**[Строковое преобразование](http://learn.javascript.ru/types-conversion" \l "%D1%81%D1%82%D1%80%D0%BE%D0%BA%D0%BE%D0%B2%D0%BE%D0%B5-%D0%BF%D1%80%D0%B5%D0%BE%D0%B1%D1%80%D0%B0%D0%B7%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)**

Строковое преобразование происходит, когда требуется представление чего-либо в виде строки. Например, его производит функция alert.

var a = true;

alert( a ); // "true"

Можно также осуществить преобразование явным вызовом String(val):

alert( String(null) === "null" ); // true

Как видно из примеров выше, преобразование происходит наиболее очевидным способом, «как есть»: false становится "false", null – "null", undefined – "undefined" и т.п.

Также для явного преобразования применяется оператор "+", у которого один из аргументов строка. В этом случае он приводит к строке и другой аргумент, например:

alert( true + "test" ); // "truetest"

alert( "123" + undefined ); // "123undefined"

## [Численное преобразование](http://learn.javascript.ru/types-conversion" \l "%D1%87%D0%B8%D1%81%D0%BB%D0%B5%D0%BD%D0%BD%D0%BE%D0%B5-%D0%BF%D1%80%D0%B5%D0%BE%D0%B1%D1%80%D0%B0%D0%B7%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)

Численное преобразование происходит в математических функциях и выражениях, а также при сравнении данных различных типов (кроме сравнений ===, !==).

Для преобразования к числу в явном виде можно вызвать Number(val), либо, что короче, поставить перед выражением унарный плюс "+":

var a = +"123"; // 123

var a = Number("123"); // 123, тот же эффект

|  | **Преобразуется в...** |
| --- | --- |
| undefined | NaN |
| null | 0 |
| true / false | 1 / 0 |
| Строка | Пробельные символы по краям обрезаются. Далее, если остаётся пустая строка, то 0, иначе из непустой строки "считывается" число, при ошибке результат NaN. |

Например

// после обрезания пробельных символов останется "123"

alert( +" \n 123 \n \n" ); // 123

еще примери

логичиские значение

alert( +true ); // 1

alert( +false ); // 0

Сравнение разных типов – значит численное преобразование:

alert( "\n0 " == 0 ); // true

При этом строка "\n0" преобразуется к числу, как указано выше: начальные и конечные пробелы обрезаются, получается строка "0", которая равна 0.

С ЛОГИЧИСКИМИ ЗНАЧЕНИЯМИ

alert( "\n" == false );

alert( "1" == true )

Сравнение разных типов – значит численное преобразование:

alert( "\n0 " == 0 ); // true

… А в логическом контексте левая часть (0) даст false, правая ("\n0\n") – true, так как любая не пустая строка в логическом контексте равна true:

if ("\n0\n") {

alert( "true, совсем не как 0!" );

}

А в логичиском преобразаваний типов в JS tnj cjdthityj yjhvfkmyj