

## ADATSZERKEZETEK, ADATFELDOLGOZÁS

Adatok ábrázolása JavaScript programozási nyelven, a JavaScript típusai. Adatok feldolgozása JavaScriptben tömbfüggvények segítségével.

### EGYSZERŰ TÍPUSOK

#### BOOLEAN

Logikai típus, értékei `true` és `false`.

#### NUMBER

Szám típus, mindenféle számot magába foglal.

#### STRING

Szöveg típus, magába foglalja a karakter típust.

#### UNDEFINED

Speciális típus egy darab lehetséges értékkel (`undefined`). Hiányzó értéket jelöl, nem inicializált változók kezdőértéke.

### ÖSSZETETT TÍPUSOK

#### ARRAY

Tömb típus, tetszőleges típusú elemek rendezett sorozata.

```
[1.47, "apple", true, []]
```

#### OBJECT

Objektum (rekord) típus, név érték párok rendezett halmaza.

```
{
  name: "Jean-Luc Picard",
  rank: "Admiral",
  active: false
}
```

#### FUNCTION

Külön típus a nyelvben, futtatható érték.

```
function f(a, b) {
  return a + b;
}
```

```
(a, b) => a + b;
```

### VÁLTOZÓK, KONSTANSOK

A JavaScript egy *dinamikusan* és *gyengén* típusos programozási nyelv.

#### VÁLTOZÓK LÉTREHOZÁSA

```
let a = 1701;
```

#### KONSTANSOK LÉTREHOZÁSA

```
const G = 9.81;
```

### TÖMBFÜGGVÉNYEK

Programozási tételek megvalósítása magasabb rendű függvényekkel.

#### MAP

Elemek leképezése (másolás programozási tétel)

```
[1, 2, 3, 4, 5].map((x) => x * 2);
// -> [2, 4, 6, 8, 10]
```

#### FILTER

Elemek kiválogatása (kiválogatás programozási tétel)

```
[1, 2, 3, 4, 5].filter((x) => x % 2 === 0);
// -> [2, 4]
```

#### REDUCE

Elemek összegzése függvénnyel (sorozatszámítás programozási tétel)

```
[1, 2, 3, 4, 5].reduce((s, x) => s + x);
// -> 15
```

### REFERENCIÁK

#### JavaScript dokumentáció

<https://developer.mozilla.org/en-US/docs/Web/JavaScript>