

Eksamensdokument

Gruppe 1

Vanskelighetsgrad

Vi har valgt å gå for vanskelighetsgrad A.

Git-Hub alias

Anders = Lubbsaur

Marie = MarieIT

Marius = MaRoelT

Mathias = matkri3110

Lenke til GitHub

https://github.com/MarieIT/UIN_Eksamensprosjekt_Gruppe1

Brukernavn/passord brukere:

Johnny Johnson:

Brukernavn: JohnJ

Passord: BodoGlimtFan55

Johanna Fjell:

Brukernavn: JoFjell

Passord: nIAhDOjdt385

Frøya Liljekonvall:

Brukernavn: Liljer

Passord: ABCILiljer123

Alexander Hakito:

Brukernavn: MetalHead39

Passord: MH2023

Maria Berg:

Brukernavn: mariab29

Passord: 29bmaria

Redegjørelser/forutsetninger

Hvis vi hadde hatt mer tid, ville vi gjerne ha jobbet litt mer med styling og det visuelle. Siden er forholdsvis responsiv og følger stort sett krav om universell utforming, men det mangler fortsatt en del i den avdelingen.

Vi har hatt en del utfordringer med asynkron fetching av Ticketmaster API-et, og hvordan få mappet ut forskjellige ting konsekvent i de forskjellige komponentene. Blant annet er det ikke så fleksibelt hvis vi forsøker å fetche ut flere ganger i løpet av kort tid.

Vi har forsøkt å gjøre så mye som mulig uten hjelp fra AI, noe som har medført mye frustrasjon og hodekløring for enkelte av gruppemedlemmene. Men dette gjør jo også at vi har lært mye mens vi har jobbet med prosjektet. I de tilfellene vi har spurt AI, har vi som regel fått ikke så gode svar, som senere gjerne har blitt erstattet med andre løsninger.

Forskjellige personer i gruppen har sine styrker og svakheter, så det er fint å kunne lene seg på hverandre. Det er vel det som er fordelene med å jobbe i team. I den forbindelse vil alle gruppemedlemmene si seg meget fornøyd med innsats og samarbeid til den enkelte.

Kilder:

AI-samtale med GPT UiO:

https://drive.google.com/file/d/1Q6n_gp5Zt5rqmEoye_YTq1wt_c4uEDoh/view?usp=sharing

Balogh, M (July 27, 2015). *Bilde med Young, Jente, Solbriller. Gratis å bruke.* [Fotografi]. Fra Pixabay.

<https://pixabay.com/no/photos/young-jente-solbriller-skj%C3%B8nnhet-858482/>

CSS-arrow. How To Create Arrows/Triangles:

https://www.w3schools.com/howto/howto_css_arrows.asp

Dung, C (April 4, 2023). *Bilde med Kvinne, Student, Bok. Gratis å bruke* [Fotografi].
Fra Pixabay.

<https://pixabay.com/no/photos/kvinne-student-bok-campus-skole-7882523/>

Filter-funksjon:

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

Guide til Hamburger-meny:

<https://khuang159.medium.com/creating-a-hamburger-menu-in-react-f22e5ae442cb>

Hamburger-icon. Fra Google Fonts, Material Symbols and Icons

https://fonts.googleapis.com/css2?family=Material+Symbols+Outlined:opsz,wght,FILL,GRAD@24,400,0,0&icon_names=menu

OpenAI. (2025, 27. mars). ChatGPT (GPT 4o) [Stor språkmodell]

<https://openai.com/index/hello-gpt-4o/>

Senyo, L (September 21, 2022). *Bilde med Student, Kvinne, Smil. Gratis å bruke*.
[Fotografi]. Fra Pixabay.

<https://pixabay.com/no/photos/student-kvinne-smil-portrett-skole-7468801/>

thelester (January 7, 2016). *Bilde med Mann, Portrett, Skjegg. Gratis å bruke*.
[Fotografi]. Fra Pixabay.

<https://pixabay.com/no/photos/mann-portrett-skjegg-stilig-1122364/>

Uten fotograf (September 18, 2014). *Bilde med Man, Mann, En person. Gratis å bruke*.
[Fotografi]. Fra Pixabay.

