Web-technológia

JavaScript

Dr. Hatwagner F. Miklós

Széchenyi István Egyetem, Győr

https://github.com/wajzy/GKxB_INTM049.git

2021. szeptember 18.

Jellemzők:

- Egyetlen típus létezik csak: 64 bites lebegőpontos ábrázolás
- Pl. `42`, `12.34`, `-34.56`, `1e3`, `-1e3`, `1e-3`, `-1e-3`, `-1.23e-4`, `-1.23E+4`, ...
- Különleges értékek: `Infinity`, `-Infinity`, `NaN`
- Pl. `0/0` → NaN, `1/0` → Infinity

Operátorok

- $+$ `5+3` → 8
- $-$ `5-3` → 2
- $\times$ `5*3` → 15
- $/$ `5/3` → 1.6666666666666667
- $\%$ `5%3` → 2, `-5%3` → -2, `5%-3` → 2
- $**$ `5**3` → 125

Precedencia táblázat

Jellemzők

- Unicode, 16 bit karakterenként
- Nincs specifikus típus egyetlen karakter tárolására
- Jelölés: ’-ok vagy "-ek között
- Pl. `’JavaScript’`, `"JavaScript"`, `"Guns ’n’ Roses"`, `"Egy\nKettő\nHárom"`, `’Guns \’n\’ Roses’`, `"Új sor \\n megadásával kérhető."`
- *Template literal:* ‘-ek között, kifejezések kiértékelése
- Pl. ‘`5 * 3 = ${5*3}`‘ $\rightarrow$ `"5 * 3 = 15"`

Operátor

+ `"Java" + ’Script’` $\rightarrow$ `"JavaScript"`

Jellemzők

- Értékek: `true`, `false`
- Pl. `5 < 3` → `false`

Logikai operátorok

és `true && false` → `false`

vagy `true || false` → `true`

nem `!true` → `false`

Short circuit evaluation (pl. alapérték megadására):
`undefined || "Gizi"` → `"Gizi"`, `null || "Gizi"` → `"Gizi"`,
`"" || "Gizi"` → `"Gizi"`, `"Gizi" || "Mari"` → `"Gizi"`

Relációs operátorok

- `==`, `!=`, `<`, `<=`, `>`, `>=`
- Pl. `"Bill" != "Gates"` → `true`, `Infinity == Infinity` → `true`, de `NaN == NaN` → `false`
- Karakterláncok összehasonlítása: karakterkódok alapján

Üres értékek: valaminek a hiányát jelzik

- `undefined`
- `null`

Egyoperandusú operátorok

típus `typeof(5)` → `"number"`, `typeof("Gizi")` → `"string"`

– `-(5)` → `-5`

Háromoperandusú operátor

?: `1<2?"kisebb":"nagyobb"` → `"kisebb"`

Néhány példa:

- `5 * null` → `0`
- `"5" - 3` → `2`
- `"5" + 3` → `"53"`
- `"öt" * 3` → `NaN`, `5 * undefined` → `NaN`
- `false == 0` → `true`, `true == 1` → `true`, `true == 2` → `false`,
- `"" == false` → `true`
- Definiált az érték? `null == undefined` → `true`, `null == 0` → `false`

Típusok egyezését megkövetelő operátorok: `===`, `!==`

Változók (*variable*, *binding*)

- Deklaráció: `let` (blokk hatáskör), `var` (függvény hatáskör)
- Inkább tekinthető értékre mutató referenciának, mint valódi tárolónak

Példa

a → ReferenceError: a is not defined
let a
a → undefined
a = 5
a → 5
let b = 3, c
a * b → 15

Konstansok

- `const`

Példa

const c = 3.14
c = 2 → TypeError: invalid assignment to const 'c'

Névadási szabályok

- betűket, számokat, $ és _ karaktereket tartalmazhat
- számjeggyel nem kezdődhet
- nem lehet foglalt szó (pl. `let`)
- kis- és nagybetűket megkülönbözteti
- javasolt stílus: *camel case* (`hosszuValtozoNeve`)

Változókkal használható (összetett és unáris) operátorok

- +=, -=, *=, /=, %=, &&=, ||=, **=, ...
- ++, --

Környezet (*environment*)

- adott pillanatban létező változók és értékeik
- gyakorlatilag soha nincs üres környezet

Megjegyzések

- // egysoros
- /* több
  soros */

Szelekció

- `if(`*feltétel*`)` *utasítás*;
- `if(`*feltétel*`) {`
  `// utasítások`
  `}`
- `if(`*feltétel*`) {`
  `// igaz ág utasításai`
  `} else {`
  `// hamis ág utasításai`
  `}`

- Mikor nem teljesül a *feltétel*?
  - false
  - 0
  - ""
  - NaN
  - null
  - undefined

Több irányú elágazás

```
switch(kifejezés) {
    case érték1:
        // utasítások
        break;
    case érték2:
    case érték3:
        // utasítások
        break;
    default:
        // utasítások
        break;
}
```

Az értéknek *és* a típusnak is egyeznie kell!
A *default* ág elhagyható.

Ciklusok

`for`(*előkészítés*; *ismétlési_feltétel*; *frissítés*) {
    // Ciklusmag utasításai
}

`while`(*ismétlési_feltétel*) {
    // Ciklusmag utasításai
}

`do` {
    // Ciklusmag utasításai
} `while` (*ismétlési_feltétel*);

`break`, `continue`

Háromszög rajzolás (megoldás)

A böngésző JavaScript konzolján egy sornyi szöveget a `console.log()` hívással tud megjeleníteni. Használja ezt a következő háromszög megrajzolására:

```
*
**
***
****
*****
```

X rajzolás (megoldás)

Most rajzoljon 5x5-ös méretű X-et csillagokból:

```
*   *
 * *
  *
 * *
*   *
```

Sakktábla (megoldás)

Rajzoljon meg egy 8x8-as méretű sakktáblát, szintén csillagokból!

```
 * * * *
* * * *
 * * * *
* * * *
 * * * *
* * * *
 * * * *
* * * *
```

FizzBuzz (megoldás)

Vizsgálja meg az egész számokat 1-től 100-ig, majd a vizsgálat eredményét jelenítse meg egymás alatti sorokban! Ha a szám osztható 3-mal, írja ki, hogy *Fizz*, ha 5-tel osztható, akkor azt, hogy *Buzz*, ha pedig 3-mal és 5-tel is osztható, akkor azt, hogy *FizzBuzz*! Ha egyik számmal sem osztható, akkor írja ki a vizsgált számot!

```
1
2
Fizz
4
Buzz
Fizz
...
```