

Avkrysningsbokser og radioknapper

Ved valg mellom flere alternativer kan avkrysningsbokser eller radioknapper være nyttige. Avkrysningsknapper er vanligvis firkantede, og det er mulig å velge flere av alternativene. Radioknapper er vanligvis runde, og det er bare mulig å velge ett alternativ.

Avkrysningsbokser og radioknapper har litt ulike egenskaper, men vi lager dem på omtrent samme måte.

Avkrysningsbokser

Vi kan bruke avkrysningsbokser når vi vil at brukeren skal kunne velge flere alternativer fra en liste med forhåndsbestemte alternativer. Det vil si at vi kan bestemme noen alternativer, og så kan brukeren velge ingen eller flere av disse alternativene.

Vi lager avkrysningsbokser med `<input>`-elementet og attributtene `name`, `value` og `type="checkbox"`:

```
<div class="checkboxradio">
  <h3>Format</h3>
  <label>
    <input type="checkbox" value="Kino" id="kino"> Kino
  </label>
  <label>
    <input type="checkbox" value="DVD" id="dvd"> DVD
  </label>
  <label>
    <input type="checkbox" value="Strømmetjeneste" id="strommetj
  </label>
  <label>
    <input type="checkbox" value="TV" id="tv"> TV
  </label>
</div>
```

Hver av boksene får sin unike id, slik at vi kan hente dem med JavaScript. Her kan du også se at vi har brukt den andre skrivemåten for `<label>`. Elementene har id, men det er likevel greit å plassere dem inni `<label>`-elementene, slik at det blir lettere å trykke på dem på nettsiden.

Her skal vi også bruke `<label>` for å angi CSS for avkrysningsknappene. Da er det en fordel å bruke denne skrivemåten. Det er også grunnen til at vi har samlet avkrysningsknappene i et `<div>`-element med en egen klasse. Vi må kunne angi CSS for disse `<label>`-elementene uten at de andre `<label>`-elementene på nettsiden blir påvirket. Vi skal snart bruke den samme klassen på radioknapper; derfor har den fått navnet «checkboxradio».

Vi skal utvide CSS-koden for disse elementene senere, men vi starter allerede nå med denne enkle koden som sørger for at avkrysningsboksene havner nedenfor hverandre:

```
.checkboxradio > label {  
  display: block;  
}
```



I nettleseren ser disse avkrysningsboksene slik ut:

HTML CSS

LIVE

```
1 <div id="hoved">  
2   <h1> Avkrysningsbokser </h1>  
3  
4   <div class="checkboxradio">  
5     <h3>Format</h3>  
6     <label>  
7       <input type="checkbox" value="Kino" id="kino"> Kino  
8     </label>  
9     <label>  
10      <input type="checkbox" value="DVD" id="dvd"> DVD  
11    </label>  
12    <label>  
13      <input type="checkbox" value="Strømmetjeneste"  
id="strommetjeneste"> Strømmetjeneste  
14    </label>  
15    <label>  
16      <input type="checkbox" value="TV" id="tv"> TV  
17    </label>  
18  </div>  
19  
20 </div>
```

Resources 1x

Legg merke til at bruken av `<label>` gjør at vi kan klikke på teksten, ikke bare i

selve firkanten.

I dybden

Bruk av id

I eksemplet ovenfor brukte vi enkle id-er som `kino`, `dvd` osv. Hvis en nettside har mange elementer, øker risikoen raskt for at flere elementer får samme id. For å unngå problemer knyttet til det kan vi bruke en egen skrivemåte for input-elementer. Vi kan for eksempel bruke id-er som `inpKino` for «input» eller `cbKino` for «checkbox». Finn gjerne din egen variant, og bruk den konsekvent.

Radioknapper

Radioknapper har samme funksjon som nedtrekksmenyer, det vil si at brukeren bare kan velge ett alternativ fra flere forhåndsbestemte alternativer. Forskjellen er at alternativene blir synligere når vi bruker radioknapper. Hvis det er mange alternativer, blir det likevel ryddigere å bruke en nedtrekksmeny.

Vi lager radioknapper med `<input>`-elementet og attributtene `type="radio"`, `name` og `value`. Ellers bruker vi same oppsett som for avkrysningsbokser:

```
<div class="checkboxradio">
```



```
<h3>Fått Oscar?</h3>
```

```
<label>
```

```
<input type="radio" name="oscar" value="Ja" id="ja"> Ja
```

```
</label>
```

```
<label>
```

```
<input type="radio" name="oscar" value="Nei" id="nei" checked=""
```

```
</label>
```

```
</div>
```

Her er det viktig at radioknappene får samme navn. Det er det som gjør at vi bare kan velge én av knappene. Vi har også lagt til `checked` for den ene knappen. Vi bør alltid velge én av radioknappene som avkrysset på forhånd. Da kan ikke brukeren la være å velge et alternativ. I nettleseren ser radioknapper slik ut:

HTML

CSS

Result

EDIT ON

LIVE

```
<div id="hoved">
  <h1> Radioknapper </h1>

  <div class="checkboxradio">
    <h3>Fått Oscar?</h3>
    <label>
      <input type="radio"
name="oscar" value="Ja"
id="ja"> Ja
    </label>
    <label>
      <input type="radio"
name="oscar" value="Nei"
id="nei" checked> Nei
    </label>
  </div>
</div>
```

Resources1x0.5x0.25xRerun

Legg merke til at bruken av `<label>` gjør at vi kan klikke på teksten, ikke bare i selve sirkelen.

