

CSS challenge #8

# DE STATES VAN CONTROLS INTRO

# 1. Buttons

# Buttons

Sample: [codepen.io/shoof/pen/jOdKjME](https://codepen.io/shoof/pen/jOdKjME)

Sample met 12  
hamburger buttons:  
[codepen.io/shoof/pen/LYMEWVm](https://codepen.io/shoof/pen/LYMEWVm)

Met `color-scheme: light dark`; passen buttons zich vanzelf aan als de gebruiker voor light of dark op zijn computer kiest. Dat werkt ook voor andere elementen.

`button`

Ik ben een button

Ik ben een button

`button:hover`

Ik ben een button

`button:focus-visible`

Ik ben een button

`button:active`

Ik ben een button

`button[disabled]`

Ik ben een button

`button[type="submit"]`

**Ik ben de submit**

Door `:focus-visible` te gebruiken i.p.v. `:focus` krijgt een button alleen focus als die met het toetsenbord geselecteerd is (en niet als er op geklikt is). Dat werkt ook voor andere elementen.

Een button met een `disabled` attribuut

Een button met het type `submit` (1 per form)

De cursor wordt standaard geen pointer bij hoveren. Dat kun je aanpassen met `cursor:pointer`; Dat moet dan weer niet voor disabled buttons.

Buttons krijgen standaard styling mee van het besturingssysteem (die is op elk systeem anders). Die styling kun je ongedaan maken met: `appearance:none`;

Het font van buttons is standaard het font dat het besturingssysteem voor buttons gebruikt. De font-size is ook meestal net wat kleiner. Dat kun je aanpassen met `font-family` en `font-size`;

## 2. Links

# Links

Sample: [codepen.io/shoof/pen/PoVavVd](https://codepen.io/shoof/pen/PoVavVd)

Sample met 32 leuke links: [codepen.io/shoof/pen/KKbweGB](https://codepen.io/shoof/pen/KKbweGB)

a (normaal)

[Dit is een link](#)

a:visited

[Dit is een link](#)

a:hover

[Dit is een link](#)

a:focus

[Dit is een link](#)

a:active

[Dit is een link](#)

Een link waarvan de href op pdf eindigt

a[href\$="pdf"]

[Dit is een docje \(pdf\)](#)

Een link waarvan de href met mailto begint

a[href^="mailto"]

[Naar email \(mail\)](#)

Het streepje kun je weghalen met `text-decoration:none;`

Maar je zou juist ook kunnen experimenteren met:

- `text-decoration-line`
- `text-decoration-color`
- `text-decoration-style`
- `text-decoration-thickness`
- `text-decoration-skip-ink`
- `text-underline-offset`
- `text-underline-position`

Dat is de outline. Die kun je weghalen met `outline:none;` Zorg dat je dan wel iets anders toevoegt voor de focus state.

# 3. Inputs

# Inputs

Sample 1/3 states:  
[codepen.io/shooft/pen/yLZEdpa](https://codepen.io/shooft/pen/yLZEdpa)

Sample 2/3 moving labels:  
[codepen.io/shooft/pen/wvNxwwP](https://codepen.io/shooft/pen/wvNxwwP)

Sample 3/3 background images:  
[codepen.io/shooft/pen/XWOBrjv](https://codepen.io/shooft/pen/XWOBrjv)

Het label en de input kunnen onder elkaar worden geplaatst. Met for en id kunnen label en input dan aan elkaar gekoppeld worden.

```
<label for="naam">naam</label>
<input type="text" id="naam" required>
```

Met required geef je aan dat de input verplicht ingevuld moet worden.

```
<label>
  email
  <input type="email" required>
</label>
```

De input kan ook in het label worden geplaatst. Het label en de input horen dan automatisch bij elkaar.

Het type van een input is van belang:

- om te controleren of de ingevoerde input aan de voorwaarden voldoet,
- voor telefoons om het handigste toetsenbord te tonen (voor email bijv. een toetsenbord met @),
- om de styling voor verschillende type inputs aan te passen.

# Inputs

Sample 1/3 states:  
[codepen.io/shooft/pen/yLZEdpa](https://codepen.io/shooft/pen/yLZEdpa)

Sample 2/3 moving  
labels: [codepen.io/shooft/pen/wvNxwwP](https://codepen.io/shooft/pen/wvNxwwP)

Sample 3/3  
background images:  
[codepen.io/shooft/pen/XWOBrjv](https://codepen.io/shooft/pen/XWOBrjv)

Indien de ingevoerde tekst  
voldoet aan de voorwaarden  
van het type van de input.

