

`</>` Web

■ JS: Hei, JavaScript!

Skrevet av: Arve Seljebu

Oversatt av: Stein Olav Romslo

Kurs: Web

Tema: Tekstbasert, Nettside

Fag: Programmering, Matematikk, Kunst og håndverk

Klassetrinn: 8.-10. klasse, Videregående skole

Introduksjon

I denne oppgåva skal du møte programmeringsspråket JavaScript. Du skal gjere den klassiske oppgåva *Hallo, verda* med ein katt.



Steg 1: Bruke JS Bin

Du skal bruke ei nettside som heiter *JS Bin* for å løyse denne oppgåva. JS Bin er ei nettside som let deg programmere JavaScript. Me kunne ha brukt nettlesaren din direkte, men i JS Bin blir stega like for alle, uavhengig av om du brukar Microsoft Edge, Mozilla Firefox, Chrome, Opera, Safari eller ein annan nettlesar. I tillegg er det mogleg å gjere denne oppgåva på eit nettbrett ved å bruke JS Bin.

Tips

Her er flere måtar å køyre JavaScript-kode på:



Lagre ei .is -fil i favoritt-teksteditoren din og åpne den i ein nettlesar

- Lagre ei .js-fil i favoritt teksteditoren din og åpne den i ein nettlesar.
- Skriv kode mellom `<script>` `</script>` i `<body>` eller `<head>` i ei HTML-fil, og åpne den i ein nettlesar.
- Skrive `<script src="fil_namn.js"></script>` i `<head>`.

✓ Sjekkliste

- ☐ Åpne JS Bin (<http://jsbin.com/?js,console>) i eit eige vindauge.
- ☐ Legg merke til at du får opp to faner; **JavaScript** og **Console**.
- ☐ Koden skal skrivast i **JavaScript**.
- ☐ I **Console** visast tekstbeskjedar som blir skrive ut.
- ☐ Skriv inn denne koden i fana **JavaScript**:

```
console.log("Hallo, verda");
```
- ☐ Trykk `ctrl + enter` (eller bruk knappen `Run`) og sjå kva som skjer i **Console**.
- ☐ Kan du endre koden til å skrive namnet ditt?

Forklaring

Her er ei forklaring til koden over:

- `console.log()` tyder at me ynskjer å køyre kommandoen med namnet `console.log`. Den skriv ut det som er mellom parentesene til `log()` i konsollen.

- ☐ `.log` er ein av fleire utskriftsmetodar. Prøv `console.error`. Ser du kva som skil den frå `console.log`?
- ☐ Tekst som blir skriven ut til konsollen ligg mellom hermeteikn ("), slik som "Hallo, verda".
- ☐ `;` tyder at kodesetninga er ferdig.

Steg 2: Lage ein funksjon

Ein funksjon er nesten som ei oppskrift. Ein brukar funksjonar for å kunne bruke same kode fleire gonger, eller bryte opp eit problem til mindre bitar. Ein funksjon kan både ta imot og gi frå seg (returnere) data. Funksjonen din skal heite `hei`, ta imot eit namn og skrive namnet til konsollen.

Sjekkliste

- ☐ Slett koden din frå steg 1.
- ☐ Skriv inn denne koden:

```
function hei(namn) {  
  console.log(namn);  
}
```

Forklaring

- ☐ `function hei` tyder at funksjonen skal heite `hei`.
- ☐ `(namn)` tyder at me kan sende inn data til funksjonen. `namn` blir ein variabel, og heldt på ein verdi som me kan endre. Me skal sjå meir på dette seinare.

- ☐ { markerer starten til funksjonen.
- ☐ } markerer slutten til funksjonen.
- ☐ Mellom { og } er det som funksjonen skal gjøre.

✓ Sjekkliste

- ☐ Akkurat nå gjør ikke funksjonen så mye, men la oss teste den likevel. Skriv dette etter funksjonen:

```
hei("Emma");  
hei("Jens");
```

- ☐ `hei("Emma")` tyder at vi skal kjøre funksjonen og sende inn "Emma".
- ☐ Trykk `ctrl + enter` for å kjøre koden.
- ☐ Ser du både "Emma" og "Jens" i konsollen?

Koden til nå:

```
function hei(namn){  
  console.log(namn)  
}  
  
hei("Emma");  
hei("Jens");
```

Tips

Du kan slette historien i **Console** ved å trykke på knappen `Clear`, eller legge inn `console.clear()` i koden din

```
console.log(hey()); // Hei, Emma!
```

Steg 3: Hei, namn!

I steg 2 gjorde me ikkje noko anna enn det som `console.log` gjer. No skal du få `hei("Emma")` til å skrive ut "Hei, Emma!".

✓ Sjekkliste

☐ I JavaScript kan me leggje saman tekst med `+`.

