

JS

# 1. Kintamieji

- Lokālūs kintamieji - tai kintamieji sukurti funkcijojē. Jie galioja sūkurtōjē funkcijojē.
- Globalūs kintamieji - tai kintamieji sukurti ne funkcijojē. Jie galioja visoje programojē.

Keyword	Scope	Redeclaration	Reassignment
<b>var</b>	function scope	yes	yes
<b>let</b>	block scope	no	yes
<b>const</b>	block scope	no	no

# Kintamųjų tipai

- Primityvūs kintamieji:

- Skaičiai / number (0, 1, 1.3, NaN, Infinity)
- Raidžių eilutės/ strings ('raidės', "eilutės")
- Loginiai/ boolean (true/false)
- null
- undefined

```
let foo = 42; // foo is now a number  
foo = "bar"; // foo is now a string  
foo = true; // foo is now a boolean
```

# Number

- NaN - (not a number) - yra speciali skaičiaus reikšmė, su kuria paprastai susiduriama, kai aritmetinės operacijos rezultato negalima išreikšti skaičiumi. Tai taip pat yra vienintelė „JavaScript“ vertė, kuri nėra lygi sau pačiam.
- Infinity - begalybė yra globalaus objekto savybė. Kitaip tariant, tai yra globalus kintamasis.

```
console.log(Infinity); /* Infinity */  
console.log(Infinity + 1); /* Infinity */  
console.log(Math.pow(10, 1000)); /* Infinity */  
console.log(Math.log(0)); /* -Infinity */  
console.log(1 / Infinity); /* 0 */  
console.log(1 / 0); /* Infinity */
```

# Null

- „JavaScript“ programoje null yra speciali reikšmė, nurodanti tuščią arba nežinomą reikšmę.

```
let number = null;
```

# Undefined

- Jei kintamasis deklaruojamas, bet reikšmė nepriskirta, to kintamojo reikšmė bus neapibrėžta.

```
let name;  
console.log(name); // undefined  
  
// Čia name yra niekam neprilygintas, todėl neveiks.  
// Dažnai meta console, kai padarome klaidą priskyrimo.
```







