# Dag 1

I dag har jeg sett gjennom alle de teoretiske dataene om prosjekttemaet. Også, etter å ha sett de tekniske spesifikasjonene for oppgave A og B, slo jeg meg på sistnevnte. Jeg brukte den resterende tiden på å fullføre nivåer på odingfantasy.com og så et par korte treningsvideoer på Flexbox og Grid på YouTube.

# Dag 2

Jeg bestemte meg for å bruke denne dagen på å løse minioppgaver på flexbox.

MINIOPPGAVE 1

Her ble det foreslått å lage et dokument slik at det får en oversiktlig layout med en beholder som inneholder flere elementer. Til denne oppgaven laget jeg et HTML-dokument som inneholder:

1) <header>, som implementerer navigering på selve siden. Navigasjon opprettes ved hjelp av HTML <nav>-elementet.

list - <ul> og listeelement - <li>.

2) <main>: Hovedinnholdet på siden. Inneholder tre seksjoner som er organisert i containere med containerklassen for strukturering og styling.

- <section id="section1" class="container">: Den første delen med overskriften "Om oss" og et avsnitt med tekst som beskriver selskapet. Det er også et bilde med attributter inni.

- <section id="section2" class="container">: Andre seksjon med tittelen "Tjenester" og en liste over tjenester representert ved bruk av en unummerert liste <ul> og listeelementer <li>.

- <section id="section3" class="container">: Den tredje delen med overskriften "Kontakt oss" og et avsnitt med kontaktinformasjon. Inne er det også en <form> med inntastingsfelt for navn (<input type="text">) og melding (<textarea>), samt en send-knapp <button type="submit">.

3) <footer>: Sidebunntekst med tekst som inneholder informasjon om selskapets opphavsrett.

Hver del har en <h2>-overskrift og innhold som utfyller den tilsvarende delen av siden.

HTML-Doc



CSS



I oppgaven står det også "Test alle mulige verdier for CSS-egenskapen justify-content i en av containerne dine" . Siden jeg brukte "flex-direction: column;" , er hovedinnholdsjusteringsaksen endret til vertikal. For å se endringene økte jeg størrelsen på beholderne med den faste verdien slik at dimensjonene på beholderen ble større enn dimensjonene på innholdet i denne beholderen. (height: 800px;)

(merk: endringene sees best i avsnitt 2 "Tjenester")

flex-start

A robot working on a table

Description automatically generated

flex-end

A robot working in a factory

Description automatically generated

Center

A robot standing on a table

Description automatically generated

space-between

A robot working on a table

Description automatically generated

space-around

A robot standing on a table

Description automatically generated

space-evenly

A robot standing on a table

Description automatically generated

MINIOPPGAVE 2

Når du setter flex: 1;, flex: 1; og flex: 1



Når du setter flex: 1;, flex: 2; og flex: 7; på elementene .elem1, .elem2 og .elem3, vil de fordele plassen i containeren i forholdet 1:2:7. Dette betyr at:

.elem1 vil ta en del av den tilgjengelige plassen.

.elem2 vil ta to deler av den tilgjengelige plassen.

.elem3 vil ta syv deler av den tilgjengelige plassen.



MINIOPPGAVE 3

I denne oppgaven la jeg ganske enkelt til 1 linje i .sec1:

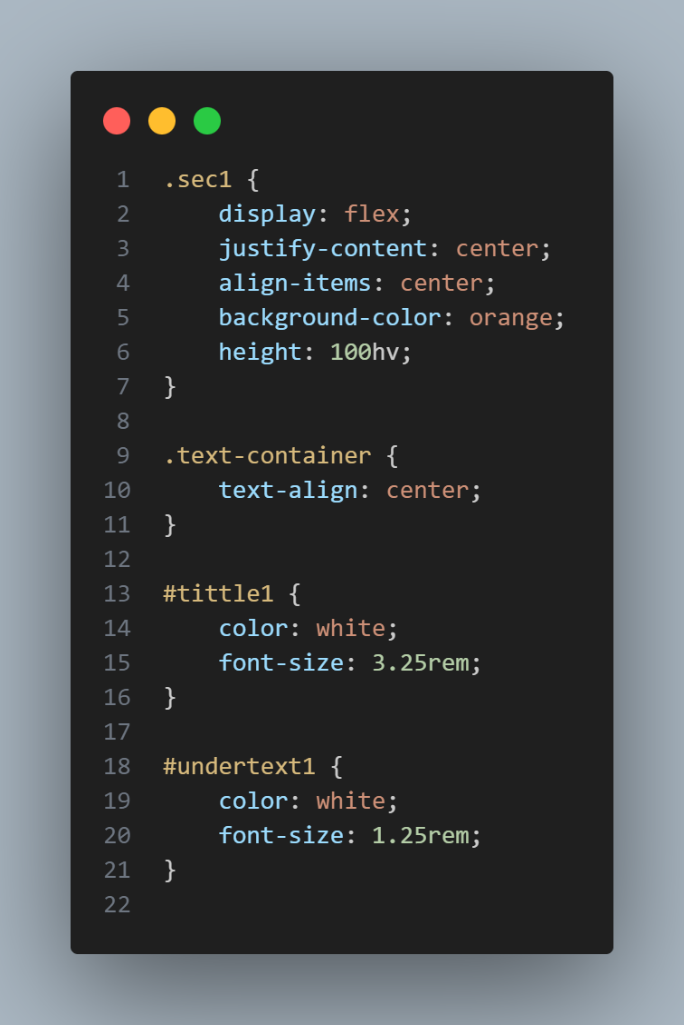
"flex-direction: column;" , som taklet oppgaven perfekt

