

# JavaScript ohjelmointi

## *Silmukat*

MARGIT TENNOSAARIN MATERIAALISTA  
LAURA JÄRVISEN MUOKKAAMA

# Kertaus viime viikolta

- Ehtolauseet antavat vaihtoehtoja ohjelman toiminnalle
- Ehtoja voi olla yksi tai useita
- If-lauseiden sijasta voi käyttää myös switch-lausetta tilanteissa, joissa vaihtoehtoja on rajattu määrä
- Voit käyttää myös ternääristä operaattoria yksinkertaisen if-lauseen korvaamisessa

# Silmukat

[https://github.com/bc-web-ohjelmistokehitys/WP25K-JS/blob/main/03\\_viikko/silmukat.md](https://github.com/bc-web-ohjelmistokehitys/WP25K-JS/blob/main/03_viikko/silmukat.md)

# Mihin silmukoita käytetään?

- Silmukoita käytetään toistamaan koodilohko useita kertoja.
- Hyödyllisiä kokoelmien läpikäymisessä, tehtävien toistamisessa ja toistuvien prosessien automatisoinnissa.
- JavaScript tarjoaa useita erilaisia silmukoita eri tilanteiden käsittelemiseksi.

# For-silmukka

- Käytetään, kun tiedetään etukäteen, kuinka monta kertaa koodilohko halutaan toistaa.

```
for (alustus; ehto; muutos) {  
    // Koodi, joka suoritetaan joka kierroksella  
}  
  
for (let i = 0; i < 5; i++) {  
    console.log(i);  
}
```

# While-silmukka

- Toistetaan niin kauan, kun ehto täyttyy

```
while (ehto) {  
    // Koodi, joka suoritetaan, kun ehto täyttyy  
}  
  
let i = 0;  
while (i < 5) {  
    console.log(i);  
    i++;  
}
```

# do ... while -silmukka

- Samantapainen, kuin *while*-silmukka, mutta koodilohko suoritetaan aina vähintään kerran

```
do {  
  // Koodi, joka suoritetaan  
} while (ehto);  
  
let i = 0;  
do {  
  console.log(i);  
  i++;  
} while (i < 5);
```

# for ... in -silmukka

- Käytetään olion ominaisuuksien läpikäymiseen - eli **harjoitellaan myöhemmin kurssilla.**

```
for (avain in olio) {  
    // Koodi, joka suoritetaan jokaiselle ominaisuudelle  
}  
  
const henkilö = { name: "Aliisa", age: 25 };  
  
for (let avain in henkilö) {  
    console.log(`${avain}: ${henkilö[avain]}`);  
}
```

# for ... of -silmukka

- Luo silmukan, joka iteroi iteroitavien olioiden (kuten taulukoiden ja merkkijonojen) yli - eli **harjoitellaan myöhemmin kurssilla.**

```
for (muuttuja of iteroitava) {  
  // Koodi, joka suoritetaan jokaiselle alkiolle  
}  
  
let hedelmät = ["omena", "banaani", "päärynä"];  
  
for (let hedelmä of hedelmät) {  
  console.log(hedelmä);  
}
```

# Silmukan ohjauslauseet

- Tilanteisiin,  
jossa silmukka  
halutaan  
lopettaa tai  
ohittaa siitä osa

```
for (let i = 0; i < 10; i++) {  
  if (i === 5) {  
    break; // silmukka lopettaa  
  }  
  console.log(i);  
}
```

```
for (let i = 0; i < 5; i++) {  
  if (i === 2) {  
    continue; // nykyinen kierros ohitetaan  
  }  
  console.log(i);  
}
```

# Prompt()

- JavaScriptissä ei ole mahdollista pyytää käyttäjältä syötettä komentorivin eli konsolin kautta. Siihen tarvitaan verkkosivua.
- Prompt() avaa ponnahdusikkunan, jonka avulla voi pyytää syötettä.
- Prompt() **jää helposti jumiin**, varsinkin silmukoiden kanssa. Silloin kannattaa:
  - siirtyä selaimen osoiterivillä toisalle (esim bc.fi) ja palata nuolella takaisin
  - TAI sulkea Live Server -yhteys ja avata se uudelleen.

# Prompt() käyttö

- Käyttäjältä syötteen kysymiseen tarvitaan html-sivu, jolle JS-linkitetään.
- Jotta syöte saadaan käyttöön, se pitää tallentaa muuttujaan

```
let syote = prompt("Anna luku välillä 1-10");
```

127.0.0.1:5500 says

Anna luku välillä 1-10

OK

Cancel

# Prompt() käyttö

- Jotta numerot toimivat aina odotetulla tavalla, kääri niitä kysyvä prompt() parseFloat()-funktion sisään:

```
let syote = parseFloat(prompt("Anna luku välillä 1-10"));  
console.log("Annettu luku:", syote);
```

# Yleisiä virheitä:

- Ikkuiset silmukat: Varmista, että silmukassa on toimiva lopetusehto.
- Yhdellä yli tai ali -virheet: Tarkista silmukan ehdot huolellisesti (< vs. <=).
- Muuttujien alustamatta jättäminen: Varmista, että silmukan muuttujat on alustettu oikein.

# Tee tehtäviä

Ensimmäinen erä silmukkkatehtäviä on lisätty viikon 3 harjoitukseen.

<https://github.com/bc-web-ohjelmistokehitys/WP25K-JS>

Huom! Opettele itse - älä pyydä tekoälyä ratkaisemaan tehtävää puolestasi.