**[Формат 32-битного целого числа со знаком](http://learn.javascript.ru/bitwise-operators" \l "signed-format)**

Побитовые операторы в JavaScript работают с 32-битными целыми числами в их двоичном представлении.

Это представление называется «32-битное целое со знаком, старшим битом слева и дополнением до двойки».

Примеры представления чисел в двоичной системе:

a = 0; // 00000000000000000000000000000000

a = 1; // 00000000000000000000000000000001

a = 2; // 00000000000000000000000000000010

a = 3; // 00000000000000000000000000000011

a = 255;// 00000000000000000000000011111111

**[Список операторов](http://learn.javascript.ru/bitwise-operators" \l "%D1%81%D0%BF%D0%B8%D1%81%D0%BE%D0%BA-%D0%BE%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%BE%D1%80%D0%BE%D0%B2)**

В следующей таблице перечислены все побитовые операторы. Далее операторы разобраны более подробно.

|  |  |  |
| --- | --- | --- |
| **Оператор** | **Использование** | **Описание** |
| Побитовое И (AND) | a & b | Ставит 1 на бит результата, для которого соответствующие биты операндов равны 1. |
| Побитовое ИЛИ (OR) | a | b | Ставит 1 на бит результата, для которого хотя бы один из соответствующих битов операндов равен 1. |
| Побитовое исключающее ИЛИ (XOR) | a ^ b | Ставит 1 на бит результата, для которого только один из соответствующих битов операндов равен 1 (но не оба). |
| Побитовое НЕ (NOT) | ~a | Заменяет каждый бит операнда на противоположный. |
| Левый сдвиг | `a << b` | Сдвигает двоичное представление a на b битов влево, добавляя справа нули. |
| Правый сдвиг, переносящий знак | `a >> b` | Сдвигает двоичное представление a на b битов вправо, отбрасывая сдвигаемые биты. |
| Правый сдвиг с заполнением нулями | `a >>> b` | Сдвигает двоичное представление a на b битов вправо, отбрасывая сдвигаемые биты и добавляя нули слева. |

## [& (Побитовое И)](http://learn.javascript.ru/bitwise-operators" \l "%D0%BF%D0%BE%D0%B1%D0%B8%D1%82%D0%BE%D0%B2%D0%BE%D0%B5-%D0%B8)

Выполняет операцию И над каждой парой бит.

Результат a & b равен единице только когда оба бита a и b равны единице.

Таблица истинности для &:

|  |  |  |
| --- | --- | --- |
| **a** | **b** | **a & b** |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

## [~ (Побитовое НЕ)](http://learn.javascript.ru/bitwise-operators" \l "%D0%BF%D0%BE%D0%B1%D0%B8%D1%82%D0%BE%D0%B2%D0%BE%D0%B5-%D0%BD%D0%B5)

Производит операцию НЕ над каждым битом, заменяя его на обратный ему.

Таблица истинности для НЕ:

|  |  |
| --- | --- |
| **a** | **~a** |
| 0 | 1 |
| 1 | 0 |

### [<< (Битовый сдвиг влево)](http://learn.javascript.ru/bitwise-operators" \l "%D0%B1%D0%B8%D1%82%D0%BE%D0%B2%D1%8B%D0%B9-%D1%81%D0%B4%D0%B2%D0%B8%D0%B3-%D0%B2%D0%BB%D0%B5%D0%B2%D0%BE)

Операторы битового сдвига принимают два операнда. Первый – это число для сдвига, а второй – количество битов, которые нужно сдвинуть в первом операнде.

Оператор << сдвигает первый операнд на указанное число битов влево. Ли

9 (по осн.10)

= 00000000000000000000000000001001 (по осн.2)

--------------------------------

9 << 2 (по осн.10)

= 00000000000000000000000000100100 (по осн.2)

= 36 (по осн.10)

### [>> (Правый битовый сдвиг, переносящий знак)](http://learn.javascript.ru/bitwise-operators" \l "%D0%BF%D1%80%D0%B0%D0%B2%D1%8B%D0%B9-%D0%B1%D0%B8%D1%82%D0%BE%D0%B2%D1%8B%D0%B9-%D1%81%D0%B4%D0%B2%D0%B8%D0%B3-%D0%BF%D0%B5%D1%80%D0%B5%D0%BD%D0%BE%D1%81%D1%8F%D1%89%D0%B8%D0%B9-%D0%B7%D0%BD%D0%B0%D0%BA)

Этот оператор сдвигает биты вправо, отбрасывая лишние. При этом слева добавляется копия крайнего-левого бита.

Знак числа (представленный крайним-левым битом) при этом не меняется, так как новый крайний-левый бит имеет то же значение, что и исходном числе.

Поэтому он назван «переносящим знак».

Например, 9 >> 2 даст 2:

9 (по осн.10)

= 00000000000000000000000000001001 (по осн.2)

--------------------------------

9 >> 2 (по осн.10)

= 00000000000000000000000000000010 (по осн.2)

= 2 (по осн.10)