

theband



HTML og css tutorial

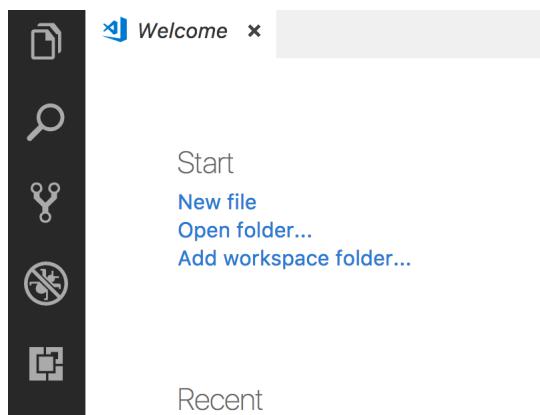
Roskilde teknikske skole · webudvikler · 2018

Forberedelse – oprettelse af mapper og filer.....	4
Oprettelse af mapper.....	4
Oprettelse af filer.....	5
Hvorfor index.html.....	5
Strukturer i html og css.....	7
Html-tags	7
CSS regler	9
Typer af selektorer.....	9
Webløsningen til The Band	10
Css filen	11
Css – layout	12
Css – navigation	13
Css – main	14
Css – footer.....	14
Index.html.....	15
Header.....	17
Main	19
Footer	19
Navigation.....	20
Indholdet på index.html	20
Undersiderne	22
Layout – Index	23
Layout – Medlemmer	23
Layout – Live.....	24
Layout – Merchandise	24

Forberedelse – oprettelse af mapper og filer

Oprettelse af mapper

Det er en god idé at have en mappe på sin computer, som er den man udvikler sine websites i. Du har sikkert allerede oprettet en mappe, lad os antage den hedder 'www'. Opret en ny mappe til projektet i din 'www' mappe, kald den f.eks. "projektnavn_ditnavn". Et mappenavn i stil med "theband_jesper" er et godt bud, hvis det altså er "The Band" opgaven og det er Jesper der arbejder med den. Inde i den mappe opretter du to nye mapper, en du navngiver img og en du navngiver css. Mappenavnet 'img' er en forkortelse for 'images'. Det er den mappe som skal indeholde de billeder som skal bruges i sitet. Mappenavnet css er en forkortelse for Cascading Style Sheets. Det er den mappe som skal indeholde dit stylesheet. Lige nu virker det måske mærligt at skulle oprette en mappe til én fil. Men det er good practise i webudvikling at adskille de forskellige filtyper i separate mapper. Det giver en god struktur i projekternes del-elementer. I takt med at udviklingsprojekterne bliver mere komplekse, vil du opleve at antallet af filer stiger i de enkelte mapper. Læg også mærke til, at alle navne er skrevet med små bogstaver. Det er endnu en good practise som du skal vænne dig til at bruge. Mapper og filer i filsystemet er nemlig 'case-sensitive'. En mappe med navnet 'Img' er således ikke en samme mappe som mappen med navnet 'img'. Når man skriver kode, er det lettest at filer og mapper har en ensartet navngivning. Så er det lettest både at skrive og huske. Derfor skrives mappe og filnavne med små bogstaver. Der kommer et afsnit senere om navngivning af filer, dokumenter og mapper. I denne opgave er alle billeder gjort klar til dig, så du kan åbne billed-mappen i opgaven og kopiere alle billeder over i img-mappen i din theband-mappe i Sites.

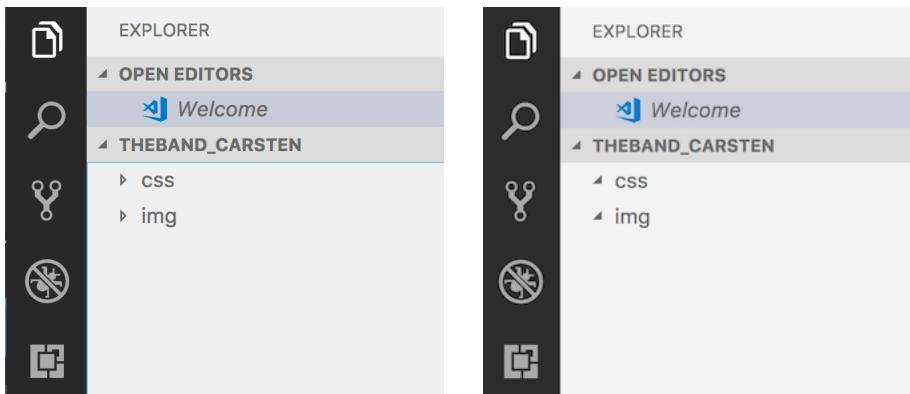


Åben Visual Studio Code ... det kan du gøre fra Windows "Start"- menuen. Visual Studio Code åbner på et velkomst-faneblad som standard. I øverste venstre hjørne fra det område der hedder 'Start' vælger du "Open folder..."

Figur 1 - Velkomst skærbilledet

Find din theband mappe og klik på åben mappe. Et nyt område kommer nu til syne i bruger-interfacet ved siden af den mørkegrå lodrette bar til venstre. Det er en fil-explorer, som giver dig overblik over de åbne filer i din editor samt indholdet af den mappe du i øjeblikket arbejder med. I området 'Open editors' kan du se de filer der i øjeblikket er åbne i programmet. Lige nu er 'Welcome'-fanen stadig åben, så den kan ses i oversigten.

Nedenunder vises indholdet af den mappe du netop har åbnet. Her vises de to mapper css og img. Mapper har små trekantede pile ud for navnet. Man kan klikke på pilen og få vist indholdet af mappen. Lige nu er mapperne tomme, så der kommer ikke noget indhold, men pilen drejer 45° og bliver helt mørkegrå, hvis du klikker på den.

