

WORKSHOP

Utvikling av interaktive nettsteder. Fredag 20. januar.

Agenda

08:15, Aud 2: Gjennomgang av oppgaver

08:45: Fordeling i klasserom, veiledet arbeid med oppgaver

11:30, Aud 2: Ta med lunsj for gjennomgang av løsningsforslag

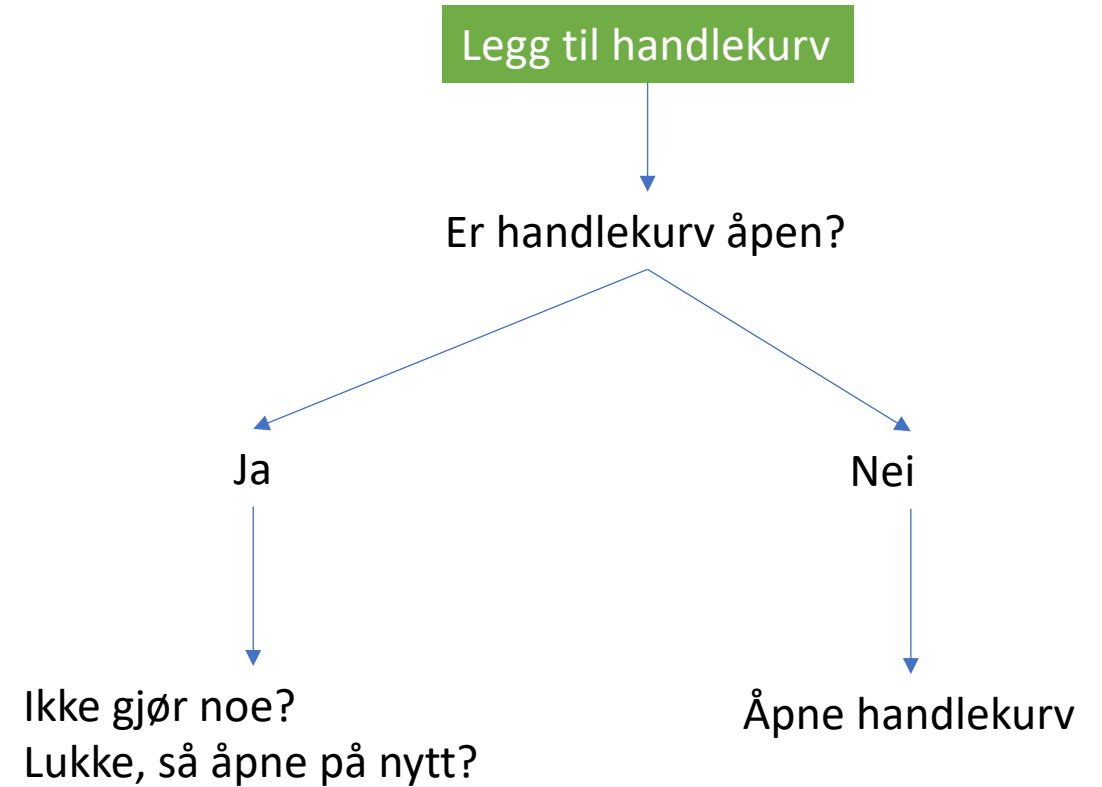
OPPGAVE 1

Åpne handlevognen når et nytt produkt legges til

Hva i koden "åpner handlevognen"?

Når må vi kalle på denne koden?

Siden vi "toggler" (slår av og på visningen), hva skjer dersom vi allerede har handlevognen åpen når vi legger til et nytt produkt?



OPPGAVE 2

Oppdater nødvendig kode for å få total pris i handlekurven.

Hvor i koden får vi tak i pris for hvert produkt?

Hvordan legger vi sammen prisene?

Når skal vi skrive ut total pris?

Hvor skal vi skrive ut total pris?

HINT:

```
let totalPrice = 0
```

*.map() henter alle produkter,
her kan vi få tak i pris...*

OPPGAVE 3

Lag slett-funksjonalitet som knyttes til X-knappen for hver produktlinje

Når en knapp trykkes, må koden vite hvilket product som skal slettes.

Det må finnes funksjonalitet for å slette produktet, men hva betyr rent praktisk "å slette"?

