

[Anasayfa](#) / [Standart Modüller](#)

Bellek Modülü

Bu modül sayesinde bellek operasyonları yapılır.

```
Bellek.Yeni( SafRakam Boyut; Bayt Kip = 0 ) İşarettir
```

Bellekte yeni bir alan rezerve eder ve o alana erişmek için, erişim adresini hedef adrese kaydeder. Alanı rezerve edebilmek için bir boyut belirlemek gerekiyor. En ufak boyut 1'dir. Boyut kipe göre değerlendirilir. Kip [BellekKipi](#) değerlerinden olmalıdır.

```
Bellek.Esnet( İşaret kaynakAdres; SafRakam Boyut; Bayt Kip = 0 ) İşarettir
```

Önceden rezerve edilmiş olan bir bellek alanı esnetilir. Yani boyutu kısalabilir, yada genişleyebilir. Parametre olarak kaynak yerine önceki adres girilir ve ikinci parametreye yeni boyut değeri girilir. "x" sonuç olarak yeni adresi alacaktır. Boyut kipe göre değerlendirilir. Kip [BellekKipi](#) değerlerinden olmalıdır.

```
Bellek.Sıfırla( İşaret kaynakAdres )
```

Bu adres mutlaka doğru girilmelidir. Doğru ise, o adres tekrar bellekten kaldırılır. Yanlış veya geçersiz ise, uygulama zorunlu hata vererek sonlandırılır.

```
Bellek.Boşalt( İşaret kaynakAdres )
```

Bu alandaki tüm baytlar sıfırlanır, fakat alanın kendisi kaldırılmaz. Adres yanlış veya geçersiz ise, uygulama zorunlu hata vererek sonlandırılır.

```
Bellek.BaytYaz( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon; Bayt Veri ) Bayttır
```

Bellek alanında istenilen pozisyona belirtilen veri girilir. Bu veri 0 ile 255 arası bir değer olabilir. BaytSonuç başarı durumunda 1, başarısızlık durumunda 0 değerini alır.

```
Bellek.BaytOku( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon ) Bayttır
```

Bellek alanında istenilen pozisyonda bulunan bayt değerini sayısal bir değişken adresine geri verir.

```
Bellek.ÇiftBaytYaz( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon; ÇiftBayt Veri ) Bayttır
```

Bellek alanında istenilen pozisyona belirtilen veri girilir. Bu veri 0 ile 65535 arası bir değer olabilir. Sonuç başarı durumunda 1, başarısızlık durumunda 0 değerini alır.

```
Bellek.ÇiftBaytOku( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon
```

) ÇiftBayttır

Bellek alanında istenilen pozisyonda bulunan çiftbayt değerini sayısal bir değişken adresine geri verir.

```
Bellek.KüçükRakamYaz( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon  
; KüçükRakam Veri ) Bayttır
```

Bellek alanında istenilen pozisyona belirtilen veri girilir. Bu veri -32768 ile +32767 arası bir değer olabilir. Sonuç başarı durumunda 1, başarısızlık durumunda 0 değerini alır.

```
Bellek.KüçükRakamOku( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon  
) KüçükRakamdır
```

Bellek alanında istenilen pozisyonda bulunan küçük rakam değerini sayısal bir değişken adresine geri verir.

```
Bellek.SafRakamYaz( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon;  
SafRakam Veri ) Bayttır
```

Bellek alanında istenilen pozisyona belirtilen veri girilir. Bu veri 0 ile 4294967295 arası bir değer olabilir. Sonuç başarı durumunda 1, başarısızlık durumunda 0 değerini alır.

```
v = Bellek.SafRakamOku( İşaret kaynakAdres; İşaretDeğil SafRakam Pozi  
syon ) SafRakamdır
```

Bellek alanında istenilen pozisyonda bulunan saf rakam değerini sayısal bir değişken adresine geri verir.

```
Bellek.RakamYaz( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon; Ra  
kam Veri ) Bayttır
```

Bellek alanında istenilen pozisyona belirtilen veri girilir. Bu veri -2147483648 ile +2147483647 arası bir değer olabilir. BaytSonuç başarı durumunda 1, başarısızlık durumunda 0 değerini alır.

```
Bellek.RakamOku( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon )  
Rakamdır
```

Bellek alanında istenilen pozisyonda bulunan rakam değerini sayısal bir değişken adresine geri verir.

```
Bellek.EsnekRakamYaz( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon  
; EsnekRakam Veri ) Bayttır
```

