

JS Workshop

Agenda

1. HTML
2. CSS
3. Oppgaver på HTML og CSS
4. JS
5. Oppgaver på JS

HTML

HTML basic structure (notepad)

```
<!DOCTYPE html>
<html>
  <head>
    <title>My Website</title>
  </head>
  <body>
    Hello, World!
  </body>
</html>
```

Some basic tags

```
<body>
  <h1>JS Workshop</h1>
  <h2>HTML</h2>
  <p>HTML defines the content of the website.</p>
  <h2>CSS</h2>
  <p>CSS styles the website.</p>
  <h2>JS</h2>
  <p>JS adds logic to the website.</p>
</body>
```

Images (attributes and self-closing tags)

```



```

Links and special symbols

```
<a href="https://knowit.no/karriere" target="_blank">Learn more about Knowit</a>
<p>Copyright &copy; G ran Humlest l 2021</p>
```

If browser doesn't handle   correctly, add `<meta charset="UTF-8" />` in the `<head>` section.

Styling

```
img {
  width: 200px;
  height: 200px;
}

h1,
h2 {
  border-bottom: 1px solid black;
}

p {
  font-size: 18px;
}
```

Browser dev tools

- View/make changes to the HTML
- Update CSS directly in the browser

Grouping/division tags

```
<body>
<header>
  <h1>JS Workshop</h1>
</header>
<main>
  <div>
    <h2>HTML</h2>
    <p>HTML defines the content of the website.</p>
    
  </div>
  <div>
    <h2>CSS</h2>
    <p>CSS styles the website.</p>
    
  </div>
  <div>
    <h2>JS</h2>
    <p>JS adds logic to the website.</p>
  </div>
</main>
</body>
```

```

</div>
</main>
<footer>
  <a href="https://knowit.no/karriere" target="_blank"
  >Learn more about Knowit</a>
  <p>Copyright &copy; Gøran Humlestøl 2021</p>
</footer>
</body>
```

Flexbox

Flexbox is a powerful tool for creating flexible layouts. https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Flexible_Box_Layout/Basic_Concepts_of_Flexbox

```
main {
  display: flex;
  gap: 20px;
  justify-content: center;
}

main div {
  border: 1px solid gray;
  background: #ffe0ce;
  padding: 10px;
  border-radius: 10px;
  width: 33%;
}
```

Media queries

Media queries allows us to conditionally change our styling based on the user's device and settings. In the example below we change the layout of the language boxes to appear left to right instead of top to bottom when the user's viewport (screen) is larger than or equal to 900px.

```
main {
  display: flex;
  flex-direction: column;
  gap: 20px;
}

main div {
  width: 100%;
  border: 1px solid gray;
  background: #ffe0ce;
  padding: 10px;
  border-radius: 10px;
}
```

```
@media (min-width: 900px) {  
  main {  
    flex-direction: row;  
    justify-content: center;  
  }  
  
  main div {  
    width: 33%;  
    max-width: 350px;  
  }  
}
```

Classes

So far we've styled **all** the **divs** in the document by using the **div** selector in our CSS. This is fine for the HTML document below, but **div** is a very common tag, so it's likely we'd add more **divs** to our document in the future. Using classes, we can be more specific in which elements to target with our CSS.

```
<body>  
<header>  
  <h1>JS Workshop</h1>  
</header>  
<main>  
  <div class="language-section html">  
    <h2>HTML</h2>  
    <p>HTML defines the content of the website.</p>  
      
  </div>  
  <div class="language-section css">  
    <h2>CSS</h2>  
    <p>CSS styles the website.</p>  
      
  </div>  
  <div class="language-section js">  
    <h2>JS</h2>  
    <p>JS adds logic to the website.</p>  
      
  </div>  
</main>  
<footer>  
  <a href="https://knowit.no/karriere" target="_blank">  
>Learn more about Knowit</a>  
>  
  <p>Copyright &copy; G ran Humlest l 2021</p>  
</footer>  
</body>
```

```
main div.language-section {
  width: 100%;
  border: 1px solid gray;
  padding: 10px;
  border-radius: 10px;
}

main div.language-section img {
  width: 200px;
  height: 200px;
  display: block;
  margin: auto;
}

main div.language-section.html {
  background: #ffe0ce;
}

main div.language-section.css {
  background: #ceebff;
}

main div.language-section.js {
  background: #fff9ce;
}

@media (min-width: 900px) {
  main {
    flex-direction: row;
    justify-content: center;
  }

  main div.language-section {
    width: 33%;
    max-width: 350px;
  }
}
```

:hover & transition

```
main div.language-section {
  ...
  transition: all 0.15s ease-in-out;
}

main div.language-section:hover {
  transform: scale(1.05);
  box-shadow: 1px 1px 10px rgba(0, 0, 0, 0.5);
}
```

Import CSS from a separate file

Add this line to the `<head>` section

```
<link rel="stylesheet" href="styles.css"/>
```