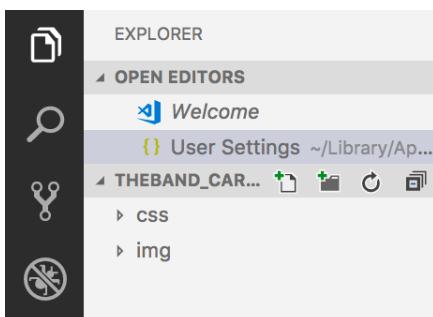


Figur 2 - fil-explorer med lukkede og åbne mapper.

Oprettelse af filer

Du skal nu oprette en fil i roden af projektet. At den ligger i roden refererer til at den er placeret i mappen "theband_ditnavn" og ikke ligger i mappen css eller img.

Hvis du fører musen ind i explorer kolonnen kommer der nogle iconer til synne som repræsenterer forskellige handlinger du kan udføre (Figur 3).



Figur 3 - Mulige handlinger i explorer

Fra venstre mod højre er mulighederne:

Opret fil. Opret mappe. Genopfrisk eller refresh. Luk alle mapper.

Du skal klikke på den første mulighed (den længst mod venstre) som opretter en ny fil. I det blanke felt som nu vises i explorer, skal du give filen det ønskede navn. Filen skal hedde index.html (Figur 4).

Hvorfor index.html

Når man åbner en url i en browser, fx www.mdn.com eller www.w3schools.com, vil den server som behandler din forespørgsel, som standard lede efter en fil der hedder index.html eller

default.html. Derfor er det vigtigt at den fil som genererer det første skærmbillede er navngivet rigtigt.

Fakta! Navngivning af mapper og filer...

Du skal navngive din startside index.html eller default.html. Resten af dine sider kan du kalde hvad du vil, så længe du holder dig til det engelske bogstavsystem og små bogstaver.

Fil og mappenavne må kun indeholde:

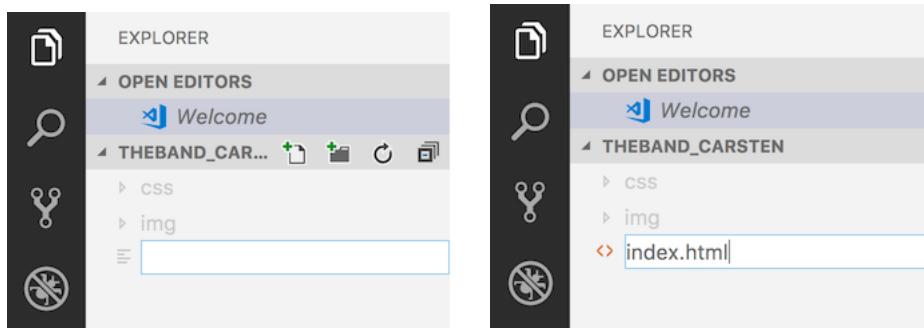
bogstaverne a-z,

tallene 0-9,

bindestreg -

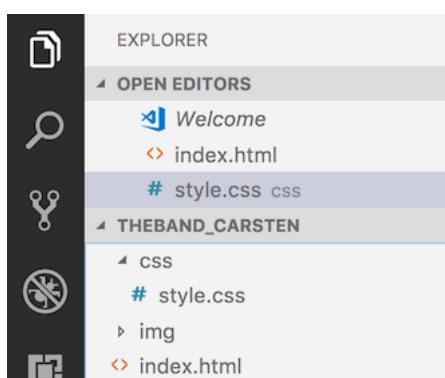
og underscore _

Dine fil- og mappenavne må heller ikke indeholde mellemrum.



Figur 4 - Opret en ny fil og navngiv den index.html ved at skrive navnet i det blanke felt.

Opret herefter filen style.css i mappen css. Du opretter en fil i en mappe ved først at klikke på mappen og derefter ikonet 'opret fil'. Når filen er oprettet, skal din fil-struktur se ud som nedenfor (Figur 5). Læg mærke til, at de nye filer automatisk er åbnet og er klar til redigering under open editors. Læg også mærke til de forskellige ikoner ud for filnavnene. De hjælper dig med at kende forskel på indholdet af de enkelte filer.

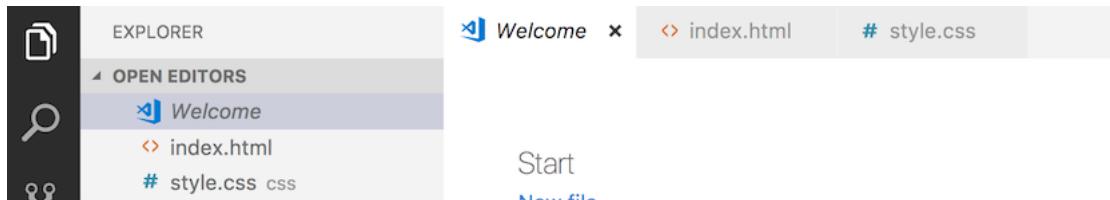


Figur 5 - Filerne index.html og style.css er nu oprettet.

Tip! Opret filer ved hjælp af menu eller tastatur-genvej

Du kan også altid oprette en ny fil ved at vælge menuen file og herefter menupunktet new file. Det opretter en ny blank fil i din editor som ikke er gemt. Den skal du herefter gemme ved at vælge Save as fra menuen filer og så navngive den i denne proces.

Du kan godt lukke Welcome-fanebladet nu. Det gør du ved først at klikke på det, så det bliver aktivt. Der er et sort kryds i højre side af det aktive faneblad. Klik på det for at lukke fanebladet (Figur 6).



Figur 6 - Du kan lukke welcome fanebladet ved at klikke på det, og derefter klikke på det sorte kryds som kommer til syn.

Strukturer i html og css

Html og css er de to kodesprog man bruger til at definere indhold og udseende på en webløsning. Html definerer indholds-strukturen og css bestemmer udseendet af de enkelte elementer.