A sign with white text

Description automatically generated

Men hvis du følger hintet. Deretter:





A yellow sign with white text

Description automatically generated

MINIOPPGAVE 4

A screenshot of a video game

Description automatically generated

A screenshot of a video game

Description automatically generated

MINIOPPGAVE 5

A screenshot of a video game

Description automatically generated

Alle de tre vanskelighetsnivåene ble fullført i 24/24

OPPGAVE B

Her avvek jeg litt (ikke med vilje) fra det tekniske oppgavesettet og lagde en nettside for salg av enheter til PC. Etter minioppgavene hadde jeg tid igjen, som jeg bestemte meg for å bruke på å lage hovedsiden til butikken.

Hele nettstedstrukturen er delt inn i 3 seksjoner:

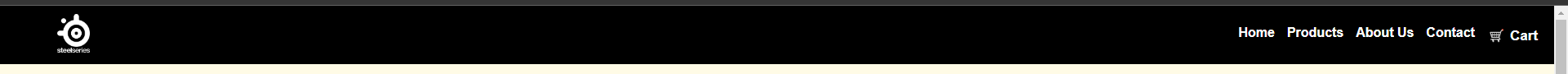
1) Overskriftsseksjon

2) Produktseksjon

3) Bunntekstseksjon

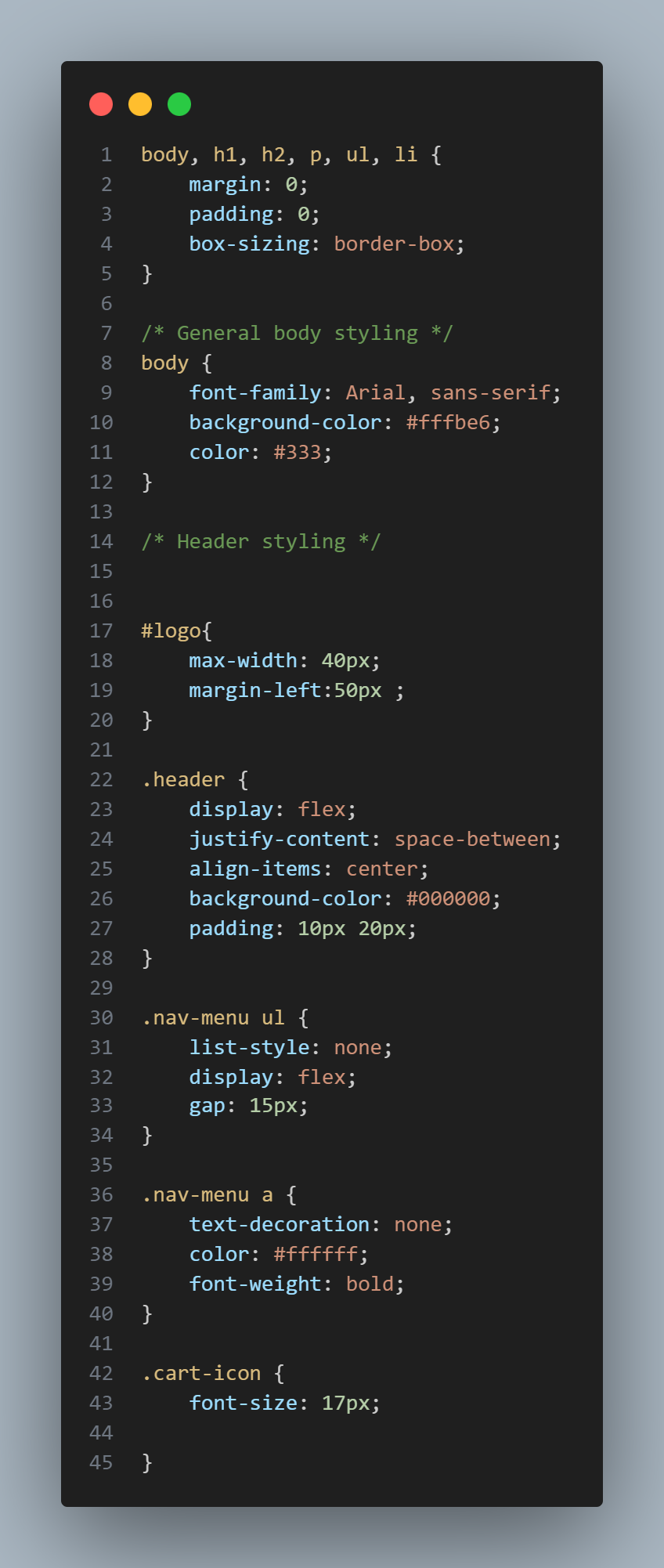
1. Overskriftsseksjon

Denne delen består av en logo og navigasjon implementert ved hjelp av <ul>. .header-klassen brukte "justify-content: space-between;" for å plassere logoen og navigasjonen på forskjellige sider av skjermen (i forhold til y-aksen). Også brukt var "align-items: center;" slik at objektene er vertikalt sentrert. Andre css-stiler ble brukt for visuelle forbedringer. Gap-egenskapen ble også brukt: 15px; i displayet: flex; brukes til å legge til 15px mellomrom mellom containerens barn.

Også for nettstedselementene ble "box-sizing: border-box;" brukt Når du bruker boksstørrelse: border-box;, er eventuelle utfyllings- og kantverdier inkludert i den totale bredden og høyden til elementet. Dette betyr at bredden og høyden på elementet ikke økes av disse polstringene og kantene. Dette gjør arbeidet med elementer mye enklere.

Html-header

Css-header+body



2.Produktseksjon

Dette er hoveddelen hvor produktkort er plassert, med produktnavn, pris og en knapp for å se. Selve seksjonen har blitt stylet med "display: grid;"



display: grid; - Angir en egenskap til beholderen .product-grid

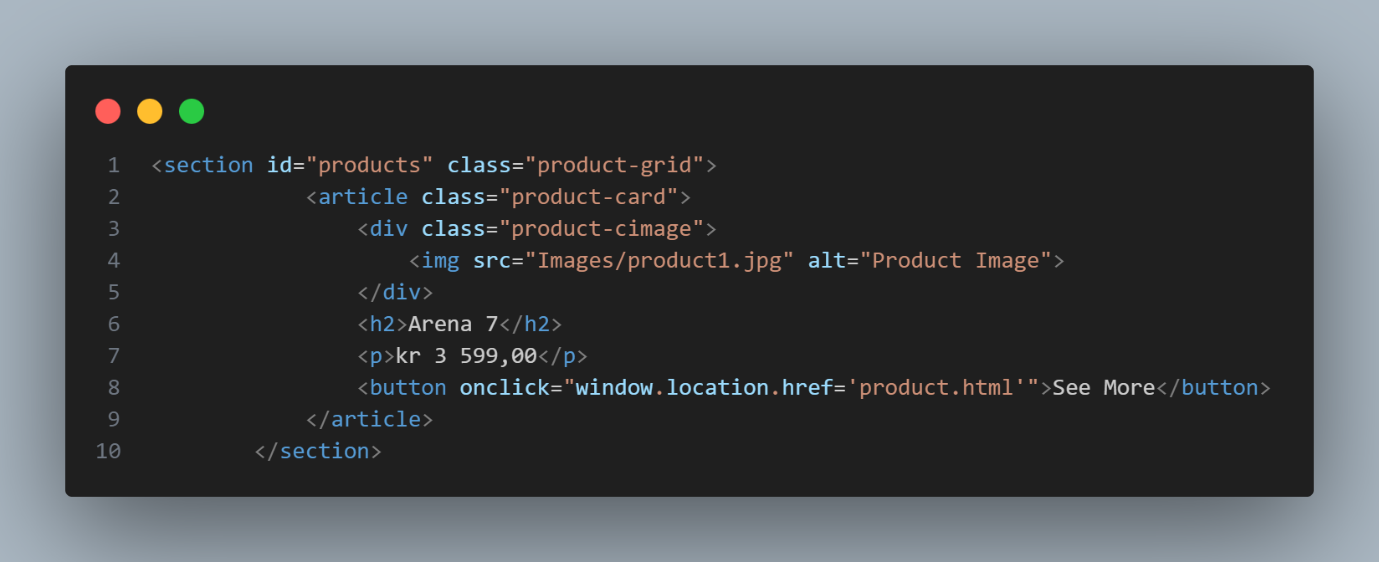
grid-template-columns: repeat(auto-fit, minmax(500px, 1fr)); - Bestemmer hvordan kolonnene skal fordeles i rutenettet.

repeat() brukes til å gjenta kolonner.

auto-fit justerer automatisk antall kolonner avhengig av tilgjengelig plass (ettersom du reduserer eller øker bredden på nettleservinduet).

