Veri tiplerini birbirinden ayıran özellik bellekte tutulma biçimleridir.

C# belleği iki bölge olarak kullanır. Bunlardan bir Stack(Yığın) diğeri ise Heap(Öbek) tir.

<https://www.youtube.com/watch?v=xtl6lynnj4Q>

**\*\*\*Değer Tipli Veri\*\*\***

* “int”, “long”, “float”, “double”, “decimal”, “char”, “bool”, “byte”, “short”, “struct”, “enum”
* Değer tipindeki veri tipleri ile bir değişken tanımlandığında değişkenin değeri Stack adı verilen bellek bölgesinde tutulur.
* Bellekte az yer kaplarlar.
* Bu verilere bellekte daha hızlı erişilir.

**\*\*\*Referans Tipli Veri\*\*\***

“string”, “object”, “class”, “interface class”, “array”, “delegate”, “pointer”, “abstract class”,

Bu verilerde hem stack hem de Heap bölgesi kullanılır.

Referans tipindeki veri tipinde bir değişken tanımlandığında ise değişkenin değeri Heap bölgesinde ve Heap bölgesindeki değeri gösteren adres değeri ise Stack bölgesinde tutulur.

Referans tipindeki bir değişkene bellekte yer ayırırken de “new” anahtar kelimesi kullanılır.

Bir ifade referans türleri içeriyorsa nesnenin adresi üzerinden işlem yapılmaktadır

Bir referans türü olan string ile ilgili bu konuda önemli bir istisna vardır. string referans türü olmasına rağmen metotlara geçirilirken değer tipiymiş gibi kopyalanırlar.

**Garbage collector referansı kalmamış nesne ile ilgilidir.**

<https://www.youtube.com/watch?v=_giFJC-07yw>

Boxing 🡪 değer tipinin referans tipine dönüştürülmesi.

Unboxing 🡪 referans tipinin değer tipine dönüştürülmesi.

\*\*\*Stack\*\*\*

değer tipleri, pointer ve adresler saklanır.

Erişim Heap tan daha hızlı.

LIFO(Last in First Out) mantığıyla çalışmaktadır. Bu sebep ile aradan herhangi bir eleman çıkartamazsınız,

birbirleri ile ilişki içerisindedirler.

\*\*\*Heap\*\*\*

referans değerleri saklanır.