Fakta! Definitioner

HTML står for Hyper Text Markup Language – det er et kodesprog som bruges til opmarkering af indholdselementer i en webløsning.

CSS står for Cascading Style Sheets – det er et kodesprog som bruges til at manipulere udseendet af en webløsning. Alle browsere har indbyggede stylesheets, som laver en grundformetting af html-elementerne.

Html-tags

De fleste html-tags hører sammen to og to, et start- og et slut tag. HTML tags sidder uden om deres indhold ligesom parenteser i matematik. Slut-tagget har en skråstreg i starten af tagget. Der er nogle få tags, der ikke har et slut-tag, det er bl.a. ****-tagget (til at indsætte billeder med) og **<hr>**-tagget (til at lave en vandret streg i dokumentet med).

```

index.html x # style.css
1  <!DOCTYPE html>
2  <html lang="da"> <!--start html tag-->
3  <head><!--start head tag-->
4    <meta charset="UTF-8">
5    <meta name="viewport" content="width=device-width, initial-scale=1.0">
6    <meta http-equiv="X-UA-Compatible" content="ie=edge">
7    <title><!--start title tag-->Demonstrations-dokument</title><!--slut title tag-->
8 </head><!--slut head tag-->
9 <body><!--start body tag-->
10   
11   <hr>
12
13 </body><!--slut body tag-->
14 </html><!--slut html tag-->
```

Kode eksempel - skal ikke skrives ind

I eksemplet her er der indsatt et billede med ``-tagget. Tagget har fået tilføjet nogle attributter, `src`-attributten fortæller billedstien, `img` er mappens navn, skråstregen viser at man er inde i mappen og `navn.jpg` er selve billedet, `alt`-attributten er vigtig, og har flere formål. Den bliver læst af søgemaskiner, så man kan lave en billedsøgning. Alt-teksten bliver også vist hvis det ikke er muligt at vise billedet og sidst men ikke mindst, bliver alt-teksten læst op af skærmlæsere, som bruges af fx svagtseende.

Alt hvad der kommer til at stå mellem `<body>` og `</body>` er det der kan ses i browser vinduet. Det der står mellem `<title>` og `</title>` er det der kommer til at stå i fanebladet i browseren. Samtidig bliver sidens titel også vist som det første i søgeresultatet når en søgemaskine lister de fundne sider.

Fakta! Søgemaskineoptimering

Søgemaskineoptimering er en vigtig ting i webudvikling. De løsninger du udvikler, skal jo helst være stærke i brugernes bevidsthed. Søgemaskineoptimering handler om at ligge højt i søgemaskinernes resultatliste. Jo højere du ligger, jo større er sandsynligheden for at en bruger vælger din løsning. Der er en række ting man kan gøre for at komme til at ligge højere i søgemaskinernes resultatlister. De ting som er beskrevet ovenfor, er helt centrale, men vi kommer til at udvide begrebet væsentligt i løbet af hele uddannelsen.

Det er vigtigt at sidens titel er beskrivende for indholdet, så brugeren får lyst til at klikke på linket. Billedernes alt-attributter og sidens title-tag er således vigtige for søgemaskineoptimering.

CSS regler

I css-dokumentet (style.css) koder du egenskaberne og udseendet til dine html-elementer. Css står som tidligere nævnt for cascading style sheets. Ligesom i html er der en bestemt struktur man skal overholde i css. CSS er inddelt i regler. En regel består af en selektor, som bestemmer hvilket element i html reglen gælder for. De enkelte html elementer har en række egenskaber fra starten, der definerer hvordan lige netop det element ser ud når man åbner det i browseren. En webudvikler ønsker tit (for ikke at sige altid), at ændre på de standard egenskaber et html element har. I css-reglen skriver man således en egenskab efterfulgt af kolon :, og så den egenskabs ønskede værdi efterfulgt af et semikolon ;. Egenskaber og værdier er omkranset af krølleparenteser {}. Samlet set kunne en css regel se sådan ud:

```
selektor {
    egenskab: værdi;
}
```

Man skriver hver egenskab og dens tilhørende værdi eller værdier på en ny linje, så er det lettere at overskue udviklingsfasen. En css regel for html elementet <body> kan fx se sådan ud:

```
body {
    margin: 0;
    background-color: yellow;
}
```

Typer af selektorer

Der findes forskellige typer af selektorer i css. Html selektorer som vælger de enkelte html elementer. I html kan man dog også tilføje et element attributter i html som du kan bruge som selektorer i css. Det er html-attributten **id** og html-attributten **class**. Attributten **id** benyttes på html elementer som er unikke for dokumentet. I html benytter vi ofte at pakke al sidens indhold ind i et overordnet element. Der er kun ét overordnet element på hver side, så derfor bruger vi ofte at give det element et id. Det kunne se sådan ud i html:

```
<div id="wrapper">
    // her er al sidens indhold
</div>
```

Id'et wrapper er unikt for dette html dokument. Det må kun bruges denne ene gang. Der må gerne være andre id'er på siden, men det enkelte id må kun optræde en gang i hvert dokument. I css tilgår vi id ved at tilføje et hashtag foran id'ets navn. Css reglen for wrapperen kunne se sådan ud:

```
#wrapper {
    // her er egenskaber og værdier for css reglen
}
```

Html attributten **class** kan også bruges til at identificere elementer på siden. For eksempel kunne man forestille sig en række billeder som alle sammen var en del af et galleri. De kunne se sådan ud i html:

```



```

De tre forskellige billeder har alle fået den samme css klasse, og vi kan tilgå dem i css-filen ved at skrive et punktum foran navnet på klassen. Det ser sådan ud:

```
.galleryimg {
    // egenskaber og værdier for galleryimg klassen.
    // Disse egenskaber og værdier for alle elementer
    // som får tilføjet class attributten med værdien
    // "galleryimg".
}
```

Webløsningen til The Band

Lad os komme i gang med at bygge webløsningen til The Band. I denne tutorial starter vi med at skrive css-reglerne til sidens layout. Det giver lidt mere indhold at kigge på, når vi går i gang med at bygge dokumenternes struktur i html. Når du bliver en lidt mere erfaren webudvikler, vil du sikkert skrive din css sideløbende med at du bygger din html struktur op.

