

eram um valor booleano (1 = true, 0 = false). Os outros três bit

Os operadores são usados ?? para mudar os bits para a esquerda e para a direita. Esses operadores não são comumente usados na programação JavaScript, e se você não estiver familiarizado com a representação binária de números inteiros, incluindo os dois

Complemento Representação de números inteiros negativos, você provavelmente pode pular esta seção.

Os operadores bitwee esperam operando inteiros e se comportar

Os valores foram representados como números inteiros de 32 bits em vez de flutuação de 64 bits valores pontuais. Esses operadores convertem seus operando em números, se necessário e depois coagir os valores numéricos a números inteiros de 32 bits por

soltando qualquer parte fracionária e quaisquer bits além do 32º. A mudança

Os operadores exigem um operando do lado direito entre 0 e 31. Depois convertendo este operando em um número inteiro de 32 bits não assinado, eles abandonam qualquer Bits além do 5º, que produz um número no intervalo apropriado.

Surpreendentemente, nan, infinito e -infinity todos se convertem para 0

Quando usado como operando desses operadores bit-bit.

Todos esses operadores bit new, exceto >>> podem ser usados ?? com regular operando numéricos ou com operando bigint (consulte §3.2.5).

Bitwise e (&)

O Operador executa um booleano e operação em cada bit de

seus argumentos inteiros. Um pouco está definido no resultado apenas se o

O bit correspondente é definido nos dois operandos. Por exemplo, 0x1234 & 0x00FF Avalia para 0x0034.

Bit nessa ou (|)

O | O operador executa um booleano ou operação em cada bit de seu