Преобразование типов

**Строковое преобразование**

Для примитивных значений преобразование происходит очевидным образом. false становится "false", null становится "null" и т.п.

**Численное преобразование**

Происходит в математических функциях и выражениях.

Например, когда операция деления / применяется не к числу:

alert( "6" / "2" ); // 3, строки преобразуются в числа

Number(value) явно преобразует value к числу.

Унарный плюс преобразует также значение к числу.

let apples = "2";

let oranges = "3";

alert( +apples + +oranges );

Сначала выполнятся унарные плюсы, которые приведут строки к числам, а затем бинарный '+' их сложит.

Если строка не может быть явно приведена к числу, то результатом преобразования будет NaN

alert(Number("Любая строка вместо числа")) // NaN, преобразование не удалось

**Правила численного преобразования:**

|  |  |
| --- | --- |
| Значение | Преобразуется в… |
| undefined | NaN |
| null | 0 |
| true / false | 1 / 0 |
| string | Пробельные символы (пробелы, знаки табуляции \t, знаки новой строки \n и т. п.) по краям обрезаются. Далее, если остаётся пустая строка, то получаем 0, иначе из непустой строки «считывается» число. При ошибке результат NaN. |

**Логическое преобразование**

Происходит в логических операциях, или явно Boolean(value).

Правила логического преобразования:

|  |  |
| --- | --- |
| Значение | Преобразуется в… |
| Значения, которые интуитивно «пустые», вроде 0, пустой строки, null, undefined и NaN | false |
| Все остальное | true |
| "0" и строки из одних пробелов типа " " | true |