

List

Kaynak: <https://gelecegiyazanlar.turkcell.com.tr/konu/android/egitim/android-101/list>

Bu Interface kullanıcılara sıralanmış bir dizi hazırlama olanağı sunar. Oluşturulan diziler istenilen sınıfta obje saklayabildiği gibi kapasiteleri de önceden belirlenmek zorunda değildir. Bununla beraber bu Interface'i kullanan sınıflar aşağıdaki metodlara sahiptir;

- **add** (E e): Bir objeyi listeye ekler.
- **add** (int index, E e): Bir objeyi listede istenen dizine (index) ekler.
- **clear**: Bütün elemanları siler.
- **contains**: Bir objenin dizi içerisinde olup olmadığını kontrol eder.
- **get** (int index): Dizi içerisinde belirli bir dizindeki objeyi verir.
- **remove** (E e): Belli bir objeyi siler.
- **remove** (int i): Belli bir dizindeki objeyi siler ve dizini günceller.
- **size**: O anda dizide kaç adet eleman olduğunu söyler (add metodu ile eklenmiş).
- **subList** (int from, int to): İki dizin arasındaki elemanlardan yeni bir dizi oluşturur.
- **set** (int index, E element): Belli bir dizindeki objeyi yenisiyle değiştirir.

Gördüğünüz gibi List Interface'i veriyi dizi içerisinde saklama konusunda bize birçok kolaylık sağlar. Şimdi bu Interface'i kullanan ArrayList adlı sınıfını çeşitli örneklerle anlatalım;

```
List<String> arrList = new ArrayList<String>();
arrList.add("Osman");
arrList.add("Ayşe");
```

Yukarıdaki örnekte yeni bir dizi oluşturuyoruz ve bu diziyi **add** metodunu kullanarak çeşitli elemanlar ekliyoruz.

Bir ArrayList içerisindeki elemanları görüntülemek için foreach döngüsünü aşağıdaki gibi kullanabilirsiniz:

```
for(String str : arrList) {
    System.out.println(str);
}
```

Burada eğer ilk değerin önüne bir eleman ekleyeceksek aşağıdaki metodu kullanmamız gerekir;

```
List<String> arrList = new ArrayList<String>();
arrList.add("Osman");
arrList.add("Ayşe");
arrList.add(0, "Ozan");
```

Bu şekilde "Ozan" objesi "Osman" objesinin önüne geçmiş olur.

Belirli bir obje dizi içerisinde mevcut mu sorgusu aşağıdaki gibi gerçekleşir:

```
List<String> arrList = new ArrayList<String>();
arrList.add("Osman");
arrList.add("Ayşe");
arrList.add(0, "Ozan");
if(arrList.contains("Osman"))
    System.out.println("Osman bulundu");
```

Liste içerisindeki eleman sayısını öğrenmek için **size** metodundan faydalanırız. Aşağıdaki örnekte liste önce **clear** metoduyla boşaltılıyor, ardından **size** metoduyla boşaldığı kontrol ediliyor.

```
List<String> arrList = new ArrayList<String>();
arrList.add("Osman");
arrList.add("Ayşe");
arrList.add(0, "Ozan");
arrList.clear();
if(arrList.size() == 0)
    System.out.println("Liste bos!");
```