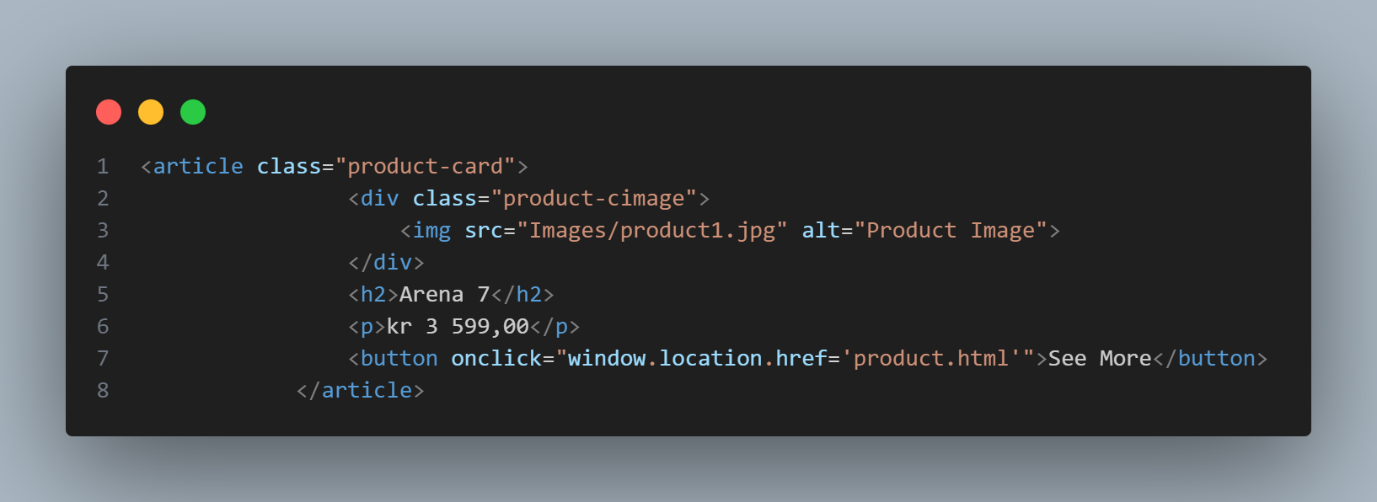
minmax(500px, 1fr) betyr at hver kolonne vil ha en minimumsbredde på 500px og en maksimal bredde lik 1 brøkdel av den gjenværende tilgjengelige plassen (1fr).

Og gap jeg har allerede brukt og forklart i avsnittet "Overskriftsseksjon".



<section> er selve produktdelen der det er produktkort (<article class="product-card">)

Her er et eksempel på det første produktkortet:



For å legge til nye produkter, kopierte og limte jeg rett og slett inn det første produktkortet.

Kortene ble også stilisert:



Eneste interessante her “overflow: hidden;” og “cursor: pointer;”

overflow: hidden - den skjuler alt innhold som ikke passer i blokken (for eksempel hvis innholdet er større enn dimensjonene til beholderen).

cursor: pointer - endrer utseendet til musepekeren når den er over et element på en nettside. Visuelt endres markøren til en hand.

A collage of images of a device

Description automatically generated

3. Bunntekstseksjon

Footer Section implementert veldig enkelt

A screenshot of a computer

Description automatically generated



# Dag 3

Produktside

Det ble opprettet en side for hvert produkt, med egne bilder, beskrivelse og popup-vindu når man legger et produkt i handlekurven (JS-funksjon). Faktisk er dette dupliserte sider der bare objektene ovenfor endres

Siden har også 3 seksjoner:

header

<main class="product-detail">

footer

De første og siste bruker samme stiler som hovedsiden.

<main class="product-detail"> den er også delt inn i 2 "seksjoner" (selv om du ikke umiddelbart kan se ut fra strukturen til koden):

1) <div class="product-image">

2) <div class="product-info">

Begge objektene har «display: flex» men har sine egne separate stiliseringer

På grunn av mangel på tid vil jeg ikke beskrive dem i detalj. Men du kan teste siden her ->