Webløsningen til The Band skal kodes i html5. Det betyder at vi har en række html tags til rådighed som kan hjælpe med at øge læsbarheden af dokumentet. Gode eksempler på html5 elementer som øger dokumentets læsbarhed, er elementer som **<header>**, **<nav>**, **<main>**, **<article>**, **<section>** og **<footer>**. Brugen af disse html elementer er med til at inddæle siden i afsnit som er lette at identificere. Det er med til at øge læsbarheden. Et andet ord for læsbarhed er semantik. Det er også det ord vi bruger for en godt opbygget og nemt læsbar html-side. At den er opbygget semantisk korrekt. God semantik er afgørende for dit samarbejde med andre webudviklere, ligesom det er nemmere for søgemaskinernes indeksnings-robotter at adskille de enkelte dele af siden fra hinanden, når siden er opbygget i en semantisk struktur.

Samtidig bygger vi css filen op på en måde som gør, at indholdselementerne bliver nemmere at genbruge på andre webløsninger du skal kode. Man kan tænke på de enkelte komponenter i webløsningen som små moduler, som skal kunne bruges igen i andre webløsninger, så du ikke gang på gang skal kode de samme ting. I webløsningen til The Band er der egentlig kun én komponent i indholdsdelens løsning. Til gengæld er den kodet på en måde som gør den fleksibel. Layoutet er bygget op med to forskellige typer af undersider, men css-koden er skrevet på en måde som kan bruges til begge muligheder. Så uanset om der er ét eller flere billeder på undersiden, kan den samme css bruges i begge tilfælde.

Css filen

Åbn dokumentet style.css og start med at indtaste indholdet som det er vist figur 7. Vi starter css filen med nogle helt overordnede ting som skal rettes til. Det er ting som skrifter og skrift-størrelser mv. Vi vælger hvilken skrift der skal være i dokumentet med css-egenskaben **font-family**. Den sætter vi på html-elementet da det er det yderste element i vores struktur. Det kaldes også for root-elementet. Så kommer det skriftvalg til at gælde for hele dokumentet. På body-elementet sætter vi margin til 0 (nul, ikke lille 0). Den står som standard til 8 pixels, og det giver en lille hvid kant hele vejen rundt, som vi ikke er interesseret i. Bagefter justerer vi lidt på skriftstørrelsen og afstanden i overskriften **h1**. I overskriften **h2** justerer vi også lidt på skriftstørrelsen, mens der i **p**-tagget justeres på linjeafstanden i indholdsteksten.

```

1  html {
2      font-family: Arial, Helvetica, sans-serif;
3  }
4  body {
5      margin: 0;
6  }
7  /* ----- text og font ----- */
8  h1 {
9      font-size: 48px;
10     margin: 0;
11 }
12 h2 {
13     font-size: 36px;
14 }
15 p {
16     line-height: 1.5;
17 }
```

Figur 7 - Indledningen i css dokumentet

Søg efter hjælp!

Selv om du måske ikke føler, at du har så mange forudsætninger for at søge informationer om html og css på nettet, kan du sagtens starte i det små. Et godt sted at starte er www.w3schools.com. Klik på tutorials lige under logoet øverst til venstre og vælg fx css tutorial. Du kan også bruge søgefunktionen ved at klikke på luppen, som er yderst til højre i den grønne bjælke. Husk at du altid kan spørge din sidemand eller læreren om hjælp til at løse opgaven, eller til at få forklaret begreber, som du synes er uklare.

Css – layout

Den næste del af css filen kommer til at styre det overordnede layout. Vi bruger flexbox i footeren, som du kender fra workshoppen. Ud over at justere horisontalt, som du har arbejdet med, justerer vi også vertikalt med egenskaben **align-items**.

```

18  /* ----- layout ----- */
19  header {
20      background-image: url('../img/banner.jpg');
21      height: 240px;
22  }
23  header nav {
24      width: 1184px;
25      height: 100%;
26      margin: 0 auto;
27  }
28
29  main {
30      width: 1184px;
31      margin: 0 auto;
32      padding: 48px 0;
33  }
34  footer {
35      height: 80px;
36      background-image: url('../img/banner_footer.jpg');
37  }
38  footer div {
39      width: 1184px;
40      height: 100%;
41      margin: 0 auto;
42      display: flex;
43      justify-content: space-between;
44      align-items: center;
45  }

```

Figur 8 - layout afsnittet i css filen

Css – navigation

Navigationen defineres i **nav** elementet. De enkelte navigationspunkter bliver skubbet lidt fra hinanden med **margin-left**, fordi de ellers ville stå helt tæt sammen. Links har som standard en streg under sig. Den vil vi gerne fjerne og derfor sættes egenskaben **text-decoration** til none. Links er også blå som standard i html, Det ændrer vi til sort med egenskaben color. Når musen føres hen over de enkelte elementer, skal de ændre udseende med de egenskaber der er beskrevet i reglen der slutter med **:hover**.

```
46  /* ----- navigation ----- */
47  nav ul {
48    padding:0;
49    list-style: none;
50    align-self: flex-end;
51    display: flex;
52    justify-content: flex-end;
53  }
54  nav img {
55    margin: 20px 0;
56  }
57  nav a {
58    font-size: 20px;
59    text-decoration: none;
60  }
61  nav li a {
62    margin-left: 50px;
63    color: black;
64  }
65  nav li a:hover {
66    color: red;
67    text-decoration: underline;
68  }
```

