3-byte veri tipi : integer ile benzeri bir yapıya sahip ve yalnızca tam sayı değerlerini almaktadır. integer'dan farklı olarak bellekte çok daha az yer kaplamaktadır. Çok büyük olmayan rakamsal veriler için tercih edilebilirler. Değer aralığı 0 - 255 arasındaki değerleri alır. Eksi negatif değer alamaz. Bellek üzerinde kapladığı alan 1 Bayt'tır. Örneğin plaka, not, renk kodları

4-Long Veri Tipi : İnteger ile benzer yapıya sahiptir. Yalnızca rakamsal ve tam sayı değerlerini içinde saklayabilir. integer'dan farklı olarak tanımlanabilecek değer aralığı çok daha büyüktür. İnteger veri tipinde saklanamayacak büyüklükte değerleri long veri tipinde saklayabiliriz.

Bunu karşılığı olarak integer veri tipine göre bellekte daha fazla yer kaplamaktadır. (2 kat daha fazla) bundan dolayı gerekmediği sürece long veri tipi kullanmaktan kaçınılmalıdır.

Değer Aralığı: (-) 9.223.372.036.854.775.808 ile (+) 9.223.372.036.854.775.807 arasındadır. Bellekte 8 byte yer alır.

5-Char veri tipi : Character kelimesinin kısaltılmış halidir. Türkçe anlamı ise Karakterdir. İsminden de anlaşılabileceği üzere içerisinde yalnızca tek bir değer tutar. Bunun dışında özel karakter veya tüm dillerin alfabesinden de herhangi bir harf veya sembol alabilir. Char Veri tipine atanacak değerler her zaman tek tırnak arasında yazılır.

Değer aralığı: tüm karakterler ve rakamlar. Bellekte kapladığı alan 2 byte.

6-Bool Veri Tipi : Matematiksel mantığın kurucusu sayılan Geoge Boole'un adına izafe edilmiştir. Mantıksal veri tipidir. Yalnızca 2 değer alabilir. Programlamada çok çok büyük önem taşır Özellikle programın akışının yönlendirilmesi aşamasında kullanılır.

Değer aralığı : true veya false (doğru yada yanlış) bellekte kullandığı alan 1 byte

7-Double, Float, Decimal Veri Tipi : Sayısal veri tipleridir. Ancak diğer tiplerden farklı kesirli sayıları barındırabilmeleridir.

Örnek olarak KDV hesaplayan bir programı int veri tipini kullanarak yapamayız.

Double : Özel veya aksi bir durum belirtmediğimiz sürece c# dilinde tanımladığımız her kesirli sayı birer "Double" veri tipi olarak algılanır.

Double veri tipinden bir değilken tanımladığımızda sonuna 'D' ya da 'd' harfi koyabiliriz bu bir zorunluluk ya da kural değildir.

Veri tipinin double olduğunu belirtmek için bu yöntem kullanılır. Bellekte 8 byte yer tutar ve maksimum 16 basamak tutabilir.

Float : Kullanım ve yapı bakımından double veri tipine benzemektedir ve aynı özelliklere sahiptir. Aralarındaki tek fark float 7 haneli sayılara kadar izin verir ve bellekte kapladığı alan double veri tipinin yarısı kadardır yani 4 bayt.

Float olarak tanımlanan bir veri tipinin sonuna 'F' veya 'f' harfinin eklenmesi GEREKLİDİR.

Decimal : Ondalık veri tiplerinin en büyüğüdür ve dolayısıyla bellekte en çok yer kaplayandır. Mantığında double ve float gibi kesirli sayıları barındırmak vardır fakat asıl kullanım alanları parasal / finansal işlemlerdir. Bunun sebebi ise decimal veri tipinin kesin sonuçlarla çalışmasıdır. Float ve double veri tipi ondalık sayılar ile işlem yaparken yazdığının veri tipinin son 5'ten küçükse bunu ekrana yazdırmaz. Ama decimal veri tipinde bu durum yoktur. Veriyi net ve tam olarak hesaplar herhangi bir yuvarlama işlemi yapmaz. bu nedenle parasal işlemlerde sıkça tercih edilir.

Bellekte 16bayt yer kaplar ve 29 haneye kadar destekler. Sonuna Büyük M koymak zorunludur.

Double Float ve Decimal arasındaki temel farklar

üç veri tipi arasındaki temel fark double ve float %100 sonuç vermez çünkü yuvarlama işlemi yaparlar. Ancak decimal veri tipi yuvarlama işlemi yapmaz ve kesin sonuç verir. bunun yanı sıra 3 veri tipinin de bellekte kapladığı alanlar birbirinden farklı olmakla beraber destekledikleri maksimum basamak sayısı da farklıdır. Yapacağımız işleme ve tutulacak veriye göre en uygun seçimin yapılması gereklidir.

Örneğin doğa olaylarını hesaplarken kesin sonuçlara ihtiyaç yoktur zaten doğa olayları da kesin olarak tahmin edilemez.

Bu nedenle float ve double uygundur. Ancak kesin sıonuca dayalı işlemler varsa decimal tercih edilir.

byte	1 bayt	0, ..., 255 (tam sayı)	`byte a=5;`
sbyte	1 bayt	-128, ..., 127 (tam sayı)	`sbyte a=5;`
short	2 bayt	-32768, ..., 32767 (tam sayı)	`short a=5;`
ushort	2 bayt	0, ..., 65535 (tam sayı)	`ushort a=5;`
int	4 bayt	-2147483648, ..., 2147483647 (tam sayı)	`int a=5;`
uint	4 bayt	0, ..., 4294967295 (tam sayı)	`uint a=5;`
long	8 bayt	-9223372036854775808, ..., 9223372036854775807 (tam sayı)	`long a=5;`
ulong	8 bayt	0, ..., 18446744073709551615 (tam sayı)	`ulong a=5;`
float	4 bayt	$\pm 1.5*10^{-45}$, ..., $\pm 3.4*10^{38}$ (reel sayı)	`float a=5F;` veya `float a=5f;`
double	8 bayt	$\pm 5.0*10^{-324}$, ..., $\pm 1.7*10^{308}$ (reel sayı)	`double a=5;` veya `double a=5d;` veya `double a=5D;`
decimal	16 bayt	$\pm 1.5*10^{-28}$, ..., $\pm 7.9*10^{28}$ (reel sayı)	`decimal a=5M;` veya `decimal a=5m;`
char	2 bayt	Tek bir karakteri tutar.	`char a='h';`
string	Sınırsız	Metin tutar.	`string a="Ben bir zaman kaybıyım, beni boşver hocam"`