

Javascript for Basic-perioden

Variabler – hvordan opprette dem?

```
let navn = "Henks"
```

Den vanligste måte å opprette en variabel på er med `let` koden. Denne skriver man før man skriver navnet til variabelen.

```
const number = document.getElementById("age-check").value
```

I motsetning til `let` variabler som er vanlige variabler så er `const` variabler med verdier som ikke kan endres. Altså en verdi du vet ikke skal endres.

Datatyper

Streng

```
let navn = "Henks"
```

Denne variabelen har en streng som verdien sin. Altså en streng datatype er tekst definert med `""` og `"` før og etter teksten.

Tall

```
let radius = 5
```

Denne variabelen har et tall som verdien sin. Altså en tall datatype er tall som du bare skriver inn som verdi av en variabel eller lignende uten noe før og etter det som strenger.

Boolean

```
let sant = true  
let usant = false
```

Disse variablene har Boolean som verdiene sine. Altså en boolean datatype er ordene `true` og `false` som skrives inn som verdier av variabler for å definere om de fins/gjelder eller ikke.

If-setninger

```
if(number == "") {  
  output.value = "Skriv inn et tall"  
}  
  
else if(checkbox.checked && number >= 16) {  
  output.value = "Du kan øvelseskjøre."  
}
```

`if` og `else` setninger er javascript kode som definerer hva som skjer hvis noe du definerer selv med din kode skjer. Et eksempel her er at spesifikasjonen er at en variabel er like stor (`==`) som en tom streng verdi, altså ingenting. Men en variabel kan også være større enn og like stor som et tall (`>=`). `else` setninger er ikke så forskjellige fra `if` setninger fordi de gjør at de legger til en ny kondisjon til den originale `if` setningen.

Funksjoner

Parameter

```
function loop(number) {  
  for (let index of number) {  
    console.log(index)  
  }  
} loop(["magnus", "jo", "henrik"])
```

Funksjoner med parameter er funksjoner som har variabler inni sinn (`number`) verdi som brukes til funksjonen selv.

Returverdi

```
function celsiusToFahrenheit(celsius) {  
  let number  
  number = (celsius * 1.8 + 32)  
  return number  
}
```

Funksjoner med returverdi er funksjoner som returnerer variabelen som har blitt fiklet med i funksjonen med `return`.

Arrays

Opprette

```
let arr = [2, "tall", 5]
```

For å opprette en array så definerer man en variabel med en verdi som har `[]` rundt seg slik at det kan være flere verdier i en variabel samtidig.

Hente verdi på index

```
let first = arr[0]
let last = arr[2]
```

Å hente Verdi på index betyr rett å slett å hente et spesifikt element fra arrayen med en ny variabel med verdien `[]` med tall inni seg som er hvilket element det er. `[0]` er første elementet, mens `[2]` er det siste elementet i denne arrayen.

Viktigste metoder, push(), pop(), sort(), includes()

```
arr.push("tekst")
```

`push()` er når man leger til et element i et array som man definerer rundt `()`.

```
last = arr.pop()
```

`pop()` Er når man fjerner et element fra et array. Dette kan man gjøre med en variabel som allerede har henta det elementet på index.

```
arr.sort()
```

`sort()` er når man automatisk sorterer elementene inni en array til første bokstav, tall osv.

```
arr.includes(4)
```

`includes()` fungerer som en boolean setning som sjekker om det elementet du definerer inni `()` fins og svarer dermed med `true` eller `false`.

Løkker

For løkke med teller

```
function loop(number) {
  for (let index = 0; index < number; index++) {
    console.log(index)
  }
}
```

For løkker med teller er en for løkke som har en variabel inni seg uten verdi (en verdi du må skriv inn i konsollen) som for loopen stopper i ved å definere at `index` stopper når `number` er større enn `index`.

For -of løkke på array

```
function addToList(students, list) {  
  list.innerHTML = ""  
  for (const student of students) {  
  }  
}
```

For-of løkke på array er en for løkke som kaller en verdi for hvert element i arrayen. Hvis du hadde `console.log` student så ville elementer i arrayen skrevet ut hele tiden etter hverandre.