Figur 9 - css til at formattere navigation

Css – main

Article elementet i main elementet kan betragtes som en komponent du kan genbruge senere. Elementet består af en section og en figure. De er sat op ved siden af hinanden ved hjælp af flexbox. I figure elementet er der igen brugt flexbox til at styre de sider der er 4 billeder på. Der er en enkelt regel hvor det måske er lidt svær at forstår hvad der sker. Det er reglen `figure img:nth-last-child(-n+2)`. Stol på den virker, og gør det den skal. Vi vender tilbage til den slags lidt mere komplekse selektorer senere i grundforløbet.

```

69  /* ----- main content ----- */
70  article {
71    display: flex;
72    justify-content: space-between;
73  }
74  article section {
75    width: 40%
76  }
77  article figure {
78    width: 58%;
79    margin: 0;
80    display: flex;
81    justify-content: space-between;
82    flex-wrap: wrap;
83  }
84  figure img {
85    margin-bottom: 15px;
86  }
87  figure img:nth-last-child(-n+2) {
88    margin-bottom: 0;
89  }

```

Figur 10 - css til main-området.

Css – footer

Den sidste afsnit af css-filen indeholder footer-elementet. Der er kun en regel for footeren, og den skal skabe afstand imellem de forskellige logoer i højre side af footeren.

```

90  /* ----- footer content ----- */
91  footer img {
92    margin-left: 32px;
93  }

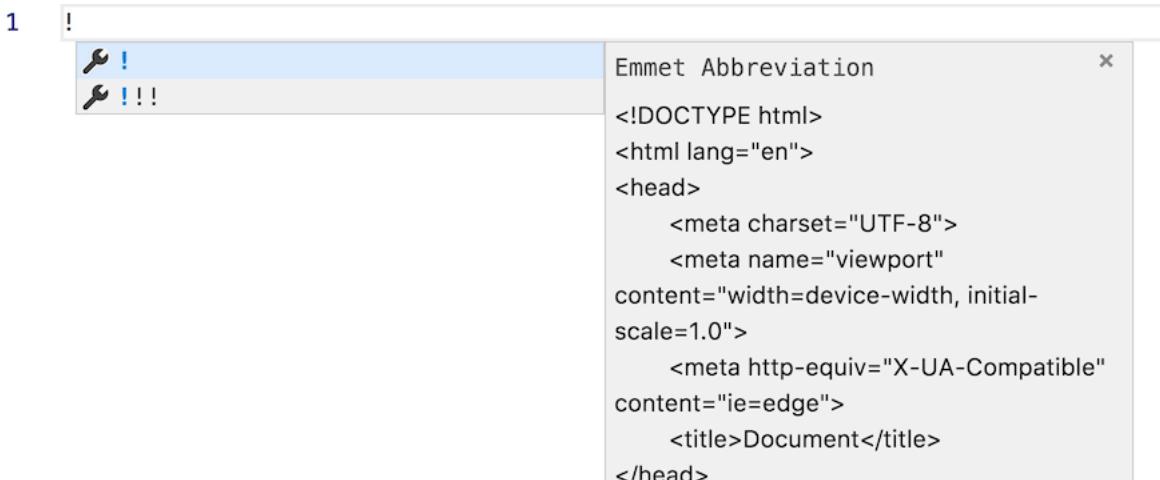
```

Figur 11 - css til footer-området.

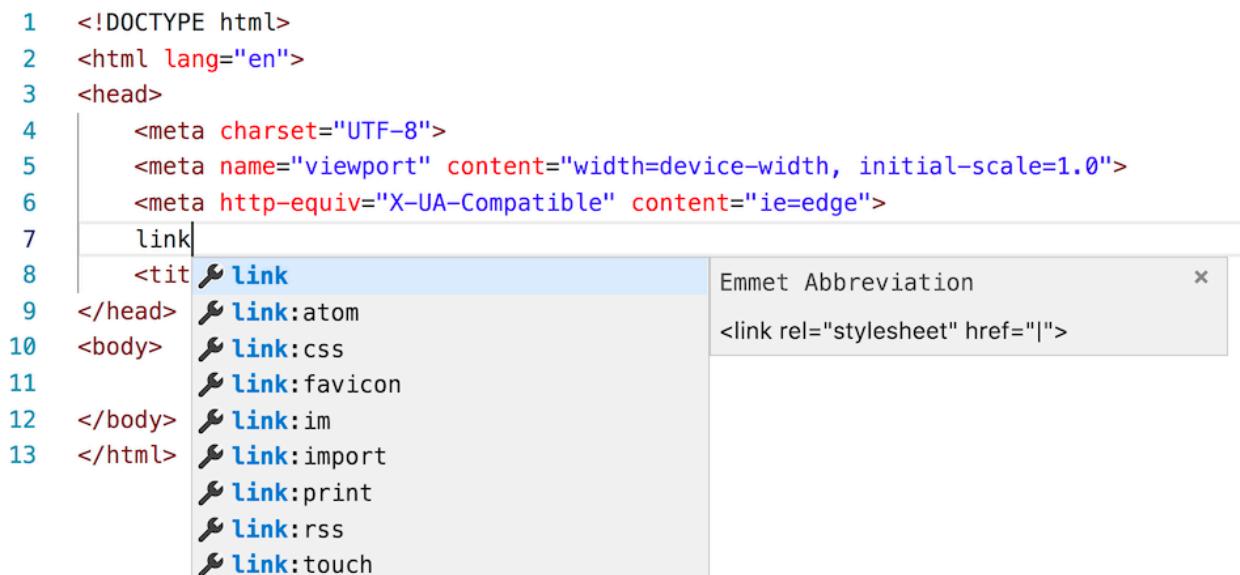
Nu er css filen skrevet færdig. Dermed kan du nu gå i gang med html delen....