<https://pixabay.com/no/photos/man-mann-%C3%A9n-person-lykke-449406/>

Samtale med ChatGPT, om styling av forms på categorypage:

Prompt:

Du jeg prøver nå å style noen forms, nå har jeg stylet det for mobile first, men når jeg prøver å få input og søkefeltene under hverandre med display flex på klassen de ligger i så blir pluteslig inputfeltene like store som browser width? Hva kan dette skyldes. Her er koden min med sass:

```
@use "./partials/colors";
```

```
@use "./partials/fonts.scss";
```

```
#root {
```

```
  margin: 0;
```

```
  padding: 0;
```

```
}
```

```
main {
```

```
  article {
```

```
    h3 {
```

```
      max-width: 300px;
```

```
    }
```

```
    img {
```

```
      width: 400px;
```

```
    }
```

```
}
```

```
#filtercategory {
```

```
  display: flex;
```

```
  flex-direction: column;
```

```
  margin: 2rem;
```

```
  label {
```

```
    font-size: 1rem;
```

```
padding: 0.2rem;

text-align: center;

}
```

```
input {

background-color: rgb(250, 250, 250);

margin: 0.5rem;

padding: 0.5rem;

align-items: center;

}
```

```
select#countries {

background-color: rgb(250, 250, 250);

margin: 0.5rem;

padding: 0.5rem;

}
```

```
select#byer {

background-color: rgb(250, 250, 250);

margin: 0.5rem;

padding: 0.5rem;

}
```

```
button {

margin: 2rem;

color: colors.$whitetext;

background-color: colors.$buttons;
```

```
padding: 0.5rem 0.5rem;

margin-left: 7rem;

margin-right: 7rem;

font-size: 1rem;

text-decoration: none;

width: 100PX;

height: 50px;

}

}
```

```
#searchform {

display: flex;

flex-direction: column;

margin: 2rem;
```

```
label {

font-size: 1rem;

padding: 0.2rem;

text-align: center;

}
```

```
input {

background-color: rgb(250, 250, 250);

margin: 0.5rem;

padding: 0.5rem;
```

```
    align-items: center;
}
```

```
select {
    background-color: rgb(250, 250, 250);
    margin: 0.5rem;
    padding: 0.5rem;
}
```

```
button {
    margin: 1.5rem;
    align-items: center;
    color: colors.$whitetext;
    background-color: colors.$buttons;
    padding: 0.5rem 0.5rem;
    font-size: 1rem;
    text-decoration: none;
    width: 100PX;
    height: 50px;
}
}
```

```
#venuesection {
    display: flex;
    flex-direction: column;
    gap: 0.5rem;
```

```
margin: 0 auto;

padding: 0 20px;

h3 {

    font-family: fonts.$quantitytext;

    font-size: 1rem;

    margin: 0 auto;

    padding: 0 10px;

}

}

}

/* Tablet: */

@media (min-width: 481px) and (max-width: 768px) {}

/* Liten desktop */

@media (min-width: 769px) and (max-width: 1024px) {

.filter-search {

    display: flex;

    flex-direction: column;

    #filtercategory {

        display: flex;

        flex-direction: column;

        margin: 2rem;

        label {

            font-size: 1rem;

            padding: 0.2rem;
```



```
text-align: center;
}
```

```
input {
  background-color: rgb(250, 250, 250);
  margin: 0.5rem;
  padding: 0.5rem;
  align-items: center;
}
```

```
select#countries {
  display: flex;
  justify-content: center;
  background-color: rgb(250, 250, 250);
  margin: 0.5rem;
  padding: 0.5rem;
  max-width: 300px;
}
```

```
select#byer {
  background-color: rgb(250, 250, 250);
  margin: 0.5rem;
  padding: 0.5rem;
}
```

```

    button {
        margin: 2rem;
        color: colors.$whitetext;
        background-color: colors.$buttons;
        padding: 0.5rem 0.5rem;
        margin-left: 7rem;
        margin-right: 7rem;
        font-size: 1rem;
        text-decoration: none;
        width: 100PX;
    }
}

}

/* Stor desktop */

@media (min-width: 1025px) {
    .filter-search {
        display: flex;
        flex-direction: column;
    }
}

}

```

Svar:

🔥 Hva som sannsynligvis skjer:

Når du bruker `display: flex` og `flex-direction: column` på `#filtercategory` og `#searchform`, og **ikke setter en `max-width` eller `width` på input-feltene**, vil de som standard ta opp **hele bredden av parent-elementet** – og hvis parenten ikke har noen begrensning, så strekker det seg til 100vw.

