

Kuyruk işlemleri 1

```
//50 Elemanlı Bir sayısal dizi oluşturuyoruz
static int[] kuyruk = new int[50];
//Başlangıç front ve rear değerlerini belirliyoruz.
static int front = 0;
static int rear = -1;

//Kuyruk için yeniden sıralama yapıp verileri sıfırlıyoruz.
static void kuyrukyenile()
{
    for (int i = front; i <=rear ; i++)
    {
        kuyruk[i-front] = kuyruk[i];
    }
    rear = front;
    front = 0;
}

//Kuyruğa eleman ekliyoruz
static void ekle(int data)
{
    //sonraki eklenenin sona gitmesi için her eklemede rear değerini arttırıyoruz
    rear++;
    kuyruk[rear] = data;
}
```

```
static int elemanal()
{
    //kuyruktaki ilk elemanı okuyoruz.
    int data = kuyruk[front];
    //ilk değeri arttırıyoruz sonraki okumada sıradaki veri okunacak.
    front++;

    //Eleman sayısını kontrol ediyoruz 0 olduysa rear ve front sıfırlıyoruz.
    if (elemansayisi() == 0)
    {
        front = 0;
        rear = -1;
    }
    //Eleman sayısı 1 ise kuyruğun son elamanı ile ilk elemanı aynı hale getiriyoruz.
    if (elemansayisi() == 1)
    {
        kuyruk[0] = kuyruk[front];
        front = 0;
        rear = 0;
    }
    return data;
}

//Eleman sayısını alıyoruz bunun için son kayıt - ilk kayıt + 1 formülü
static int elemansayisi()
{
    return rear-front+1;
}

static void Main(string[] args)
{
    //listenin içine 20 eleman dolduruyoruz
    for (int i = 0; i < 20; i++)
    {
        ekle(i);
    }

    //Doldurulan bu 20 elemanı ekrana yazdırıyoruz.
    for (int i = 0; i < 20; i++)
    {
        Console.WriteLine(elemanal());
    }

    Console.ReadLine();
}
```