

PHP'DE BIT İŞLEM OPERATÖRLERİ

Bit işlem operatörleri, bitwise (*bit düzeyi*) alanında, boolean (*mantıksal*) işlemler gerçekleştirmek için kullanılır. Bit işlem operatörleri iki operand alır ve her iki operand birbirleri ile karşılaştırılır. Bit işlem operatörlerinin operandlarında muhakkak numerik (*sayı*) veri türünde içerikler olmalıdır. Bit işlem operatörlerini inceleyecek olur isek;

| Operatör | Tanım | Açıklama | Örnek | | |
|----------|-------------------------|--|-------|----------------------|---|
| & | Bitsel ve | Operandların sayısal değerlerinin bitsel şablonları üzerinde karşılıklı olarak bitsel karşılaştırma için kullanılır. Bitsel ve 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. | \$a | 00010110 | 22 |
| | | | \$b | 00000011 | 3 |
| | | | İşlem | \$a & \$b | |
| | | | Sonuç | 00000010 | 2 |
| | Bitsel veya | Operandların sayısal değerlerinin bitsel şablonları üzerinde karşılıklı olarak bitsel karşılaştırma için kullanılır. Bitsel veya 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. | \$a | 00010110 | 22 |
| | | | \$b | 00000011 | 3 |
| | | | İşlem | \$a \$b | |
| | | | Sonuç | 00010111 | 23 |
| ^ | Bitsel ayrıcalıklı veya | Operandların sayısal değerlerinin bitsel şablonları üzerinde karşılıklı olarak bitsel karşılaştırma için kullanılır. Bitsel ayrıcalıklı veya 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. | \$a | 00010110 | 22 |
| | | | \$b | 00000011 | 3 |
| | | | İşlem | \$a ^ \$b | |
| | | | Sonuç | 00010101 | 21 |
| ~ | Bitsel değil | Eklendiği operandın sayısal değerinin bitsel şablonu üzerinde bitsel değerinin mantıksal olarak ters değerini döndürür. Bitsel değil işlemi, her bit için ayrı ayrı olacak şekilde işlem yapar. 1 değeri karşılaştırılmasında 0, 0 değeri karşılaştırılmasında 1 değerleri döner. | \$a | 00010110 | 22 |
| | | | İşlem | ~\$a | |
| | | | Sonuç | 11101001 | 233 |
| | | | << | Bitsel sola kaydırma | İlk operandın sayısal değerinin bitsel şablonu üzerinde ikinci operand ile belirlenecek değer kadar sola kaydırma işlemi yapar. İşaret biti değişmez. Dolayısı ile negatif bir sayıda olabilir. Sola kaydırma işleminden sonra ikinci operand ile belirlenmiş değer kadar ilk operand değerinin sağına 0 ekler. |
| \$b | 3 | | | | |
| İşlem | \$a << \$b | | | | |
| Sonuç | 10110000 | 176 | | | |
| >> | Bitsel sağa kaydırma | İlk operandın sayısal değerinin bitsel şablonu üzerinde ikinci operand ile belirlenecek değer kadar sağa kaydırma işlemi yapar. İşaret biti değişmez. Dolayısı ile negatif bir sayıda olabilir. Sağa kaydırma işleminden sonra ikinci operand ile belirlenmiş değer kadar ilk operand değerinin soluna 0 ekler. | \$a | 00010110 | 22 |
| | | | \$b | 3 | |
| | | | İşlem | \$a >> \$b | |
| | | | Sonuç | 00000010 | 2 |