

VERİ YAPILARILARI VE ALGORİTMALAR

Collections

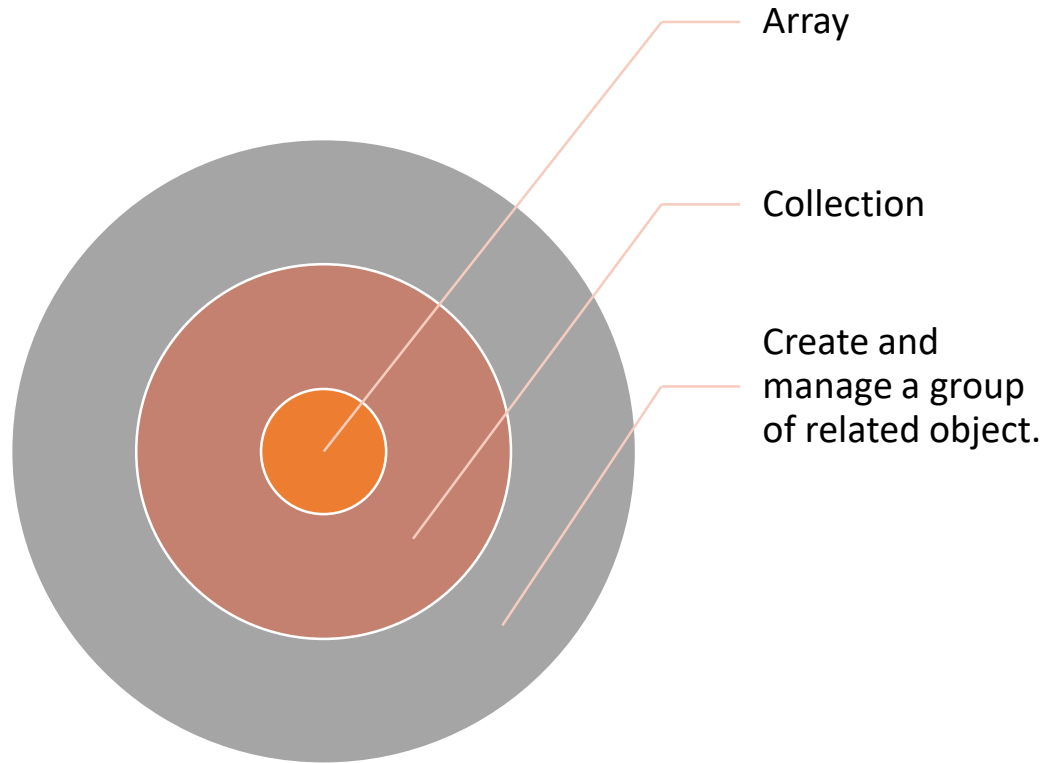
Koleksiyonlar

Collections

- Bir grup nesnenin organize şekilde yönetildiği yapılar **Collections** olarak ifade edilebilir.
- Bu yapılarda ilgili veri yapısına ekleme yapma, araya ekleme yapma, arama, sıralama ya da özel bir takım fonksiyonlar içerir.
- **Collections** ifadeleri **generic** ya da **non-generic** olarak tanımlanabilirler.

Koleksiyonlar

Collections



Aynı türden verilerin
organizasyonunu sağlamak üzere;

- Array
- Collections kullanılabilir.

Koleksiyonlar

Collections

- **Array**, en çok sabit sayıda güçlü biçimde yazılmış nesneler oluşturmak ve bunlarla çalışmak için kullanışlıdır.
- **Collections**, nesne gruplarıyla çalışmak için daha esnek bir yol sağlar.
- Dizilerden farklı olarak, birlikte çalıştığınız nesne grubu, uygulamanın ihtiyaçları değiştikçe dinamik olarak büyüyebilir ve küçülebilir (**growing and shrinking**).

Koleksiyonlar

Collections

- Bazı koleksiyonlarda, koleksiyona eklediğiniz herhangi bir nesneye bir anahtar atayabilirsiniz, böylece anahtarı kullanarak nesneyi hızlı bir şekilde alabilirsiniz.

