

Javascript | Crash Course

# Javascript loops



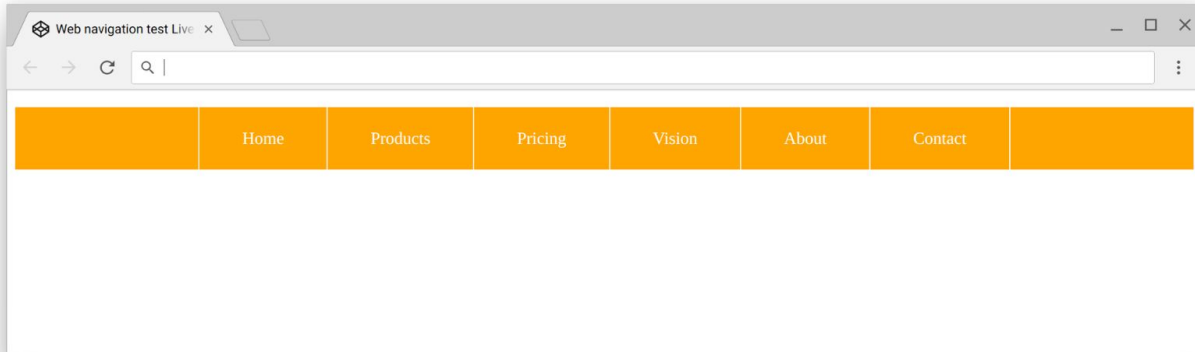
## Opgave: Menu med JS loop

I denne opgave skal du udvikle et JavaScript, der kan generere en menu med punkter baseret på elementerne i et associative array. Den færdige HTML-kode til menuen skal se ud som i eksemplet herunder.

Javascript:

```
<nav class="menu">
  <ul>
    <li><a href="home.html">Home</a></li>
    <li><a href="products.html">Products</a></li>
    <li><a href="pricing.html">Pricing</a></li>
    <li><a href="vision.html">Vision</a></li>
    <li><a href="about.html">About</a></li>
    <li><a href="contact.html">Contact</a></li>
  </ul>
</nav>
```

Menuen skal også styles med CSS. Det færdige resultat i browseren skal se ud på følgende måde:



## Det skal du bruge

De enkelte menupunkter skal ikke "hardcodes" i HTML, men genereres automatisk af JavaScript baseret på indholdet i et JS associative array. Du skal benytte et associative array, fordi opgaven både kræver en titler og et links, hvilket vi kan bruge et associative array til.

Start med at oprette et associative array med seks key/value pairs.

Javascript:

```
var menu = {  
  'home': 'home.html',  
  'products': 'products.html',  
  'pricing': 'pricing.html',  
  'vision': 'vision.html',  
  'about': 'about.html',  
  'contact': 'contact.html'  
}
```

## JavaScript methods

Opgaven kræver kendskab til en håndfuld JS methods, der skal benyttes på forskellig vis. Her har du en liste over hvilke, samt en forklaring på hver af dem og et lille eksempel på brug.

## document.createElement()

Til at oprette elementerne <nav>, <ul>, <li> og <a> skal du benytte metoden **createElement()**. I eksemplet herunder oprettes et <p> element, som gemmes i variabelen **el**.

Javascript:

```
var el = document.createElement('p');
```

## document.createTextNode()

Tekst til de enkelte menupunkter skal oprettes med JS metoden **createTextNode()**. Den ønskede tekst sættes i anførselstegn og placeres i parentesen, som vist i eksemplet herunder.

Javascript:

```
var el = document.createTextNode('Hello world');
```

## classList.add()

Hvis du vil tilføje CSS til dine elementer, bør du gøre det med JS metoden **classList.add()**. Denne metode gør det muligt at tilføje CSS classes til HTML-elementer. Husk også at oprette dem i dit stylesheet.

Javascript:

```
var box = document.querySelector('body');  
box.classList.add('.home');
```

## appendChild()

Når du opretter elementer og nodes, så bliver de ikke sammensat automatisk. Det skal du selv gøre. Til opgaven skal du benytte JS metoden **appendChild()**, der gør det muligt at tilføje nye child-elementer til eksisterende parent-elementer.

Javascript:

```
var el = document.createElement('p');  
var txt = document.createTextNode('Hello world');
```



```
e1.appendChild(txt);  
document.querySelector('body').appendChild(e1); // <p>Hello world</p>
```

## querySelector()

Hvis du vil indsætte indhold i browseren med JavaScript, skal du først vælge det element, som dit indhold skal placeres under. I eksemplet herunder vælges et element med en CSS class ved navn `.box`, der derefter gemmes i variabelen `box`.

Javascript:

```
var box = document.querySelector('.box');
```

## Sådan gør du

1. Opret et nyt HTML-dokument og navngiv det **js-menu.html**
2. Opret et **associative array** med titler og links (som vist på side 2 i dette dokument)
3. Brug JS metoden **createElement()** til at oprette følgende elementer `<nav>`, `<ul>`, `<li>` og `<a>`
4. Brug JS metoden **createTextNode()** til at oprette følgende tekst og links
5. Brug JS metoden **classList.add()** til at tilføje de nødvendige CSS classes
6. Brug **CSS** til at style menuen, så den ligner screenshot fra browser på side 2
7. Brug metoden **querySelector()** til at vælge `<body>` elementet, som skal være parent-element for menuen
8. Brug JS metoden **appendChild()** til at tilføje menuen til `<body>` elementet