✅ Løsning:

🔧 Legg til `max-width` eller `width` på `input`, `select` og `form-container`

Eksempel rett i din eksisterende kode:

Legg til dette i både `#filtercategory` og `#searchform`:

```
scss
#filtercategory,
#searchform {
  max-width: 500px;
  margin: 0 auto;
  width: 100%;
}
```

Og legg til begrensning på `input/select`:

```
scss
input,
select {
  width: 100%;
  max-width: 100%;
  box-sizing: border-box; // Viktig så padding ikke øker bredden
}
```

🔥 Forklart visuelt:

Element	Effekt uten begrensning	Effekt med <code>max-width</code>
<code>input</code>	Blir like bred som browser	Blir maks 100% av 500px container
<code>form</code>	Tar opp full skjermbredde	Sentrert og begrenset i bredde

🔧 Bonus: Unngå “`margin-left/right: 7rem`” på knappene

Hvis du vil sentrere knappene, bruk:

```
scss
button {
  margin: 2rem auto;
}
```

🌟 Forslag til forbedret snippet:

scss

Copy

```
#filtercategory,
#searchform {
  max-width: 500px;
  margin: 2rem auto;
  display: flex;
  flex-direction: column;
  width: 100%;

  input,
  select {
    width: 100%;
    max-width: 100%;
    padding: 0.5rem;
    margin: 0.5rem 0;
    background-color: rgb(250, 250, 250);
    box-sizing: border-box;
  }

  button {
    width: 120px;
    height: 50px;
    margin: 2rem auto;
    background-color: colors.$buttons;
    color: colors.$whitetext;
    font-size: 1rem;
  }
}
```

Vil du at jeg hjelper deg med å gjøre filter-search til en grid på desktop også, så alt ligger pent ved siden av hverandre?

Refleksjon:

Jeg prøvde å implementere de foreslåtte endringene, men problemet jeg hadde dukket fortsatt opp. Så disse endringene ble ikke brukt.

Prompt: (Media queries, feilsøking)

Når jeg går over 1024px til siste media query så funker ikke flexen min lenger, ting blir gjort om til row og ikke coulumn

Svar:

! Problemet:

Du har denne media queryen:

scss

Copy

```
/* Stor desktop */
@media (min-width: 1025px) {
  .filter-search {
    display: flex;
    flex-direction: column;
  }
}
```

Men *inni* `.filter-search` ligger f.eks. `#filtercategory`, `#searchform` osv., og hvis **de ikke har eksplisitt `flex-direction: column`**, vil de **arve layout fra `display: flex` på `.filter-search`**, og da får du **flex-row som default** 😞

✅ Løsning:

Sørg for at **alle containerne** (`#filtercategory`, `#searchform`, osv.) har sin egen `flex-direction: column` spesifisert:

scss

Copy

```
@media (min-width: 1025px) {
  .filter-search {
    display: flex;
    flex-direction: column; // for å sikre vertikal layout

    #filtercategory,
    #searchform {
      display: flex;
      flex-direction: column; // ← viktig!
      max-width: 500px;
      margin: 0 auto;
    }
  }
}
```

🔪 Hvorfor skjer dette?

`display: flex` alene = default `flex-direction: row`

Så hvis du setter `display: flex` på `.filter-search`, og barna inni (som `#filtercategory`) ikke spesifiserer sin egen `flex-direction`, så får de layout i row dersom du legger mange inputs i samme linje.

✨ Ekstra tips:

Hvis du **vil at bare forelderen** skal være flex, og barna (som `form`, `section`, `div`) skal være kolonner, så må du alltid sette `flex-direction: column` eksplisitt per nivå.