Indien de ingevoerde tekst niet  
voldoet aan de voorwaarden  
van het type van de input.

input

email...

input: hover

email...

input: focus

|email...

input: valid

aap@mies.nl

input: invalid

aap.mies.nl

input: placeholder-shown

Wat zoek je...

input[disabled]

Wat zoek je...

Inputs krijgen standaard styling  
mee van het besturingssysteem  
(die is op elk systeem anders). Die  
styling kun je ongedaan maken  
met: `appearance: none;`

Het font van input is standaard het  
font dat het besturingssysteem  
voor buttons gebruikt. De font-size  
is ook meestal net wat kleiner. Dat  
kun je aanpassen met font-  
family en font-size;

Met `color-scheme: light  
dark;` passen inputs zich vanzelf  
aan als de gebruiker voor light of  
dark op zijn computer kiest.

De kleur van de caret kun je met  
`caret-color` aanpassen.

De placeholder kun je met  
`input::placeholder` aanpassen.

Een input met een `disabled`  
attribuut



# Inputs

Sample 1/3 states:

[codepen.io/shooft/pen/yLZEdpa](https://codepen.io/shooft/pen/yLZEdpa)

Sample 2/3 moving

labels: [codepen.io/](https://codepen.io/shooft/pen/wwNxwwP)

[shooft/pen/](https://codepen.io/shooft/pen/wwNxwwP)

[wwNxwwP](https://codepen.io/shooft/pen/wwNxwwP)

Sample 3/3

background images:

[codepen.io/shooft/](https://codepen.io/shooft/pen/XWOBrjv)

[pen/XWOBrjv](https://codepen.io/shooft/pen/XWOBrjv)

Extra padding aan de  
linkerkant zodat de tekst  
na het plaatje start.

```
padding:.25em .25em .25em 2em;  
background-image: url("lock.png");  
background-size:1.5em auto;  
background-repeat:no-repeat;  
background-position:.25em center;
```

Je kunt een  
background-image aan  
een input toevoegen.



## **4. Checkboxes en radio buttons**

# Checkboxes en radio buttons

Sample: [codepen.io/shoof/pen/BaMPBeG](https://codepen.io/shoof/pen/BaMPBeG)

De input en het label kunnen onder elkaar worden geplaatst. Met for en id kunnen label en input dan aan elkaar gekoppeld worden.

```
<input id="pannenkoek" type="checkbox">  
<label for="pannenkoek">pannenkoek</label>
```

```
<label>  
  <input type="checkbox">  
  <span>poffertjes</span>  
</label>
```

De input kan ook in het label worden geplaatst. Het label en de input horen dan automatisch bij elkaar.

In beide gevallen kun je fijn ook op het label klikken om de checkbox (of radio button) aan/uit te zetten. Het target wordt zo een stuk groter dan het kleine vierkantje of rondje.

# Checkboxes en radio buttons

Sample: [codepen.io/shoof/pen/BaMPBeG](https://codepen.io/shoof/pen/BaMPBeG)

`input[type="checkbox"]:checked`

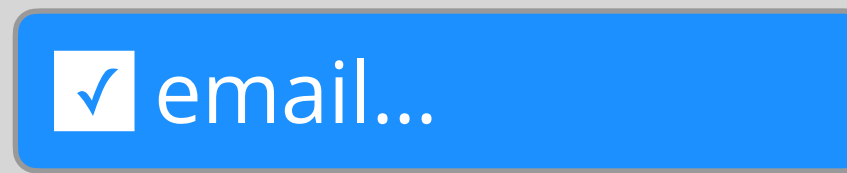
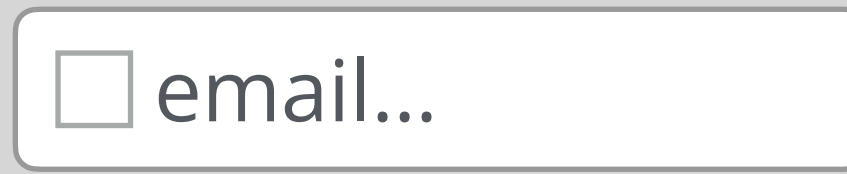
De kleur van de gecheckte box kun je aanpassen met `accent-color`.

Verdere aanpassingen waren lange tijd niet mogelijk.

Nu kan met `appearance:none`; de standaard styling van de checkbox ongedaan gemaakt worden. Om daarna je eigen styling toe te passen.

`input[type="checkbox"]:checked + span` is de manier om de span naast de checkbox te stylen als de checkbox gecheckt is. `+` is de direct sibling selector.

