

Programmeerimise algõpetus Javascripti baasil

Martti Raavel

martti.raavel@tlu.ee

Kolmas loeng

- Millest rääkisime eelmisel korral?
 - Funktsioonid
 - Massiiv
 - Tegevused massiividega
- Sisendi kontrollimine
- Tegevused stringidega
- Math funktsioonid
- Harjutused

Sisendi kontrollimine

Sisendi kontrollimiseks on mitmeid võimalusi, näiteks:

- kas üldse sisestati midagi?
- kas sisestatud väärtus on õige tüübiga?
- kas sisestatud väärtus on õiges vahemikus?
- kas sisestatud väärtus vastab mõnele muule tingimusele?

Väärtuste teisendumine tõeväärtuseks

Väärtus	Tõeväärtus
undefined	false
null	false
0 (arv null)	false
NaN	false
'' (tühi string)	false
false	false

Väärtuste teisendumine tõeväärtuseks

```
const age = prompt('Sisesta oma vanus');  
if (!age) {  
    console.log('Sisestasid tühja väärtuse');  
} else {  
    console.log('Sisestasid väärtuse ' + age);  
}
```

Sisendi kontrollimine tüübi järgi (typeof)

```
typeof 1; // 'number'  
typeof '1'; // 'string'  
typeof true; // 'boolean'  
typeof undefined; // 'undefined'  
typeof null; // 'object'  
typeof []; // 'object'  
typeof {}; // 'object'
```

Sisendi teisendamine ja kontrollimine (string numbriks)

```
const age = Number(prompt('Sisesta oma vanus'));
if(age) {
    console.log('Sa oled ' + age + ' aastat vana');
} else {
    console.log('Sa ei sisestanud arvu');
}
```

Sisendi kontrollimine kindlate väärtuste suhtes

```
const choice = prompt('Sisesta oma valik (kivi, paber, käärid)');
if (choice === 'kivi' || choice === 'paber' || choice === 'käärid') {
    console.log('Sisestasid valiku ' + choice);
} else {
    console.log('Sisestasid vale valiku');
}
```

Sisendi kontrollimine kindlate väärtuste suhtes

```
const choices = ['kivi', 'paber', 'käärid'];
const choice = prompt('Sisesta oma valik (kivi, paber, käärid)');
if (choices.includes(choice)) {
    console.log('Sisestasid valiku ' + choice);
} else {
    console.log('Sisestasid vale valiku');
}
```


Tühikute eemaldamine sisendist (trim())

```
const choices = ['kivi', 'paber', 'käärid'];
const choice = prompt('Sisesta oma valik (kivi, paber, käärid)');
if (choices.includes(choice.trim())) {
    console.log('Sisestasid valiku ' + choice);
} else {
    console.log('Sisestasid vale valiku');
}
```

Ülesanne: Kontrollige, kas sisestatud number on vahemikus 1 kuni 100.

Looge funktsioon nimega `checkInRange`, mis võtab argumendina numbri.

Funktsiooni sees kontrollige, kas antud number on suurem kui 1 ja väiksem kui 100.

Kui see vastab tingimustele, väljastage konsooli sõnum: "Number on vahemikus 1 kuni 100".

Kui see ei vasta tingimustele, väljastage konsooli sõnum: "Number ei ole vahemikus 1 kuni 100".

Tegevused stringidega

Kuigi stringide puhul on JavaScripti tegemist primitiivse andmetüübiga, on stringidel olemas ka meetodid, mida saab kasutada erinevate toimingute sooritamiseks stringidega. Stringide meetodite puhul JavaScriptis ei ole tegemist otseselt stringi manipuleerimisega, vaid pigem uue stringi loomisega, mis on eelmisest stringist erinev. Ehk stringide meetodid tagastavad uue stringi, mitte ei muuda olemasolevat.

Stringi pikkus

```
const name = "John Doe";  
console.log(name.length); // 8
```

Stringi ühendamine

```
const firstName = "John";  
const lastName = "Doe";  
const fullName = firstName.concat(" ", lastName);  
console.log(fullName); // John Doe
```

Väiketähtedeks muuttamine

```
const name = "John Doe";  
console.log(name.toLowerCase()); // john doe
```

Suurtähtedeks muutmine

```
const name = "John Doe";  
console.log(name.toUpperCase()); // JOHN DOE
```

Alamstringi olemasolu kontrollimine

Meetod `includes()` tagastab tõeväärtuse, kas string sisaldab määratud alamstringi.

```
const name = "John Doe";  
console.log(name.includes("John")); // true  
console.log(name.includes("Doe")); // true  
console.log(name.includes("john")); // false
```


Stringi poolitamine

Meetod `slice()` tagastab stringi osa, mis algab määratud indeksist ja lõpeb määratud indeksiga (indeksid on nullist alates).

```
const name = "John Doe";  
console.log(name.slice(0, 4)); // John  
console.log(name.slice(5, 8)); // Doe
```

Stringi tükeldamine massiiviks

Meetod `split()` tagastab massiivi, kus string on tükeldatud määratud alamstringi kohalt.

```
const name = "John Doe";  
console.log(name.split(" ")); // ["John", "Doe"]  
console.log(name.split("")); // ["J", "o", "h", "n", " ", "D", "o", "e"]
```

Ülesanne: Kontrollige, kas sisestatud string on email.

Looge funktsioon nimega `checkEmail`, mis võtab argumendina stringi.

Kui see vastab emaili formaadile, tagastage `true`.

Kui see ei vasta emaili formaadile, tagastage `false`.

Ülesanne: Looge funktsioon, mis muudab stringi esimese tähe suureks

Looge funktsioon nimega `capitalizeFirstLetter`, mis võtab sisendina stringi.

Funktsioonis kasutage JavaScripti stringi meetodeid, et muuta stringi esimene täht suureks.

Tagastage muudetud string.

Math objekti omadused ja meetodid

Math objekt on JavaScripti sisseehitatud objekt, millel on omadused ja meetodid matemaatiliste konstantide ja funktsioonide jaoks.

Math objekti omadused (konstandid)

- Math.PI - pi konstant

Math objekti meetodid

- `Math.abs(x)` - tagastab arvu absoluutväärtuse
- `Math.ceil(x)` - tagastab arvu ümardatuna ülespoole
- `Math.floor(x)` - tagastab arvu ümardatuna allapoole
- `Math.round(x)` - tagastab arvu ümardatuna lähima täisarvuni
- `Math.max(x, y, z, ..., n)` - tagastab suurima arvu
- `Math.min(x, y, z, ..., n)` - tagastab väikseima arvu
- `Math.pow(x, y)` - tagastab arvu x astmes y
- `Math.sqrt(x)` - tagastab arvu ruutjuure
- `Math.random()` - tagastab suvalise arvu vahemikus 0 kuni 1

Math.random() meetodi kasutamine (täring)

```
const dice = Math.floor(Math.random() * 6) + 1;  
console.log(dice);
```


Ülesanne: Looge funktsioon, mis võtab stringi ja tagastab selle pööratud kujul

Looge funktsioon nimega `reverseString`, mis võtab sisendina stringi.

Kasutage JavaScripti stringi ja massiivi meetodeid, et pöörata string ümber.

Tagastage pööratud string.

Kodune ülesanne

- Kodune ülesanne