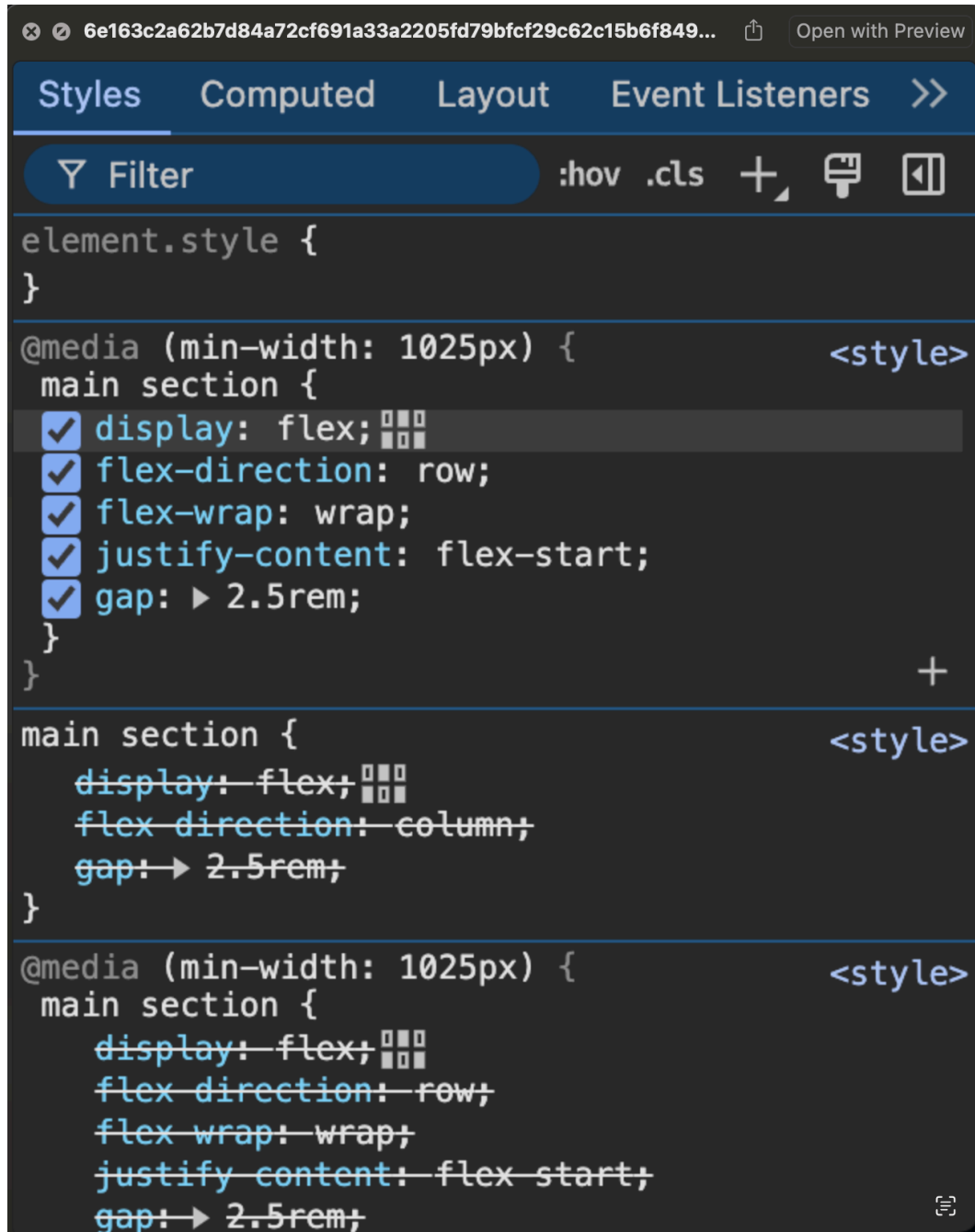
Vil du ha hjelp til å lage en layout der f.eks. **venstre side har filtrering og høyre side har søk**, men mobilversjonen fortsatt er stacked (column)? Da kan vi bruke grid eller flex med media queries 💖

📄 🔊 🍏 🗑️ ↻

Refleksjon:

Når det nærmet seg tid for å skrive media queries for categorypage, slik at vi siden skulle bli responsive så møtte jeg på noen problemer. Problemet viste seg å være at tidligere skrevet kode ble arvet og dermed overskrev koden jeg skrev i media queries seksjonen. Så her valgte jeg å høre med ChatGPT om den kunne finne ut av problemet.

Problemet rundt arvede sass egenskaper dukket fremdeles opp. GPT kom ingen vei, så derfor valgte jeg å gi den mer kontekst. Så jeg gav nesten alle sass-filene jeg tenkte problemet kunne komme fra. I tillegg til et bilde av dev-tools i inspector:



Prompt:

Gir mer kontekst:

Dette er de filene vi har:

@use "../partials/colors";

```
@use "../partials/fonts.scss";

//Mobile-first

#root {
    margin: 0, auto;
    width: 100%;
    font-family: fonts.$quantitytext;
}

main {
    display: flex;
    flex-direction: column;
    justify-content: center;
    align-items: center;
    margin: 3rem auto;
    padding: 0 20px;
    h1 {
        font-family: fonts.$titles;
        font-size: 2.3rem;
        margin: 0 auto;
        padding: 0 20px;
    }
    section {
        display: flex;
        flex-direction: column;
        gap: 2.5rem;
        article {
            display: flex;
            flex-direction: column;
            margin: 1rem 1rem;
        }
    }
}
```

```
padding: 0 20px;
```

```
gap: 1rem;
```

```
img {
```

```
  width: 300px;
```

```
  height: 200px;
```

```
}
```

```
.mainEventBtn {
```

```
  max-width: 300px;
```

```
  color: colors.$whitetext;
```

```
  background-color: colors.$buttons;
```

```
  padding: .5rem 1rem;
```

```
  border-radius: 2px;
```

```
  text-decoration: none;
```

```
  width: auto;
```

```
  height: 25px;
```

```
  text-align: center;
```

```
  display: inline-block;
```

```
}
```

```
}
```

```
}
```

```
.by-knapper {
```

```
  display: flex;
```

```
  flex-direction: column;
```

```
  align-items: center;
```



```
padding: 3rem 0 2rem 0;

article {

display: flex;

flex-direction: row;

flex-wrap: wrap;

  a {

    text-decoration: none;

    color: colors.$whitetext;

    background-color: colors.$buttons;

    padding: .7rem 1.2rem;

    border-radius: 2px;

    text-decoration: none;

    width: auto;

    height: 25px;

  }

}

}

.artikkel-fra-byer {
```

```
article{

  border: 1px solid black;

  width: 300px;

  padding: 1rem;

  h3{

    min-height: 46px;

  }

  p{

    line-height: 5px;
```

```

        }
    }
}

}

body {
    background-color: colors.$backgroundcolor;
}

/* Tablet: */

@media (min-width: 481px) and (max-width: 768px) {
}

/* Liten desktop */

@media (min-width: 769px) and (max-width: 1024px) {
    main {
        display: block;

        h1 {
            font-family: fonts.$titles;

            font-size: 2rem;

            margin: 0 auto;

            padding: 0 20px;
        }

        section {
            display: flex;

            flex-direction: row;

            flex-wrap: wrap;

            justify-content: space-between;

            gap: 2.5rem;

            .mainEventBtn {

```

```

        max-width: 300px;

        color: colors.$whitetext;

        background-color: colors.$buttons;

        padding: .7rem 1.2rem;

        border-radius: 2px;

        text-decoration: none;

        height: 25px;

        display: inline-block;
    }
}
}
}

/* Stor desktop */
@media (min-width: 1025px) {
    main {
        display: block;

        h1 {
            font-family: fonts.$titles;

            font-size: 2rem;

            margin: 0 auto;

            padding: 0 20px;
        }

        section {
            display: flex;

            flex-direction: row;

            flex-wrap: wrap;

            justify-content: flex-start;

            gap: 2.5rem;

```

```
    }  
  }  
}  
  
@use "../partials/colors";  
@use "../partials/fonts.scss";  
  
#root {  
  margin: 0;  
  padding: 0;  
}  
  
main {  
  article {  
    h3 {  
      max-width: 300px;  
    }  
    img {  
      width: 400px;  
    }  
  }  
  
  #filtercategory {  
    display: flex;  
    flex-direction: column;  
    margin: 2rem;  
  
    label {  
      font-size: 1rem;  
      padding: 0.2rem;  
      text-align: center;  
    }  
  }  
}
```

```
input {  
  background-color: rgb(250, 250, 250);  
  margin: 0.5rem;  
  padding: 0.5rem;  
  align-items: center;  
}  
  
select#countries {  
  background-color: rgb(250, 250, 250);  
  margin: 0.5rem;  
  padding: 0.5rem;  
}  
  
select#byer {  
  background-color: rgb(250, 250, 250);  
  margin: 0.5rem;  
  padding: 0.5rem;  
}
```

```
button {  
  margin: 2rem;  
  color: colors.$whitetext;  
  background-color: colors.$buttons;  
  padding: 0.5rem 0.5rem;  
  margin-left: 7rem;  
  margin-right: 7rem;  
  font-size: 1rem;  
  text-decoration: none;  
  width: 100PX;
```

```
    height: 50px;
  }
}

#searchform {
  display: flex;
  flex-direction: column;
  margin: 0 auto;

  label {
    font-size: 1rem;
    padding: 0.2rem;
    text-align: center;
  }

  input {
    background-color: rgb(250, 250, 250);
    margin: 0.5rem;
    padding: 0.5rem;
    align-items: center;
  }

  select {
    background-color: rgb(250, 250, 250);
    margin: 0.5rem;
    padding: 0.5rem;
  }

  button {
```

```

    margin: 1.5rem;
    align-items: center;
    color: colors.$whitetext;
    background-color: colors.$buttons;
    padding: 0.5rem 0.5rem;
    margin-left: 7rem;
    margin-right: 7rem;
    font-size: 1rem;
    text-decoration: none;
    width: 100PX;
    height: 50px;
  }
}

#venuesection {
  display: flex;
  flex-direction: column;
  gap: 0.5rem;
  margin: 0 auto;
  padding: 0 20px;
  h3 {
    font-family: fonts.$quantitytext;
    font-size: 1rem;
    margin: 0 auto;
    padding: 0 10px;
  }
}
}

/* Tablet: */

