

Arrays og loops

Ny datatype: **Array**

Datatyper: tekster, tal, boolean og nu også **array** - kendes på []

Et **array** er en liste/tabel med flere elementer, som regel af samme type eller samme kategori som f.eks. "titler", "navne", "kæledyr", "postnumre" og lign.

Variabler med forskellige datatyper:

```
const minTekst = "en kort eller lang sætning"; // string " "  
const pris = 200; // number  
const team = ["Martin", "Klaus", "Peter", "Alan"]; // array  
console.log(team[0]); // returnerer "Martin"  
console.log(team[3]); // returnerer "Alan"  
team.length // giver antallet af elementer i array'et lærerTeam = 4
```

Tomt array:

```
let team=[];
```



OBS!
Første
index
hedder **0**

Læs mere om Array:

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array

Loop: forEach

En “loop” kalder man den mekanisme, hvor man gentager de samme kodelinjer for flere forskellige variabler - f.eks et arrays elementer.

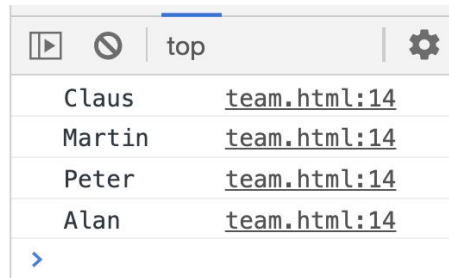
forEach er én af de loop-syntaxer javascript byder på.

```
<script>
  const team = ["Claus", "Martin", "Peter", "Alan"];

  team.forEach(visHvertElement)

  function visHvertElement(element) {
    console.log(element);
  }
</script>
```

element er et variabelnavn vi selv finder på, som skal repræsentere **hver enkelt værdi** i arrayet.



Claus	team.html:14
Martin	team.html:14
Peter	team.html:14
Alan	team.html:14

HTML DOM `querySelectorAll()` Method, W3schools: https://www.w3schools.com/jsref/met_document_queryselectorall.asp
JavaScript Array `forEach()` Method, W3schools: https://www.w3schools.com/jsref/jsref_foreach.asp

Øvelse 8 - array

Lav en ny html-fil (eller gem en gammel) og kald den **array.html**

Lav et html-skelet med et liste element i <body> med id *display*
`<ul id="display">`

Lav herunder et javascript:

Opret et **array** “playliste” med navnene på dine 10 yndlingstracks.

Loop arrayet igennem og få hvert tracknavn til at stå på hver sin linie på listen.

Commit og push til gitHub, når du er tilfreds med opgaven.

Playliste

- Two of Us
- Dig a Pony
- Across the Universe
- I Me Mine
- Dig It
- Let It Be
- Maggie Mae
- I've Got a Feeling
- One After 909
- The Long and Winding Road
- For You Blue
- Get Back

Vælg og loop igennem
flere DOM-elementer

querySelectorAll


Eksempel: Alle elementer på siden med class="red", skal forsvinde:

```
document.querySelector(".red").style.display="none";
```

Problem: kun det første element forsvinder!

For at få fat i dem alle:

```
const alle = document.querySelectorAll(".red");
```



Variablen **alle** bliver en
liste (**array**) som vi kan
loop'e igennem med
forEach

For at "gøre noget" ved alle elementer, i eksemplet: få dem til at forsvinde:

```
alle.forEach (element => element.style.display = "none");
```

querySelectorAll + event + this = **interaktion**

Det bliver endnu mere interessant, når vi kombinerer tingene.

F.eks. kan vi tilføje en eventListener til alle elementer i et array, så de bliver interaktive:

```
<body>
  <div class="boks"></div>
  <div class="boks"></div>
  <div class="boks"></div>

  <script>
    const alleBokse = document.querySelectorAll(".boks");
    alleBokse.forEach(boks=>boks.addEventListener("click", aktiver));

    function aktiver(){
      this.classList.toggle("aktiv");
    }
  </script>
</body>
```

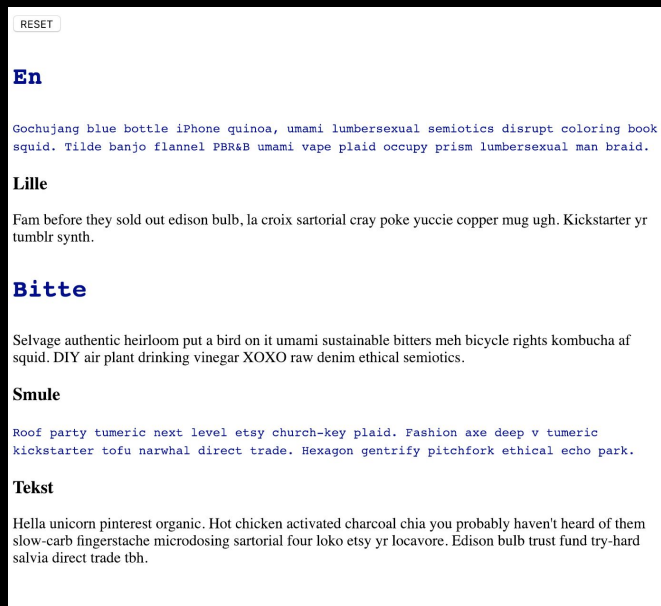
```
<style>
  .boks{
    width: 10vw;
    height: 10vw;
    margin: 1em;
    background-color: orange;
  }

  .aktiv{
    background-color: blue;
  }
</style>
```

Øvelse 9: Eventlisteners på alle p-tags

NB. I videoen hedder denne øvelse nr. 8

1. Tag udgangspunkt i denne [klargjorte html-fil](#), og gem den som **09-alle.html**
2. I scriptet - lav en eventlistener der tjekker om DOM'en er indlæst. Sæt herefter en eventlistener på hvert p-tag: Når der klikkes på en sætning, skal den styles med en klasse "typo".
3. Lav en reset-knap øverst på siden.
4. Når der klikkes på knappen, skal class "typo" fjernes fra alle p-tags igen.
5. Udvid programmet med en klasse, **typo2**, som styles alle overskrifter på samme måde - og med dobbelt størrelse.



Øvelse 9b: Skift billedkategori ved klik

1. Gem **06-skiftBillede.html** fra øvelse 6 i en ny kopi med navnet **09-skiftKategori.html**.
2. Byg videre på dit script, så det andet billede skifter kategori i stedet, når der klikkes på det.
3. Lav først et array med de fem kategorier som der tilbydes fra placeimg.com:
["animals", "arch", "people", "nature", "tech"]
4. Tilføj en eventlistener til det andet billede som kalder funktionen **skift()** ved click.
5. Lav funktionen **skift()** så den vælger en tilfældig kategori fra dit nye array og skifter både src og alt på billedet til denne kategori.

Hint:

```
function skift() {  
    const nummer = Math.round(Math.random() * 4);  
    billede.src = "https://placeimg.com/600/400/" + kategorier[nummer];  
    billede.alt = kategorier[nummer];  
};
```

Opsamling

querySelectorAll()

forEach

Array

Array-index