Index.html

Du skal nu åbne filen index.html i Visual Studio Code. Det er en helt blank fil. Ligesom da vi startede med style.css. Som det allerførste i index.html skal du taste et udråbststegn. En lille drop-down menu som på billede nedenfor. Tryk herefter på 'enter' eller 'tabulator'.



Der dukker nu 12 linjer html op på skærmen. Det er det grundlæggende html i et nyt dokument. Vi har tidligere gennemgået hovedtrækkene i de 12 linjer så de vil ikke blive gennemgået nærmere her. Du skal nu linke css filen til din index.html. Lav en blank linje i `<head>` delen af dokumentet. Det kan fx være lige før `<title>`-tagget. På den blanke linje skriver du ordet "link". I lighed med før dukker en menu op. Du skal blot vælge den første mulighed som bare hedder "link"



Når du trykker på tabulator eller enter-tasten indsætter Visual Studio Code den linje kode som du skal bruge til at linke din css fil til html filen. Cursoren står klar i attributten href.

```

6 |     <meta http-equiv="X-UA-Compatible" content="IE=edge">
7 |     <link rel="stylesheet" href="">
8 |     <title>Document</title>

```

Vi har lagt css filen vi gerne vil linke til i mappen css. Vi har tidligere installeret den lille extension til Visual Studio Code som hedder 'Path Intellisense'. Den kan hjælpe os nu til at finde den fil vi gerne vil linke til. Tast **ctrl+mellemrum**, og der kommer nu en oversigt til syne over den mappe du har åbnet i Visual Studio Code.

```

5 |     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6 |     <link rel="stylesheet" href="">
7 |     <title>Document</title>
8 |     </head>
9 |     <body>
10 |     <h1>The Band</h1>
11 |

```

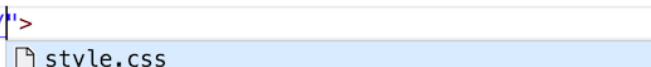


Tryk på **enter** for at vælge mappen **css**. En ny menu dukker op med fil-indholdet af mappen **css**. Tryk på **enter** for at vælge filen **style.css**.

```

7 |     <link rel="stylesheet" href="css/"/>
8 |     <title>Document</title>
9 |     </head>
10 |    <body>
11 |      <h1>The Band</h1>
12 |

```



Du har nu oprettet et link til css-filen i din index.html fil. Det betyder at browseren nu vil bruge de css-regler du har skrevet i din egen fil når den skal vise filen for en bruger. Det færdigt indsatte link til css filen, skal gerne se sådan ud:

```

7 |     <link rel="stylesheet" href="css/style.css"/>
8 |     <title>Document</title>
9 |     </head>
10 |    <body>
11 |      <h1>The Band</h1>
12 |

```

Når du har linket css filen til din html fil kan du begynde at kode selve indholdet til index-siden. Det er en god idé at starte med at kode den del af siden som er ens på både forside og undersider. Så kan du lave kopier af filen til undersiderne. Husk at tilføje titel på dit dokument. På index.html skal titlen være "The Band".

Imellem **<body>** og **</body>** skriver du:

```

8 |     <title>The Band</title>
9 |     </head>
10 |    <body>
11 |      <div id="wrapper">
12 |
13 |        </div> <!-- wrapper slut -->
14 |      </body>
15 |    </html>

```

Tip! Kommentarer i html

Det er vigtigt at du indsætter en kommentar i din kode. Kommentaren er det, som står med grønt ved </div>-tagget. <div> er et universelt tag, som ikke har nogen særlig semantisk betydning. Derfor bliver tagget betragtet som en slags schweizer-kniv i html, altså et tag som kan bruges til lidt af det hele. Derfor kan du godt komme ud for at have flere div-tags som lukker lige efter hinanden. Så er det meget rart, at de enkelte kodelinjer er kommenteret, så man kan se hvilken linje som lukket hvilket tag. I Visual Studio Code indsætter du html-kommentarer med taste-kombinationen shift-alt-a. Du kan selvfølgelig også taste <!--, skrive din kommentar og bagefter taste -->, men det er lettere at buge tastatur-genvejen. Husk at kommentarer er med til at fremme læsbarheden af din kode. Både for dig selv, men også for andre webudviklere.

I mellem wrapper start og wrapper slut indsætter du de tre overordnede elementer, header, main og footer, på siden (se næste side). De skal lave den overordnede inddeling af siden. De er alle tre html5 elementer. De har forskellige funktioner. Tagget header definerer det område som er det indledende område på siden. Indholdet i header-tagget vil typisk være det samme på alle undersider i en webløsning. Det er med til at gøre brugeroplevelsen ensartet. Efter headeren kommer main-tagget. Det er her det indhold som er selve sidens formål præsenteres. Indholdet af main-tagget er i de fleste tilfælde unikt. Indholdet i footer-tagget er typisk også ens på alle siderne i webløsningen. Her vil du typisk finde oplysninger om afsenderen, og i vores tilfælde links til forskellige sociale medier.

```

11   <div id="wrapper">
12     <header>
13
14   </header>
15
16   <main>
17
18   </main>
19
20   <footer>
21
22   </footer>
23 </div> <!-- wrapper slut -->
```