Bellek alanında istenilen pozisyona belirtilen veri girilir. Bu veri 32Bit derlemede -2147483648 ile +2147483647 ve 64Bit derlemede -9223372036854775808 ile +9223372036854775807 arası bir değer olabilir. Sonuç başarı durumunda 1, başarısızlık durumunda 0 değerini alır.

```
Bellek.EsnekRakamOku( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon  
) EsnekRakamdır
```

Bellek alanında istenilen pozisyonda bulunan saf rakam değerini sayısal bir değişken adresine geri

verir.

```
Bellek.SafEsnekRakamYaz( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon; SafEsnekRakam Veri ) Bayttır
```

Bellek alanında istenilen pozisyona belirtilen veri girilir. Bu veri 32Bit derlemede 0 ile 4294967295 arası ve 64Bit derlemede 0 ile 18446744073709551615 arası bir değer olabilir. BaytSonuç başarı durumunda 1, başarısızlık durumunda 0 değerini alır.

```
Bellek.SafEsnekRakamOku( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon ) SafEsnekRakamdır
```

Bellek alanında istenilen pozisyonda bulunan saf rakam değerini sayısal bir değişken adresine geri verir.

```
Bellek.GenişRakamYaz( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon; GenişRakam Veri ) Bayttır
```

Bellek alanında istenilen pozisyona belirtilen veri girilir. Bu veri -9223372036854775808 ile +9223372036854775807 arası bir değer olabilir. BaytSonuç başarı durumunda 1, başarısızlık durumunda 0 değerini alır.

```
Bellek.GenişRakamOku( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon ) GenişRakamdır
```

Bellek alanında istenilen pozisyonda bulunan saf rakam değerini sayısal bir değişken adresine geri verir.

```
Bellek.SafGenişRakamYaz( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon; SafGenişRakam Veri ) Bayttır
```

Bellek alanında istenilen pozisyona belirtilen veri girilir. Bu veri 0 ile 18446744073709551615 arası bir değer olabilir. Sonuç başarı durumunda 1, başarısızlık durumunda 0 değerini alır.

```
Bellek.SafGenişRakamOku( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon ) SafGenişRakamdır
```

Bellek alanında istenilen pozisyonda bulunan saf rakam değerini sayısal bir değişken adresine geri verir.

```
Bellek.KüçükVirgölYaz( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon; KüçükVirgöl Veri ) Bayttır
```

Bellek alanında istenilen pozisyona belirtilen veri girilir. Bu veri -1175494e-38 ile +3402823e+38 arası bir değer olabilir. Sonuç başarı durumunda 1, başarısızlık durumunda 0 değerini alır.

```
Bellek.KüçükVirgölOku( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon ) KüçükVirgöldür
```

Bellek alanında istenilen pozisyonda bulunan virgöl değerini virgüllü sayısal bir değişken adresine geri

verir.

```
Bellek.VirgölYaz( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon; V  
irgöl Veri ) Bayttır
```

Bellek alanında istenilen pozisyona belirtilen veri girilir. Bu veri -22250738585072013e-308 ile +17976931348623157e+308 arası bir değer olabilir. Sonuç başarı durumunda 1, başarısızlık durumunda 0 değerini alır.

```
Bellek.VirgölOku( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon )  
Virgüldür
```

Bellek alanında istenilen pozisyonda bulunan virgöl değerini virgüllü sayısal bir değişken adresine geri verir.

```
Bellek.MetinYaz( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon; Me  
tin Veri ) Bayttır
```

Bellek alanında istenilen pozisyona belirtilen veri girilir. Alan metni alacak kadar geniş bir alana sahip olmalıdır, aksi taktirde metin sadece girilebilecek yer kadar girilir, gerisi girilmez. Karakter başına iki bayt gerektirir. Sonuç başarı durumunda 1, başarısızlık durumunda 0 değerini alır.

```
Bellek.MetinOku( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon; Ka  
rakterAdeti = 0 ) Metindir
```

Bellek alanında istenilen pozisyonda bulunan metinsel değeri metinsel bir değişken adresine geri verir. KarakterAdeti = 0 ise, metin sonuna kadar okunur, aksi taktirde belirtilen adet kadar okunur. Maksimum mevcut karakterden fazla okunmaz.

```
Bellek.AsciiMetniYaz( İşaret kaynak; İşaretDeğil SafRakam Pozisyon; As  
ciiMetni Veri ) Bayttır
```

Bellek alanında istenilen pozisyona belirtilen veri girilir. Alan metni alacak kadar geniş bir alana sahip olmalıdır, aksi taktirde metin sadece girilebilecek yer kadar girilir, gerisi girilmez. Karakter başına bir bayt gerektirir. Sonuç başarı durumunda 1, başarısızlık durumunda 0 değerini alır.

```
Bellek.AsciiMetniOku( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon  
; KarakterAdeti = 0 ) AsciiMetnidir
```

