



Embora já saibamos que em Javascript temos os valores "booleanos" verdadeiro (**True**) e falso (**False**), agora surge a pergunta: a que se referem então o **truthy** e o **falsy**?

Em palavras simples, podemos dizer que cada elemento da linguagem possui um valor booleano intrínseco-primitivo. Mas não se preocupe, não é tão complexo quanto parece.

**Pequeno resumo:**

Em JavaScript as variáveis ​​são fracamente e dinamicamente tipadas, ou seja, a linguagem não se importa como são declaradas ou como se alteram seus valores:

|  |
| --- |
| let a; a = 1; // a é um número. a = "1"; // a é uma string. |

Valores que, embora sejam "diferentes", são iguais a verdadeiros quando comparados com **==** (comparação fraca) porque o JavaScript não compara os tipos de dados em jogo (string, número, etc ...):

|  |
| --- |
| // Todos são verdadeiros. 100 == '100'; 100 == [100]; |

Por outro lado, se usarmos o operador de igualdade estrita **===**, resultados falsos serão obtidos porque, neste caso, além dos valores, o tipo de dado também é comparado:

|  |
| --- |
| // Todos são falsos 100 === '100'; 100 === [100]; |

Internamente, o JavaScript define um valor de um dos seis tipos de dados primitivos para nossas variáveis:

* **Undefined** (uma variável sem valor definido).
* **Null** (um simples valor nulo)
* **Boolean** (verdadeiro [**true**] ou falso [**false**])
* **Números** (incluindo infinito [**infinity**] e “não número” [**NaN**])
* **String** (dados de texto)
* **Symbol** (tipo de dado cujo valor é único e imutável)

Todos os demais são objetos. Então:

**O que é Truthy e Falsy?**

Além de um tipo, cada valor também possui um valor booleano inerente que é essencial e permanente, que faz parte de sua natureza, geralmente conhecido como **truthy** ou **falsy**. Algumas das regras são um pouco estranhas, portanto, entender os conceitos e o efeito da comparação ajuda ao utilizar esses valores.

Os seguintes valores são sempre **falsy**:

* **false**
* **0** (zero)
* **‘’** ou **“”** (string vazia)
* **null**
* **undefined**
* **NaN** (Ex: o resultado de 1/0)

Todo o resto é **truthy**, o que inclui:

* **‘0’** (uma string que contenha um simples 0)
* **‘false’** (uma string que contenha o texto “false”)
* **[]** (un array vazio)\*
* **{}** (um objeto vazio)\*
* **function(){}** (uma função vazia)

\*Embora ainda não tenhamos visto esses tipos de dados, não se assuste, são tópicos que veremos em futuras aulas. Mais tarde, quando virmos Arrays e objetos, você vai poder voltar a esta parte e entender melhor o que queremos dizer!

Levando em consideração todas essas novas informações, se atente para o seguinte: às vezes, podem ocorrer situações inesperadas ao comparar valores **truthy** e **falsy** usando == igualdade. Abaixo, observe na tabela alguns valores que seriam obtidos: