

Frontender[1.0] JavaScript - Преобразование и приведение типов (type conversion & casting)

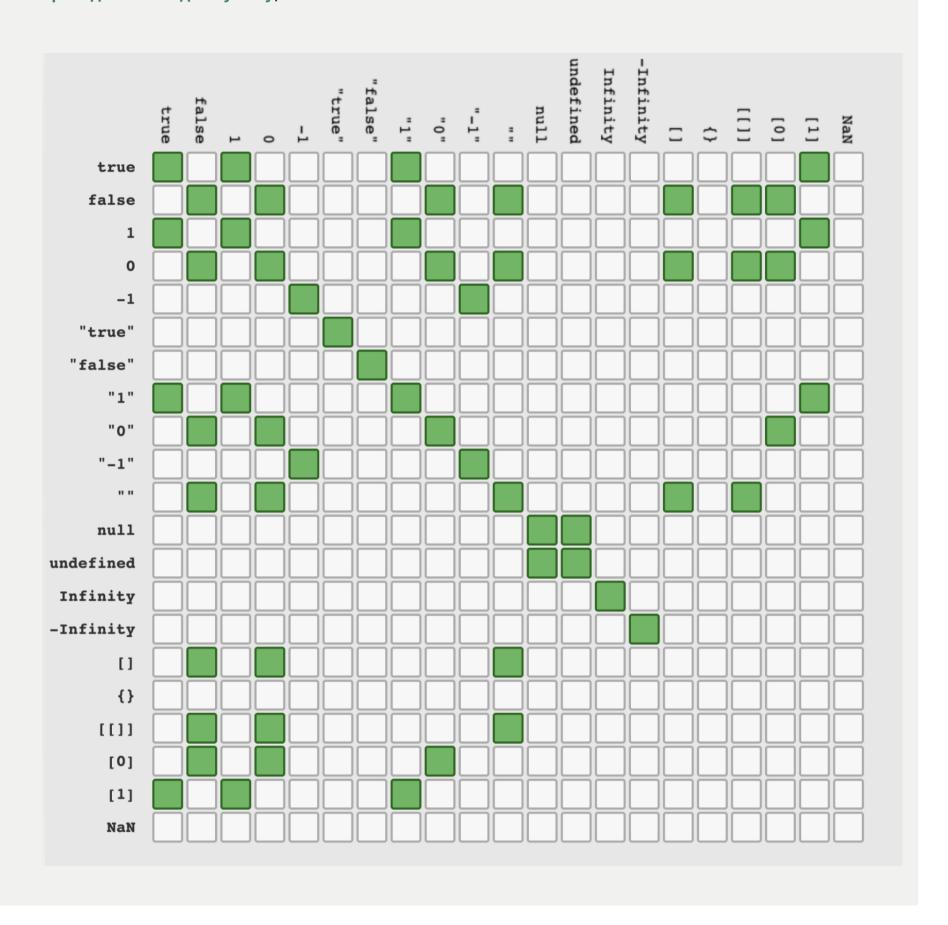
| | https://youtu.be/YI-AUBPNxms |
|---------------|---|
| | https://t.me/Dmitry_Kolotilshikov |
| | https://github.com/DmitryKolotilshikov/ |
| ⊗ Boosty | https://boosty.to/dmitry_ko |
| # Номер урока | 6 |



