n \t1. Insert \t2. Delete \t3. Display \t4. Exit ");
        printf("\n Enter your choice: ");
        scanf("%d",&choice);
        switch (choice) {
            case 1:
            insert();
            break;
            case 2:
            delete();
            break;
            case 3:
            display();
            break;
            case 4:
            printf("\nProgram Finished !");
            break;
            default:
            printf("\n Please enter a valid choice (1, 2, 3, 4)");
            break;

        }

    } while(choice != 4);
}
```

```c
// Function to insert element
void insert() {
    if (REAR >= n - 1) {
        printf("\n Queue Overflow");
    }
    else {
        printf("\n Enter the element to insert: ");
        scanf("%d",&x);
        REAR++;
        Q[REAR] = x;
        if (FRONT == -1) {
            FRONT = 0;
        }
    }
}
// Function to delete element
void delete() {
    if (FRONT == -1) {
        printf("\n Queue is underflow");
    }
    else {
        printf("\n The deleted element is : %d", Q[FRONT]);
        if (FRONT == REAR) {
            FRONT = REAR = - 1;
        }
        else {
            FRONT++;
        }
    }
}
```

```c
// Function to display queue
void display() {
    if (REAR < 0) {
        printf("\n Queue is empty");
    }
    else {
        printf("\n The elements in the queue are: \n");
        for (i = FRONT; i < n; i++) {
            printf(" %d \n",Q[i]);
        }

    }
}
```

```
Welcome to Implementation of Queue using Array !
 Enter the size of Queue (Maximum size = 100): 3

 Queue Operation available:
       1. Insert       2. Delete       3. Display      4. Exit
 Enter your choice: 1

 Enter the element to insert: 1

 Queue Operation available:
       1. Insert       2. Delete       3. Display      4. Exit
 Enter your choice: 1

 Enter the element to insert: 2

 Queue Operation available:
       1. Insert       2. Delete       3. Display      4. Exit
 Enter your choice: 1

 Enter the element to insert: 3

 Queue Operation available:
       1. Insert       2. Delete       3. Display      4. Exit
 Enter your choice: 3

 The elements in the queue are:
 1
 2
 3
```

```
 Queue Operation available:
       1. Insert       2. Delete       3. Display      4. Exit
 Enter your choice: 2

 The deleted element is : 1
 Queue Operation available:
       1. Insert       2. Delete       3. Display      4. Exit
 Enter your choice: 2

 The deleted element is : 2
 Queue Operation available:
       1. Insert       2. Delete       3. Display      4. Exit
 Enter your choice: 3

 The elements in the queue are:
 3

 Queue Operation available:
       1. Insert       2. Delete       3. Display      4. Exit
 Enter your choice: 4

Program Finished !
```