```

```
@media (min-width: 481px) and (max-width: 768px) {}

/* Liten desktop */

@media (min-width: 769px) and (max-width: 1024px) {
.filter-search {
    display: flex;
    flex-direction: column;

    #filtercategory {
        display: flex;
        flex-direction: column;
        margin: 0 auto;
    }

    #searchform {
        display: flex;
        flex-direction: column;
        margin: 0 auto;
    }
}

/* Stor desktop */

@media (min-width: 1025px) {
section.filter-search {
    display: flex;
    flex-direction: column;

    #filtercategory {
        display: flex;
        flex-direction: column;
        margin: 0 auto;
    }
}
```



```
#searchform {  
  display: flex;  
  flex-direction: column;  
  margin: 0 auto;  
}  
}  
}  
}  
  
@use "../partials/colors";  
@use "../partials/fonts.scss";  
  
//Mobile-first  
  
#root {  
  margin: 0, auto;  
  width: 100%;  
  font-family: fonts.$quantitytext;  
}  
  
main {  
  display: flex;  
  flex-direction: column;  
  justify-content: center;  
  align-items: center;  
  margin: 3rem auto;  
  padding: 0 20px;  
  h1 {  
    font-family: fonts.$titles;  
    font-size: 2.3rem;  
    margin: 0 auto;  
    padding: 0 20px;
```

```
}  
  
h2{  
    margin-top: 1rem;  
}  
  
#infosection {  
    display: flex;  
    flex-direction: column;  
    gap: 0.5rem;  
    h3{  
        font-family: fonts.$quantitytext;  
        font-size: 1rem;  
        margin: 0 auto;  
        margin-top: 1rem;  
        padding: 0 20px;  
    }  
    ul {  
        display: flex;  
        flex-direction: column;  
        gap: 0.5rem;  
        margin: 0 auto;  
        padding: 0 20px;  
        li {  
            font-family: fonts.$quantitytext;  
            font-size: 1rem;  
            margin: 0 auto;  
            padding: 0.5rem;  
            list-style-type: none;  
            a {
```

```

        color: colors.$header;
    }
}
}
article {
    display: flex;
    flex-direction: column;
    margin: 0.5rem 0.5rem;
    padding: 0 20px;
    img {
        width: 300px;
        height: 200px;
        margin: 1rem;
    }
}
}
}
#festivalinfo {
    article {
        p{
            font-family: fonts.$quantitytext;
            font-size: 1rem;
            margin: 0;
        }
        h3 {
            font-family: fonts.$quantitytext;
        }
    }
}
}

```

```

}

/* Tablet: */

@media (min-width: 481px) and (max-width: 768px) {}

/* Liten desktop */

@media (min-width: 769px) and (max-width: 1024px) {
  main {
    display: block;

    h1 {
      font-family: fonts.$titles;

      font-size: 2rem;

      margin: 0 auto;

      padding: 0 20px;
    }

    #infosection {
      display: flex;

      flex-direction: column;

      margin-right: 35rem;

      gap: 1rem;

      margin: 1 auto;
    }
  }
}

/* Stor desktop */

@media (min-width: 1025px) {
  main {
    display: block;

    h1 {
      font-family: fonts.$titles;