Header

Inde i header-tagget skal du indsætte et <nav> tag. I denne løsning bruger vi headeren til navigation. Vi har lavet baggrund på headeren. Med css har vi begrænset bredden på nav-elementet med css-egenskaben width. Nav elementet er også centreret på skærmen. Det er css-egenskaben margin der centererer. Margin har 2 værdier. Først værdien 0 (nul). Det

betyder at der ikke skal være noget margin i toppen og i bunden af nav-elementet. Herefter kommer værdien 'auto'. Den fordeler den overskydende luft ved siden af elementet med lige meget på hver side, så nav-elementet kommer til at stå i midten af skærmen. Inde i <nav> elementet skal du først indsætte The Band logoet. Herefter laver vi en liste med alle menu-elementerne i. Lister er en måde at organisere opremsninger på. I html findes der forskellige typer af lister. De to mest brugte er uordnede lister og ordnede lister. I de uordnede lister starter hvert listepunkt med en prik. I ordnede lister starter hvert listepunkt med et tal, startene med tallet 1 for det første punkt, 2 for det næste punkt osv.

```

12   <header>
13     <nav>
14       
15       <ul>
16         <li><a href="#">Medlemmer</a></li>
17         <li><a href="#">Live</a></li>
18         <li><a href="#">Merchandise</a></li>
19         <li><a href="#">Info</a></li>
20       </ul>
21     </nav>
22   </header>
```

Ekstra! Lister

Prøv at oprette nyt html dokument og opret en ordnet og en uordnet liste med et par listepunkter i hver, og se forskellen i browseren. Husk at indsætte html-grundstrukturen i det nye dokument med udråbstegn (!).

Hvis du prøver at vise siden i broswseren (alt+B) burde siden se nogenlunde sådan ud:



Logoet er lagt ind i navigationsområdet, men lige nu er det ikke et link. De fleste brugere forventer at komme tilbage til forsiden når de klikker på logoet.

Indsæt et link rundt om logoet der fører brugeren tilbage til index.html.

Main

I main-området skal du kun gøre klar til at indsætte indhold. Main kommer til at bestå af en article som er delt op i to under-elementer. En section, som er der hvor al teksten skal stå, og en figure til at indsætte billederne i.

```

26   <main>
27     <article>
28       <section>
29
30       </section>
31       <figure>
32
33       </figure>
34     </article>
35   </main>
```

Footer

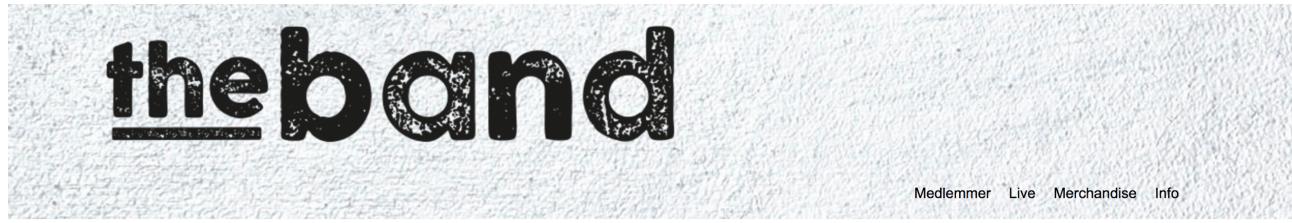
Nu mangler du kun at lave footeren, og så er skabelonen til The Band færdig. I footeren skal der være en div, som skal begrænse bredden. Et booking-telefonnummer og et lille område med links til forskellige sociale medier.

```

37   <footer>
38     <div>
39       <h2>Booking: +45 2143 6587</h2>
40       <nav class="footernav">
41         
42         
43         
44         
45       </nav>
46     </div>
47   </footer>
```

Indsæt links på logoerne i footernav'en. Logoet skal linke til index.html. De andre ikoner skal linke til forsiderne på de tre services. De tre eksterne links skal selvfølgelig åbne i en ny fane i browseren. Kan du huske hvordan?

Nu er grundskabelonen til siderne færdig. Hvis du viser den i browseren skal den se ud som på billedet nedenfor.



Navigation

Inden du opretter alle filerne til undersiderne kan du med fordel lige opdatere navigationen i headeren

```

17      <ul>
18          <li><a href="medlemmer.html" alt="Medlemmer">Medlemmer</a></li>
19          <li><a href="live.html" alt="Live">Live</a></li>
20          <li><a href="merchandise.html" alt="Merchandise">Merchandise</a></li>
21          <li><a href="info.html" alt="Info">Info</a></li>
22      </ul>
```

Du kan nu oprette filerne til undersiderne. Lav kopier af index.html ved at vælge 'Save as' fra 'File'-menuen. Du skal oprette filerne medlemmer.html, live.html, merchandise.html og info.html.

Indholdet på index.html

Hent teksten til forsiden i tekst-filen i opgavemappen til The Band. Filen hedder 'The Band Tekster.txt'. Indsæt teksten i filen index.html i det tomme section-tag som ligger inde i article tagget. Indsæt h1- og p-tags som vist.

```

27  <article>
28      <section>
29          <h1>The Band spiller op</h1>
30          <p>The Band er et rock band med base i det århusianske musik-miljø
31          som spiller koncerter i hele skandinavien. </p>
32          <p>Band blev etableret i 2007 som ungdomsband i Århus. Vi har
33          udgivet to full-length albums og 5 ep'er. Vi udgav vores første album
34          i 2013. I 2015 fulgte album nummer to. Det er vores ambition at
              udgive en ep hvert halve år. </p>
35          <p>Vores seneste ep udkom i starten af 2018. Vi elsker at spille live
              og vi glæder os til at se jer til en af efterårets koncerter. </p>
36      </section>
37      <figure>
```

Billedet af 'The Band' skal du indsætte i figure-tagget. Figure er et html5 element som bruges til at opmarkere illustrationer med. Husk at tilføje en alt-tekst til billedet.

```
55 |           </section>
34 |           <figure>
35 |             
36 |           </figure>
37 |       </article>
```

Den færdige index.html skal se sådan ud i browseren:



The Band spiller op

The Band er et rock band med base i det århusianske musik-miljø som spiller koncerter i hele skandinavien.

