

JAVASCRIPT'TE BITSEL ATAMA OPERATÖRLERİ

Bitisel atama operatörleri bir değişkene bit düzeyli ve işlemli değer / veri atamak için kullanılır. İşlemli atama, atama sırasında işlem gerçekleştirilerek sonucun atamasını sağlar. Bitisel atama operatörlerini inceleyecek olur isek;

Operatör	Tanım	Açıklama	Örnek		
<<=	Bitisel sola kaydırma ataması	Değişkene değer / veri atamak için kullanılır. Değişkene değer atanırken, değişkenin kendi sayısal değerinin bitisel şablonu üzerinde, diğer değişkenin değeri kadar sola kaydırma işlemi yapılır. İşaret biti değişmez. Dolayısı ile negatif bir sayıda olabilir. Sola kaydırma işleminden sonra ikinci değişkenin değeri ile belirlenmiş değer kadar ilk değişken değerinin sağına 0 ekler ve işlem sonucunda oluşan değer değişkene atanmış olur.	a	00010110	22
			b	3	
			İşlem	a <<= b	
			İşlem	a = a << b	
			Sonuç	10110000	176
>>=	Bitisel sağa kaydırma ataması	Değişkene değer / veri atamak için kullanılır. Değişkene değer atanırken, değişkenin kendi sayısal değerinin bitisel şablonu üzerinde, diğer değişkenin değeri kadar sağa kaydırma işlemi yapılır. İşaret biti değişmez. Dolayısı ile negatif bir sayıda olabilir. Sağa kaydırma işleminden sonra ikinci değişkenin değeri ile belirlenmiş değer kadar ilk değişken değerinin soluna 0 ekler ve işlem sonucunda oluşan değer değişkene atanmış olur.	a	00010110	22
			b	3	
			İşlem	a >>= b	
			İşlem	a = a >> b	
			Sonuç	00000010	10
>>>=	Bitisel işaretsiz sağa kaydırma ataması	Değişkene değer / veri atamak için kullanılır. Değişkene değer atanırken, değişkenin kendi sayısal değerinin bitisel şablonu üzerinde, diğer değişkenin değeri kadar sağa kaydırma işlemi yapılır. İşaret biti 0 olur. Dolayısı ile negatif bir sayı alamaz. Sağa kaydırma işleminden sonra ikinci değişkenin değeri ile belirlenmiş değer kadar ilk değişken değerinin soluna 0 ekler ve işlem sonucunda oluşan değer değişkene atanmış olur.	a	00010110	22
			b	3	
			İşlem	a >>= b	
			İşlem	a = a >> b	
			Sonuç	00000010	10
&=	Bitisel ve ataması	Değişkene değer / veri atamak için kullanılır. Değişkene değer atanırken, değişkenin kendi sayısal değerinin bitisel şablonu ile, diğer değişkenin sayısal değerinin bitisel şablonu karşılaştırılarak işlem yapılır. Bitisel ve ataması işlemi karşılaştırılmasında her bit için karşılaştırma ayrı ayrı olacak şekilde sağlanır. 1 ile 0 değeri karşılaştırılmasında 0, 1 ile 1 değeri karşılaştırılmasında 1, 0 ve 0 değeri karşılaştırılmasında 0 değerleri döner. İşlem sonucunda oluşan değer değişkene atanmış olur.	a	00010110	22
			b	00000011	3
			İşlem	a &= b	
			İşlem	a = a & b	
			Sonuç	00000010	2
=	Bitisel veya ataması	Değişkene değer / veri atamak için kullanılır. Değişkene değer atanırken, değişkenin kendi sayısal değerinin bitisel şablonu ile, diğer değişkenin sayısal değerinin bitisel şablonu karşılaştırılarak işlem yapılır. Bitisel veya ataması işlemi karşılaştırılmasında her bit için karşılaştırma ayrı ayrı olacak şekilde sağlanır. 1 ile 0 değeri karşılaştırılmasında 1, 1 ile 1 değeri karşılaştırılmasında 1, 0 ve 0 değeri karşılaştırılmasında 0 değerleri döner. İşlem sonucunda oluşan değer değişkene atanmış olur.	a	00010110	22
			b	00000011	3
			İşlem	a = b	
			İşlem	a = a b	
			Sonuç	00010111	23

Operatör	Tanım	Açıklama	Örnek		
^=	Bitsel ayrıcalıklı veya ataması	Değişkene değer / veri atamak için kullanılır. Değişkene değer atanırken, değişkenin kendi sayısal değerinin bitsel şablonu ile, diğer değişkenin sayısal değerinin bitsel şablonu karşılaştırılarak işlem yapılır. Bitsel ayrıcalıklı veya ataması işlemi karşılaştırılmasında her bit için karşılaştırma ayrı ayrı olacak şekilde sağlanır. 1 ile 0 değeri karşılaştırılmasında 1, 1 ile 1 değeri karşılaştırılmasında 0, 0 ve 0 değeri karşılaştırılmasında 0 değerleri döner. İşlem sonucunda oluşan değer değişkene atanmış olur.	a	00010110	22
			b	00000011	3
			İşlem	a ^= b	
			İşlem	a = a ^ b	
			Sonuç	00010101	21