Når noe endres i produktvisningen, må handlevognen og antallet oppdateres.

HINT:

```
Array.splice()
```

Sende index som parameter i arrow-function i .map()?

Hva er denne parameteren?

```
Function deleteProduct(product)
```

Når må denne funksjonen kalles/kjøres?

Oppgave 3.1 (Ikke obligatorisk)

Få et produkt til å fade ut fra handlevognvisningen når man klikker slett (øvelse I å koble CSS og JavaScript)

Dette er visuelt lureri. Vi må bruke CSS for å fade ut (gi `` klassen “hidden”). Men – vi må fortsatt fjerne produktet fra `cart-arrayen`, men ikke før vi har faded det fra visningen.

HINT:

En kombinasjon av dynamisk id på ``-elementet (`<li id=“prod-0”>`, `<li id=“prod-1”>` osv) som kan brukes som selector i `deleteProduct`-funksjonen, `classList.add()`-metoden i kombinasjon med `hidden`-klassen vi allerede har laget og `setTimeout()`-funksjonen løser denne oppgaven.

PS: Du må også legge på `transition`-regler i CSS-reglene til ``.

Avanserte oppgaver

Avanserte oppgaver

- I de avanserte oppgavene skal vi gjøre innholdet enda mer dynamisk.
- Hent ned legodudes.js fra kursets GitHub-repository (ligger i legodudes_webshop-mappen)
- Kopier denne inn i ditt webshop-repository
- Bytt ut referansen til shopscripts.js i index.html-filen med legodudes.js

De videre oppgavene gjør visningen av innholdet og funksjonaliteten i nettbutikken betydelig mer dynamisk.

Oppgave 4

Lag nødvendig funksjonalitet for å liste opp produktene i products-arrayen i legodudes.js.

Hvilket HTML-element må vi skrive dem til?

Hvilke metoder må vi benytte?

Hvilke funksjonskall må opplistingen ta høyde for?

PS: Bruk prodid for å hente informasjon fra product—arrayen (eks: Når tittel skal skrives ut, bruk prodid for å finne riktig product, og hent tittel derfra)

PPS: HTML-en som skrives ut skal være lik HTML-en fra de forrige oppgavene, men nå skrives ut ved hjelp av JavaScript og products-arrayen.

Oppgave 5

- 1) Legg til antall på lager i visningen av produktet. Ta høyde for at tallet som vises skal oppdateres.
- 2) Dersom et produkt ikke finnes på lager (storage: 0), skal Legg i handlekurv-knappen ikke være tilgjengelig (disabled)

PS: Sett et av produktenes storage til 0 for å teste at dette funker.

Oppgave 6

Lag legg til handlekurv-funksjonalitet.

Vi ønsker nå kun en produktlinje per produkt. Det vil si: dersom et produkt med id 1 ligger i handlekurven, og vi forsøker å legge til et nytt produkt med samme id, skal antallet på denne varelinjen oppdateres.

Det skal ikke legges til en ny produktlinje med det samme produktet. Når et produkt legges til handlekurven, må antall på lager minskes i product-arrayen.

Hint: Når man legger til produkt i handlekurv, sjekk om prodid allerede finnes i cart-arrayen.

Oppgave 7

Lag slett-funksjonalitet for en produktlinje i handlekurven. Denne skal aktiveres når slett-knappen på en linje klikkes.

OBS: Her kan store deler av funksjonen fra oppgave 3 benyttes.

HUSK: Når vi sletter en produktlinje fra handlekurven, må antallet produkter på lager økes.

TIPS: Vi skal ved flere anledninger oppdatere lagerantall. Kanskje en funksjon som tar imot produkt og lageroppdatering er fornuftig, så vi kan kalle denne enkelt når vi skal oppdatere lager?

Oppgave 8

Gjør om visningen av antall produkter i handlevognen til å vises i et input-felt.

Til venstre for input-feltet, legg en knapp med minustegn. Ved klikk på knappen, reduser antall med en. Oppdater storage-antall for produktet med +1.

Hvis antall blir null (antall er 1 , klikk på knapp gir - 1), skal produktlinjen slettes.

Til høyre for input-feltet, legg en knapp med plusstegn. Ved klikk på knappen, forhøy antall med en. Oppdater storage-antall for produktet med -1.