<TKey,TValue>

Koleksiyonlar

Collections

- **System.Collections**
 - ArrayList, Hashtable, Queue, Stack
- **System.Collections.Generic**
 - Dictionary<TKey, TValue>, List<T>, Queue<T>, SortedList<TKey>, Stack<T>, SortedSet<T>, HashSet<TKey, TValue>
- **System.Collections.Concurrent**
- **System.Collections.Specialized**
- **System.Collections.Immutable**

Ortak Koleksiyon Özellikleri

Common Collections Features

- **Koleksiyonu numaralandırma yeteneği**
 - System.Collection.IEnumerable veya System.Collections.Generic.IEnumerable<T>
- Bir **Enumerator**, koleksiyondaki herhangi bir öğeye taşınabilir bir işaretçi olarak düşünülebilir.
- Bir **foreach** döngüsü **GetEnumerator** metodu kullanarak taşınabilir işaretçi yardımıyla koleksiyondaki öğeler üzerinde dolaşabilir.

Ortak Koleksiyon Özellikleri

Common Collections Features

- **`System.Collections.Generic.IEnumerable<T>`** sorgulanabilir bir tip olarak düşünülebilir ve LINQ ifadeleriyle sorgulanabilir.
- LINQ sorguları, verilere erişim için ortak bir model sağlar.
- LINQ filtreleme, sıralama, gruptama yetenekleriyle veri erişim performansını artırır.

Ortak Koleksiyon Özellikleri

Common Collections Features

- **Koleksiyon içeriğini bir diziye kopyalama yeteneği**
 - Tüm koleksiyonlar **CopyTo** yöntemi kullanılarak bir diziye kopyalanabilir; ancak, yeni dizideki öğelerin sırası, numaralandırıcının onları döndürdüğü sıraya bağlıdır.
- **Capacity and Count Propert**
 - Bir koleksiyonun kapasitesi, içerebileceği öğe sayısıdır. Bir koleksiyon için **Count**, aslında içerdiği öğelerin sayısıdır. Bazı koleksiyonlar **Capacity** veya **Count** veya her ikisini birden gizler.

Ortak Koleksiyon Özellikleri

Common Collections Features

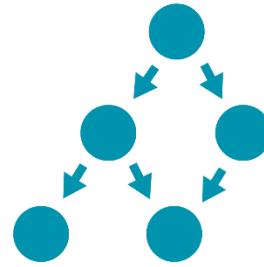
- **Tutarlı bir alt sınır**
 - Bir koleksiyonun alt sınırı, ilk öğesinin dizinidir. `System.Collections` ad alanlarındaki tüm dizinlenmiş koleksiyonların alt sınırı sıfırdır, yani 0 dizinlidir.
- **Birden çok iş parçacığından erişim için senkronizasyon**
 - `System.Collections` ad alanındaki genel olmayan koleksiyon türleri, senkronizasyonla birlikte bazı iş parçacığı güvenliği sağlar; genellikle **`SyncRoot`** ve **`IsSynchronized`** üyeler aracılığıyla gösterilir.

Ortak Koleksiyon Özellikleri

Common Collections Features

Mutable	Amortized	Worst Case	Immutable	Complexity
Stack<T>.Push	O(1)	O(n)	ImmutableStack<T>.Push	O(1)
Queue<T>.Push	O(1)	O(n)	ImmutableQueue<T>.Enqueue	O(1)
List<T>.Add	O(1)	O(n)	ImmutableList<T>.Add	O(logn)
List<T>.Item[Int32]	O(1)	O(1)	ImmutableList<T>.Item[Int32]	O(logn)
List<T>.Enumerator	O(n)	O(n)	ImmutableList<T>.Enumerator	O(n)
HashSet<T>.Add, lookup	O(1)	O(n)	ImmutableHashSet<T>.Add	O(logn)
SortedSet<T>.Add	O(logn)	O(n)	ImmutableSortedSet<T>.Add	O(logn)
Dictionary<T>.Add	O(1)	O(n)	ImmutableDictionary<T>.Add	O(logn)
Dictionary<T>.lookup	O(1)	O(1) – or strictly	ImmutableDictionary<T> lookup	O(logn)
SortedDictionary<T>.Add	O(logn)	O(nlogn)	ImmutableSortedDictionary<T>.Add	O(logn)

<https://docs.microsoft.com/en-us/dotnet/standard/collections/>



Veri Yapıları ve Algoritmalar

ZAFER CÖMERT

Öğretim Üyesi