☐ La oss endre funksjonen `hei`:

```
function hei(namn) {  
  console.log("Hei, " + namn + "!");  
}
```

☐ Her har me lagt saman "Hei, ", namnet som blir sendt inn og "!"

☐ Køyr koden din.

☐ Står det "Hei, Emma!" og "Hei, Jens!" i **Console**?

☐ Kan du endre koden slik at den skriv ut namnet ditt?

Forklaring

I funksjonen `hei` vil `namn` vere ein variabel. Den heldt på ein verdi, og denne verdien kan endrast. Fyrst er `namn` det same som `Emma`, og så blir `namn` til `Jens`. Variablar er veldig nyttig i programmering, og me kjem til å bruke variablar mykje framover.

Steg 4: Katten seier hei

I dette siste steget skal du lage ein katt som seier hei.

✓ Sjekkliste

- ☐ Bytt ut `function hei` ved å kopiere koden under:

```
function hei(namn) {  
  console.log("< Mjau, " + namn + "! >");  
  console.log("  \\  
  console.log("    /\_\_\_\_/\ \  
  console.log("      ( o o )  
  console.log("      \\  
}
```

- ☐ Kjør `hei` med ditt eige namn.
- ☐ Er det ein katt som seier `mjau` til deg i konsollen?
- ☐ Klarar du å lage ein hund eller eit anna dyr?

Bakoverstrek

`\` brukast til spesielle bokstavar. Til dømes tyder `\n` *ny linje*. Difor må du skrive `\\` viss du vil ha ein vanleg bakoverstrek.

Steg 5: Dele prosjektet

Kanskje hadde det vore kult å sende ei kattehelsing til nokre av dei beste venene dine? La oss lage ei personleg helsing til kvar av dei!

✓ Sjekkliste

- ☐ Syt for at du køyrer `hei` med namnet til venen din:

```
hei("Namnet til venen din");
```

☐ Hald inne `ctrl + S` for å lagre.

☐ Legg merke til at det no står eit tal i adressa. Talet er versjonen av programmet ditt.

Versjon 9:  jsbin.com/dojola/9/edit?js,console

Kvar gong du endrar programmet og trykkar `ctrl + S` får programmet ein ny versjon. Slik kan du lage fleire variantar av same program, med ulike namn.

☐ Trykk på **JavaScript** slik at kodefana blir skjult:



☐ Trykk på **Run** og sjekk at det ser riktig ut.

☐ Kopier nettadressa og del med venen din.

☐ Gjenta stega for fleire av venene dine.

☐ Viss du vil åpne prosjektet att seinare kan du berre ta vare på ei av lenkene du sendte til venene dine.

Ei anna måte å dele programmet

Hugsar du oppgåva HTML: Publisert nettsida di (`./publiser/publiser_nn.html`)? Kanskje du kan laste opp JavaScript-koden til GitHub -sida di? Eit lite tips er å leggje JavaScript-koden inn i `<head>`-taggen ved å bruke desse taggane:

```
<script>
</script>
```

Du kan òg lagre JavaScript-koden i ei eiga `.js`-fil, til dømes `katt.js`. Så kan du referere til den frå ei HTML-fil på denne måten:

```
<head>
  <script src="katt.js"></script>
</head>
```

Hugs at `katt.js` må liggje i same mappe som HTML-fila. Dette skal me prøve ut seinare.

```
document.writeln();
```

```
document.body.style.whiteSpace = "pre"; //gjer at du kan ha fleire  
mellomrom etter kvarandre
```

No ser koden slik ut:

```
function hei(namn) {
  document.writeln("< Mjau, " + namn + "! >");
  document.writeln("  \\");
  document.writeln("    /\\" + "___" + "\\");
  document.writeln("      ( o o )");
  document.writeln("    \\ =0= /");
}
document.body.style.whiteSpace = "pre";
```

Steg 6: Lagre prosjektet på datamaskina di

Det er mogleg å laste ned prosjektet og lagre det på datamaskina di. La oss gjere det.



Sjekkliste



Trykk på **File > Download**.

- ☐ Ei .html -fil blir lasta ned.
- ☐ Når fila er lastet ned åpner du fila.
- ☐ Du får opp eit tomt vindauge, og utskrifta av katten blir sendt til konsoll. La oss åpne konsollen i nettlesaren din.
- ☐ **Chrome og Firefox:** Trykk på knappen **F12**. Vel **Console**.
- ☐ **Internet Explorer og Microsoft Edge:** Trykk på knappen **F12**. Vel **Konsoll**.
- ☐ I konsollen skal du sjå katten mjaue til deg.



Viss du ikkje ser katten, gjer neste steg.

- ☐ Skriv inn `hei("Namnet ditt");` og trykk **Enter**.
- ☐ Kva skjer?

Gratulerer! Du har skrive ditt fyrste JavaScript-program!

Lisens: CC BY-SA 4.0 (<http://creativecommons.org/licenses/by-sa/4.0/deed>)