De input zelf stylen is nog best lastig (en wordt niet door alle browsers ondersteund). Daarom wordt de checkbox zelf vaak verborgen en wordt het label (of de span) zo gestyled dat die op een checkbox lijkt. Daarvoor wordt vaak `::before` en `::after` en/of images gebruikt.



Artikel van Sara Soueidan over het toegankelijk stylen van checkboxes en radio buttons:  
[www.sarasoueidan.com/blog/inclusively-hiding-and-styling-checkboxes-and-radio-buttons/](https://www.sarasoueidan.com/blog/inclusively-hiding-and-styling-checkboxes-and-radio-buttons/)

# Checkboxes en radio buttons

Sample: [codepen.io/shoof/pen/BaMPBeG](https://codepen.io/shoof/pen/BaMPBeG)

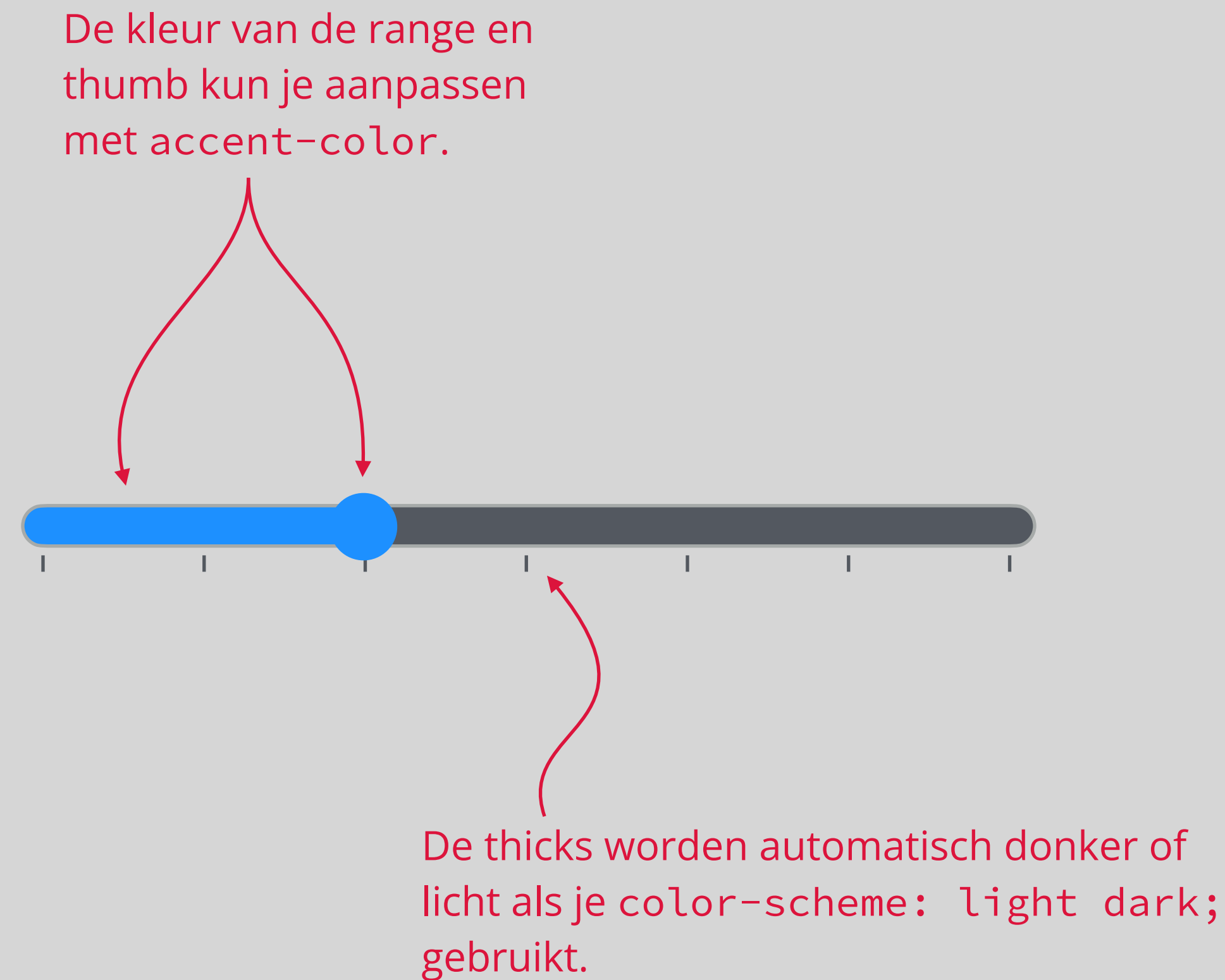


Artikel van Sara Soueiden over het toegankelijk stylen van een toggle switch (spoiler → 2 radio buttons):  
[www.sarasoueidan.com/blog/toggle-switch-design/](https://www.sarasoueidan.com/blog/toggle-switch-design/)