Bellek alanında istenilen pozisyonda bulunan metinsel değeri metinsel bir değişken adresine geri verir. KarakterAdeti = 0 ise, metin sonuna kadar okunur, aksi taktirde belirtilen adet kadar okunur. Maksimum mevcut karakterden fazla okunmaz.

```
Bellek.YapıYaz( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon; İşa  
retli Yapı yapıKaynak ) Bayttır
```

Bellek alanına bir yapı taşıını olduğu gibi yazar. Fakat kaynak yapı taşı dizilimsel veriler içeriyor ise, buna [AsciiMetni](#) ve [Metin](#) türleri dahil, bunların sadece işaretleri yazılır, kendileri yazılmaz. Bu durum [Esnek](#), [Yapı](#), [Obje](#) ve [Bileşen](#) türü elemanlar içinde geçerlidir.

```
Bellek.Yapı0ku( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon; Hacim ) Yapıdır
```

Bellek alanında bulunan bir yapı taşını geri verir. Hacim yapının toplam Bayt adetini belirtmelidir. Bunu yapı taşının tanımlamasını girerek elde edebilirsiniz.

```
Bellek.Yapı0ku( İşaret kaynakAdres; İşaretDeğil SafRakam Pozisyon; İşaretli Yapı hedefYapı )
```

Bellek alanında bulunan bir yapı taşını hedef yapı taşına kopyalar. Eğer işlem başarılı olursa geriye 1, aksi taktide 0 verir.

```
Bellek.Kopyala( İşaret hedefAdres; kaynakAdres )
```

Bir bellek alanını diğer bellek alanına kopyalar. Hedef alan yeterli kapasiteye sahip olmalıdır.

```
Bellek.Kopyala( İşaret hedefAdres; kaynakAdres; İşaretDeğil SafRakam Pozisyon = 0 )
```

Bir bellek alanını diğer bellek alanına, pozisyonundan itibaren kopyalar. Hedef alan yeterli kapasiteye sahip olmalıdır.

```
Bellek.Kopyala( İşaret hedefAdres; İşaretDeğil SafRakam hedefPoz; hedefBoyut; İşaret kaynakAdres; İşaretDeğil SafRakam kaynakPoz; kaynakBoyut )
```

Bir bellek alanından bir bölümü diğer bellek alanının bir bölümüne kopyalar. Hedef alan yeterli kapasiteye sahip olmalıdır.

```
Bellek.Kopyala( İşaret kaynakAdres; İşaretDeğil Bayt Kip = VeriTürü ) Dizilimsel Esnektir
```

Bellek alanında bulunan verileri bir dizilime kopyalar. Değerlerin nasıl kopyalanacağını kip belirler. Geçerli kip değerleri **VeriTürü** seçeneklerinden ilk gruptadır ve şöyledir: ascii, bayt, yarıbayt, çiftbayt, küçükkrakam, safküçükkrakam, karakter, rakam, safrakam, esnekrakam, safesnekrakam, genişrakam, safgenişralam, küçükvirgöl, virgöl.

```
Bellek.Oluştur( Dizilimsel Esnek kaynakAdres; Değişken Karakter Kip = 0 ) İşarettir
```

Esnek türlü bir dizilimden yola çıkarak yeni bellek alanı oluşturulur ve dizilimin içerdiği değerler o bellek alanına kopylanır. Değerlerin nasıl kopyalanacağını kip belirler. Geçerli kip değerleri **VeriTürü** seçeneklerinden ilk gruptadır ve şöyledir: ascii, bayt, yarıbayt, çiftbayt, küçükkrakam, safküçükkrakam, karakter, rakam, safrakam, esnekrakam, safesnekrakam, genişrakam, safgenişralam, küçükvirgöl, virgöl.

```
Bellek.Tersle( İşaret kaynakAdres; İşaretDeğil Bayt Kip = 0 ) Bayttır
```

Bellek alanındaki verileri tersine çevirir. Başta bulunan baytlar sona, sondaki başa geçer. Verilerin

çevirimi kipe göre değerlendirilir. Kip **BellekKipi** değerlerinden olmalıdır. Başarı durumunda 1, aksi taktirde 0 geri verir.

From:

<https://www.basit.web.tr/dokuwiki/> - **Basit Yazılım Dili Dokümantasyonu**

Permanent link:

<https://www.basit.web.tr/dokuwiki/doku.php?id=bellek>

Last update: **28.03.2021 18:33**