Snakk!

Nedtrekksmeny eller avkrysningsbokser?

Hva er den praktiske forskjellen mellom avkrysningsbokser og en nedtrekksmeny? Hvorfor trenger vi begge variantene?

Finne verdiene med JavaScript

I kodene ovenfor har alle avkrysningsbokser og radioknapper fått hver sin id. Det betyr at vi kan hente dem inn hver for seg, og sjekke om de er avkrysset ved hjelp av egenskapen `checked` . La oss se på radioknappene vi laget ovenfor:

```
<div class="checkboxradio">  
  <h3>Fått Oscar?</h3>  
  <label>  
    <input type="radio" name="oscar" id="ja"> Ja  
  </label>  
  <label>  
    <input type="radio" name="oscar" id="nei" checked> Nei  
  </label>  
</div>
```

Her kan vi hente inn de to knappene slik:

```
let jaEl = document.querySelector("#ja");  
let neiEl = document.querySelector("#nei");
```

Vi kan nå sjekke tilstanden til de to radioknappene, for eksempel slik:

```
if jaEl.checked() {  
  // Filmen har fått Oscar  
} else {  
  // Filmen har ikke fått Oscar  
}
```

Her klarer vi oss med `else` på slutten fordi ett av de to alternativene må være valgt av brukeren. Hvis vi har flere radioknapper, kan vi legge til flere betingelser med `else if`.

Avkrysningsbokser fungerer på samme måte. Hvis vi har hentet inn elementene, kan vi sjekke om de er huket av med egenskapen `checked`.

I eksemplet nedenfor har vi gjort det samme for avkrysningsboksene beskrevet ovenfor. Vi har lagt til en knapp, og når den trykkes, vises en liste over hva brukeren har valgt.

Bruk av `querySelectorAll()`

I stedet for å gi hvert element en id, kan vi hente inn alle elementer av samme type med `querySelectorAll()` :

```
let cbEl = document.querySelectorAll("input[type='checkbox']");
```

Her bruker vi `"input[type='checkbox']"` som gir oss alle `<input>`-elementer som har `type="checkbox"`. Hvis vi bare skriver `"input"`, vil vi også få alle tekstfelter, tallfelter og radioknapper som er på samme nettside.

Snakk!

Flere grupper med avkrysningsbokser

Hvis du har flere grupper med avkrysningsbokser på nettsiden din, vil det ikke fungere å skrive

```
let cbEl = document.querySelectorAll("input[type='checkbox']");
```

Da ville vi få alle avkrysningsboksene på siden, ikke bare de vi ønsker oss.

Hvordan kan vi løse det?

I stedet for å ha én variabel per element, har vi nå én array som inneholder alle elementene. Den kan vi løpe gjennom med en `for`-løkke, sjekke hvilke som er huket av, og skrive ut deres value:

```
for (let i = 0; i < cbEl.length; i++) {  
  if (cbEl[i].checked) {  
    console.log(cbEl[i].value);  
  }  
}
```



I eksemplet nedenfor har vi gjort det samme som i eksemplet lenger opp på siden, men her har vi brukt `querySelectorAll()` i stedet for å hente elementene hver for seg.

Oppgaver

- 4 Lag en valutakalkulator med radioknapper og et tallfelt. Brukeren skal skrive inn en sum i tallfeltet og velge valuta med radioknappene. Du velger selv hvilke valutaer kalkulatoren skal gjøre om mellom.
- 5 Lag en nettside med fire avkrysningsbokser og ett `<div>`-element. Valgene brukeren tar, skal påvirke utseendet til `<div>`-elementet. Eksempler på alternativer er kantlinje (tykkelse og stil), avrundede hjørner og med/uten farge.
- 6 Utvid «visittkortgeneratoren» med noen avkrysningsknapper og radioknapper:
 - a Avkrysningsknapper som lar brukeren velge hvilken informasjon som skal tas med på visittkortet (for eksempel knapper som lar dem velge om de vil vise telefonnummer, beskrivelse og skoletrinn).
 - b Radioknapper som lar brukeren velge mellom to ulike visittkortdesign (for eksempel to forskjellige bakgrunnsfarger).