# 5. Ranges

# Ranges

Sample: [codepen.io/shoof/pen/rNPrBKa](https://codepen.io/shoof/pen/rNPrBKa)



Met wat gedoe zijn ranges verder aan te passen:

[www.smashingmagazine.com/2021/12/create-custom-range-input-consistent-browsers](https://www.smashingmagazine.com/2021/12/create-custom-range-input-consistent-browsers)

Een collectie custom ranges van Ana Tudor: [codepen.io/collection/DgYaMj/](https://codepen.io/ana-tudor/collection/DgYaMj/)

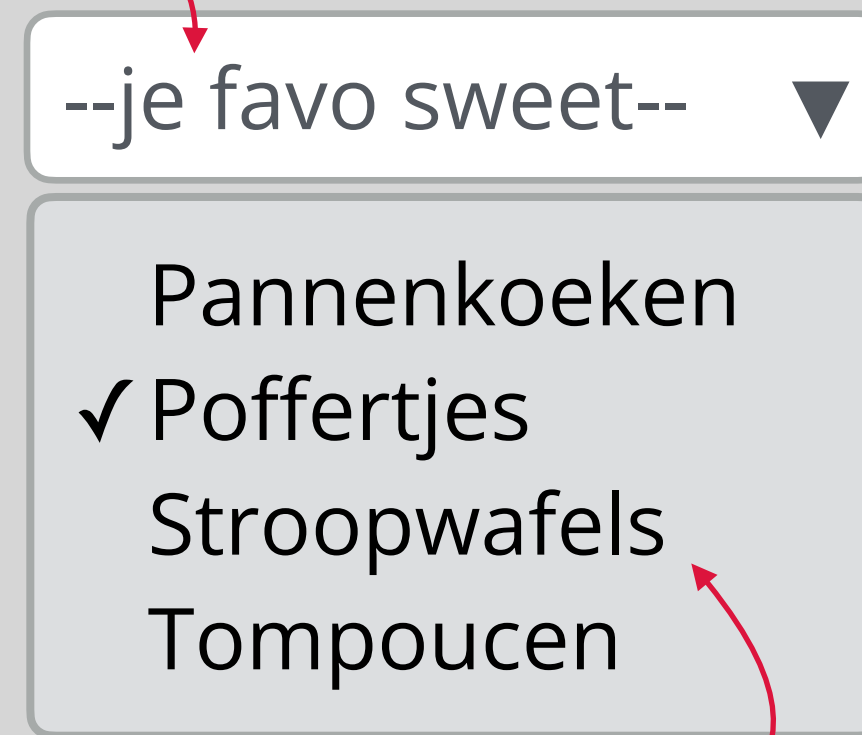
# 6. Selects



# Selects

Sample: [codepen.io/shoof/pen/MWLBgqg](https://codepen.io/shoof/pen/MWLBgqg)

De box kun je ongeveer op dezelfde manier stylen als inputs



De dropdown is nu ook te stylen :)

Selects worden nu vaak nog zelf gemaakt. Helaas meestal met veel divjes en niet toegankelijk. Maar dat hoeft binnenkort niet meer!

Selects zijn inmiddels te stylen :) Dat gaan we doen in een eigen challenge in semester 2: [una.im/select-updates/](https://una.im/select-updates/)

# **7. Form, fieldsets en legends**

# Form, fieldsets en legends

Sample: [codepen.io/  
shoof/pen/oNmMNpy](https://codepen.io/shoof/pen/oNmMNpy)

Een form is :valid als  
alle fieldsets in het  
form :valid zijn.

<form>

Fieldsets zijn de sections van forms.

Legends zijn de headings van fieldsets.

<fieldset>

<legend>contact</legend>

<label>naam <input type="text" required></label>

<label>email <input type="email" required></label>

</fieldset>

Een fieldset is :valid als  
alle required velden in de  
fieldset :valid zijn.

<fieldset>

<legend>prefs</legend>

<label><input type="checkbox" required> zoet</label>

<label><input type="checkbox" required> hartig</label>

</fieldset>

<button type="submit">go</button>

</form>

# INTRO COMBOGS LAW DE STATES

CSS challenge #8