Задачи к этому уроку тут https://boosty.to/dmitry_ko

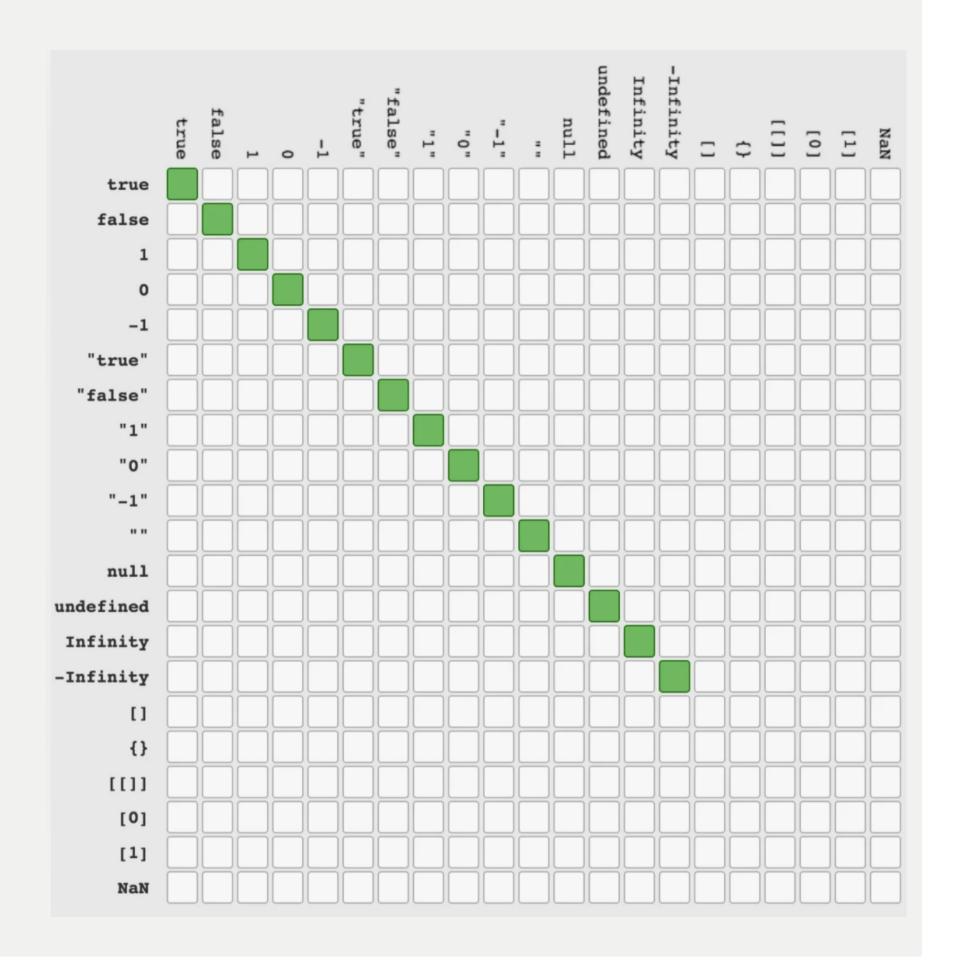


== (двойное равно используется для сравнения двух значений на равенство) (**сравнивается с приведением к одному типу**)





=== (тройное равно используется для сравнения двух значений на равенство) (сравнивается по типу и по значению, без приведения типов)





Использование === **предпочтительнее**, так как оно предсказуемо и помогает избежать ошибок в программе, связанных с неочевидным преобразованием типов.

```
</body>
</html>
```

```
// Type conversion
let value = 10;
// число в строку
value = value + '';
value = `${value}`;
value = value.toString();
value = String(value);
// строку в число
value = Number(value);
value = +value;
value = parseInt(value);
value = '10.22';
value = parseFloat(value);
value = 10.29442313.toFixed(3);
// console.log(value);
// -----
// Casting
let val = 10;
val = 10 + '10'; // ('1010') string
val = 10 * '10'; // (100) number
val = 10 / '10'; // (1) number
val = 10 - '10'; // (0) number
val = 10 % '10'; // (0) number
val = 10 + true; // (11) true приводится в 1
val = 10 + false // (10) false приводтся к 0, так как это falsy значение
val = 10 + null // (10) null приводтся к 0, так как это falsy значение
val = 10 + undefined // (NaN)
val = 10 + NaN // (NaN)
val = 10 + [] // ('10') string
val = 10 + \{\} // (10[object Object]) string
val = true + true; // (2) number
val = true + false; // (1) number
val = false + false; // (0) number
// console.log(val);
// -----
// Операторы сравнения == и ===
// == сравнивается с приведением к одному типу
// === сравнивается по типу и по значению, без приведения типов
// Результат сравнения всегда true или false
let res;
res = true == true; // true
```

```
res = true === true; // true
res = true == 1; // true
res = true === 1; // false

res = false == 0; // true
res = false === 0; // false

res = 1 == '1'; // true
res = 1 === '1'; // false

res = null == undefined; // true
res = null === undefined; // false

res = '' == []; // true
res = '' == []; // false

// и тд, есть много комбинаций, смотри таблицы приведения типов в pdf к уроку

console.log(res);
```

5