



Bölüm 12: Kümeler

Veri Yapıları



Kümeler

- Benzersiz öğelerin koleksiyonunu temsil eden veri yapılarıdır.
- Elemanlar sırasızdır, yani elemanların eklenme sırasını korumaz.
- Bir küme içinde her eleman yalnızca bir kez bulunur (benzersiz elemanlar).
- Elemanlar genellikle küme işleçleri ile işlenir.



Küme İşleçleri

- **Birleştirme (Union):** İki kümenin birleşimini oluşturur.
- **Kesişim (Intersection):** İki kümenin kesişimini alır.
- **Fark (Difference):** İki küme arasındaki farkı hesaplar.
- **Alt küme (Subset):** Bir kümenin başka bir kümenin alt kümesi olup olmadığını kontrol eder.



Kümelerin Gösterimi

- $A = \{1, 2, 3, 4, 5\}$
- $B = \{3, 4, 5, 6, 7\}$
- Birleştirme: $A \cup B = \{1, 2, 3, 4, 5, 6, 7\}$
- Kesişim: $A \cap B = \{3, 4, 5\}$
- Fark: $A - B = \{1, 2\}$
- Alt küme: $A \subseteq B$ (A, B'nin alt kümesi değildir)



java.util.Set Arabirimi Metodları

- `add(E eleman)`: Belirtilen elemanı kümeye ekler.
- `remove(Object eleman)`: Belirtilen elemanı kümeden kaldırır.
- `contains(Object eleman)`: Küme içinde belirtilen elemanın bulunup bulunmadığını kontrol eder.
- `size()`: Kümenin eleman sayısını döndürür.
- `isEmpty()`: Küme boşsa true, doluysa false döndürür.
- `clear()`: Kümedeki tüm elemanları kaldırır.



java.util.Set Örnek Sınıfları

- HashSet: Elemanların sırasız bir şekilde depolandığı bir küme uygulaması.
- LinkedHashSet: Elemanların eklenme sırasına göre depolandığı bir küme uygulaması.
- TreeSet: Elemanların sıralı bir şekilde depolandığı bir küme uygulamasıdır (küme elemanları karşılaştırılabilir olmalıdır).





Kümelerin Yerine Hash Tabloların Kullanılması

- Hash tablolar hızlı aramalar sunar, ancak bu hız beraberinde önemli miktarda gereksiz bellek kullanımını getirir.
- Her ekleme işlemi hem bir anahtar (key) hem de bir değer (value) gerektirir. Bu nedenle, kümeye kıyasla iki kat daha fazla bellek kullanır.
- Yalnızca benzersiz öğeleri takip etmeniz gerektiğinde, değerleri, tekrarlanma sıklıklarını veya benzer bilgileri saklamadığınızda, küme veri yapısı ideal bir tercihtir.



Kümeler Üzerinde Liste/Dizi İşlemleri

- Kümeler ve listeler, temel işlemler açısından farklıdır.
- Liste işlemleri, öğelerin sırası ve karışık türlerle çalışabilirken, küme işlemleri sırasızdır ve genellikle tek türde öğelerle kullanılır.
- Kümelere öğe eklenebilir, ancak öğenin her zaman listenin sonunda olacağı garanti edilmez. Bu nedenle bir kümeden "pop" yapılamaz, bunun yerine öğe kümeden çıkarılabilir.



Kümeler Üzerinde Liste/Dizi İşlemleri

İşlem	Listede Nasıl Çalışır?	Kümede Nasıl Çalışır?
Öge Ekleme	Listenin sonuna ekler.	Kümeye ekler.
Öge Çıkarma	Belirli bir konumdan çıkarır.	Belirli bir öğeyi çıkarır.
Dilimleme	Bir aralıktaki öğeleri alır.	Dilimleme işlemi yoktur.
Öge Sırası	Öğelerin sırası değişebilir.	Kümede öğeler sırasızdır.
Karışık Türler	Farklı türde öğeleri içerebilir.	Genellikle tek tür kullanılır.



SON