Band blev etableret i 2007 som ungdomsband i Århus. Vi har udgivet to full-length albums og 5 ep'er. Vi udgav vores første album i 2013. I 2015 fulgte album nummer to. Det er vores ambition at udgive en ep hvert halve år.

Vores seneste ep udkom i starten af 2018. Vi elsker at spille live og vi glæder os til at se jer til en af efterårets koncerter.



Booking: +45 2143 6587

theband

Undersiderne

Undersiden medlemmer.html er bygget op på samme måde som index.html. Strukturen i teksten er dog lidt anderledes, da de fire bandmedlemmers navne er fremhævet i forbindelse med den tekst der beskriver dem hver især. Du kan her overveje om det skal være en fremhævelse af teksten med fed skrift, eller navnene måske er en overskrift. Det bør indgå i dine overvejelser at der på siden live.html også er en fremhævet tekst, som indleder bandets tour-plan. Der er også fire billeder på denne side i stedet for et. Den css du tidligere har skrevet, er forberedt til fire billeder. Heldigvis må der også gerne være mere end et billede inde i et figure tag, så på medlemmer.html bør dit figure-tag se sådan ud:

```

34 |     <figure>
35 |       
36 |       
37 |       
38 |       
39 |   </figure>

```

Filen live.html er bygget op på samme måde som index.html. Husk at være konsekvent i forhold til dit valg omkring de fremhævede tekster på siden medlemmer.html. Filen merchandise.html ligner medlemmer.html.

Ekstraopgave A

Det er beskrevet tidligere i denne tutorial hvordan indholdet i main-området kan betragtes som en komponent der kan genbruges. Tilføj en class-attribut på dit article-tag og tilret css-filen så *alle* reglerne der relaterer til komponenten, starter med det navn du valgte til din class. Så kan du nemmere genbruge komponenten i et senere projekt.

Ekstraopgave B

Den sidste side, omos.html er der ikke noget planlagt indhold til. Du kan, hvis du er blevet hurtigt færdig, selv finde på noget indhold til denne side. Siden behøver ikke nødvendigvis hedde 'Om os' hvis du synes der er noget andet der passer bedre. Det kan være en side med fan-historier eller en side med bandets udgivelser. Altså navne på albums og ep'er. Måske med billeder af covers. Her er der altså plads til at være lidt kreativ.

Hvis du har brug for at ændre på det visuelle udseende af siden, kan du jo tilføje noget css så det kommer til at passe. Hvis siden ender med at have et markant anderledes layout, bør du aktivt overveje om det er en helt ny komponent du er i gang med at skabe. Hvordan kan du gøre denne komponent brugbar i et senere projekt?

Layout – Index

the band

Medlemmer Live Merchandise Info

The Band spiller op

The Band er et rock band med base i det århusianske musik-miljø som spiller koncerter i hele skandinavien.

The Band blev etableret i 2007 som ungdomsband i Aarhus. Vi har udgivet to full-length albums og 5 ep'er. Vi udgav vores første album i 2013. I 2015 fulgte album nummer to. Det er vores ambition at udgive en ep hvert halve år.

Vores seneste ep udkom i starten af 2018. Vi elsker at spille live og vi glæder os til at se jer til en af efterårets koncerter.



Booking: +45 2143 6587

the band

Layout – Medlemmer

the band

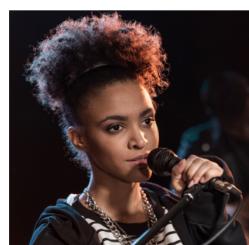
Medlemmer Live Merchandise Info

Medlemmer

The Band består fortsat af de fire oprindelige medlemmer. Vi har spillet sammen i mere end ti år og det betyder at vi kender hinanden rigtig godt. Det er en fordel både i studiet og på scenen at vi kender hinanden små finurligheder.

Alexander

Alexander er vores bassist og lidt af en practical joker. Så hvis kaffen smager salt eller mælken er blå, så er det stensikkert Alexander der har været på spil.



Yasmin

Yasmin spiller trommer på lille del af vores numre og er sanger på de andre. Yasmin har altid sunget og hendes stemme er næsten synonym med The Band.



Kristine

Kristine er gruppens guitarist. Det er også Kirstine som synger kor på de fleste af vores studieindspligner. Kristine er også gruppens barista - hun ved bare alt om kaffe!

Johan

Johan er gruppens trommeslager. Han stjæler dog af og til mikrofonen fra Yasmin, når de allerlest rockede numre kræver hans rå vokal.

Booking: +45 2143 6587

the band

Layout – Live

[Medlemmer](#) [Live](#) [Merchandise](#) [Info](#)

The band - Live

Musik skal opleves, og allerhelst live. Vi er booket til at spille koncerter en række steder rundt omkring i Danmark i den kommende tid. Vi planlægger en skandinavisk turné i foråret 2019. Vi offentliggør løbende datoerne i løbet af efteråret. Følg med her på hjemmesiden eller på vores facebook-side.

Bookede koncerter

Roskilde hallerne - 26 september 2018
Kolding Musikhus - 4 oktober 2018
Aalborg Koncertus - 23 oktober 2018
Tobakken, Esbjerg - 12 november 2018
Ringsted Kongrescenter - 27 november 2018
Tømmerup Hallen, Kalundborg - 12 december 2018



Booking: +45 2143 6587

[theband](#)

Layout – Merchandise

[Medlemmer](#) [Live](#) [Merchandise](#) [Info](#)

Merchandise

The Band merchandise sælges eksklusivt til vores live-koncerter. Vi har et bredt sortiment af merchandise så der burde være noget som falder i enhver superfans smag.

Her på siden kan du se et lille udvalg af de varer der er i vores merchandise sortiment. Hvis du har en god ide til vores merchandise butik, er du altid velkommen til at lave et opslag på vores Facebook side.



Booking: +45 2143 6587

[theband](#)