```

```

        font-size: 2rem;

        margin: 0 auto;

        padding: 0 20px;
    }

    section {

        display: flex;

        flex-direction: row;

        flex-wrap: wrap;

        justify-content: flex-start;

        gap: 2.5rem;

    }

    #infosection {

    }

}

}

@use "./partials/colors";

@use "./partials/fonts.scss";

//Mobile-first

#root {

    margin: 0

}

.event-info{

    list-style: none;

    li{

        margin: 10px 0;

    }

}

.event-menu{

```

```
display: flex;
justify-content: space-between;
list-style: none;
margin: 0;
li{
  a, button{
    background-color: colors.$buttons;
    color: colors.$whitetext;
    text-decoration: none;
    display: inline-block;
    padding: 0.5rem 1rem;
    text-align: center;
    border-style: none;
    border-radius: 2px;
    font-family: fonts.$quantitytext;
    box-sizing: content-box;
    font-size: large;
  }
}
}
.star{
  width: 30px;
  height: 30px;
  position: absolute;
  top: 20px;
  right: 20px;
  fill: colors.$header;
}
```

```
.event-card{  
    position: relative;  
}
```

Svar:

! Dette er roten til problemet:

scss

Copy

```
/* Stor desktop */  
@media (min-width: 1025px) {  
  main {  
    display: block;  
  
    ...  
  
    section {  
      display: flex;  
      flex-direction: row; // ← DENNE OVERSKRIVER ALT  
      flex-wrap: wrap;  
      justify-content: flex-start;  
      gap: 2.5rem;  
    }  
  }  
}
```

Denne globale regelen for alle section inne i main overstyrer alle mer spesifikke flex-direction: column du prøver å sette andre steder – selv inne i .filter-search.

✓ Slik fikser du det

◆ 1. Spesifiser kun layout for ønskede seksjoner

Endre:

scss

Copy

```
section {  
  display: flex;  
  flex-direction: row;  
  ...  
}
```

til noe mer presist, som kun gjelder arrangementskort e.l.:

scss

Copy

```
section.card-grid {
  display: flex;
  flex-direction: row;
  flex-wrap: wrap;
  justify-content: flex-start;
  gap: 2.5rem;
}
```

Da kan du i HTML skrive:

jsx

Copy

```
<section className="card-grid">...</section>
```

◆ 2. Behold .filter-search og lignende i column

Da slipper section.filter-search å bli overstyrt. Eksempel:

scss

Copy

```
@media (min-width: 1025px) {
  section.filter-search {
    display: flex;
    flex-direction: column;
    align-items: center;

    #filtercategory,
    #searchform {
      display: flex;
      flex-direction: column;
      max-width: 500px;
      width: 100%;
      margin: 1rem auto;
    }
  }
}
```


✅ Bonus: Hva du kan gjøre videre

- Flytt alle generelle layoutregler (som `main section`) til `layout.scss` eller `base.scss`
- I komponentfiler (som `filter.scss`), **unngå styling på bare `section` eller `main`** – bruk klasser
- Når du vil ha flere layouts (grid, list, column), lag egne `.grid-layout`, `.column-layout` osv.

💬 Oppsummert

🔧 Feilen er at du har:

scss

📋 Copy

```
main {
  section {
    display: flex;
    flex-direction: row; // ← globalt og farlig!
  }
}
```

💡 Løsningen er å **fjerne eller begrense dette** til f.eks.:

scss

📋 Copy

```
main {
  section.card-grid {
    ...
  }
}
```

➡ Da får `section.filter-search` og andre komponenter tilbake kontrollen over sin layout.

Trenger du hjelp til å rydde opp og splitte Sass-filen i partials med struktur? Jeg kan sette opp en for deg som gir deg mobil/tablet/desktop-responsiv kontroll for hver komponent 💖

📄 🔊 👍 🗨️ ↻

Refleksjon:

Det viste seg at feilen vi opplevde kom fra scss arket til home komponentet. Da vi hadde brukt veldig generelle tags for styling i home + at de var veldig lite beskrivende. Dette førte senere til at når vi prøvde å bruke `section {}` taggen i styling så ble den automatisk overskrevet med det innholdet som sto i den globale `section` taggen i `home.scss`.

For å løse problemet, benyttet vi mer spesifikke tagger i koden. På denne måten kan vi presisere at stylingen kun skal gjelde et sted i koden. Ingen kode ble direkte kopiert fra dette forslaget, men konseptet med enda mer spesifiserte tagger ble brukt. I tillegg til at vi brukte AI til å finne ut av hvor problemet vårt oppsto.