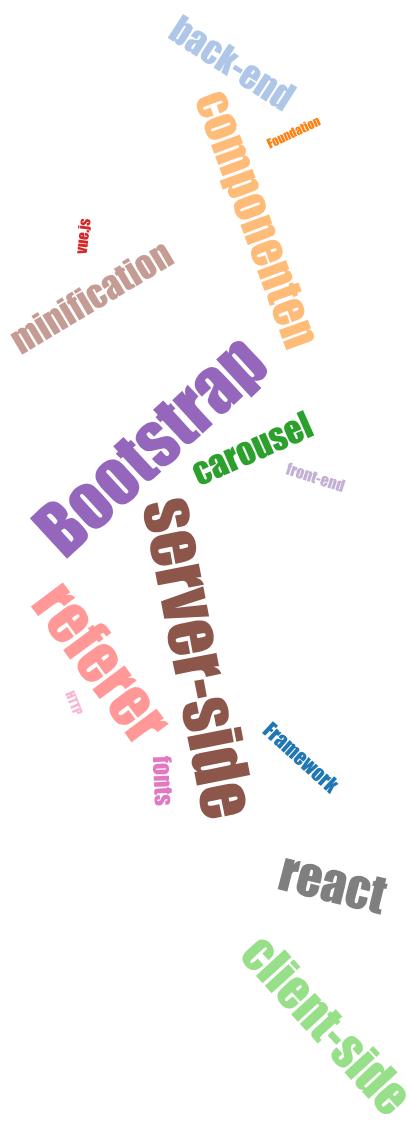


FRONT-END: BOOTSTRAP

Les 2 Webdevelopment

Klik hier voor pdf...



WAAR STAAN WE?

1. Frontend: HTML, CSS, Javascript
2. **Frontend: Bootstrap**
3. Backend: C#
4. MVC
5. Backend: C# LINQ
6. ORM
7. Testen
8. Layout
9. Zoeken/filteren, Sorteren, Pagineren
10. Web API, JSON, Ajax, Azure
11. Security
12. Architectuur

VANDAAG OP HET PROGRAMMA

- HTML/CSS/Javascript (vervolg)
- HTTP
- Bootstrap
 - Goed opgeleid?
 - Inleiding opdracht

BEKIJK OOK EENS DE LESDOELEN!

HTML/CSS/Javascript (VERVOLG)

WEBFONTS

Zie de MDN web docs. Laad het font bestand in:

```
1 @font-face {
2   font-family: "mijnfont";
3   src: url("fonts/mijnfont.eot");
4   src: url('fonts/mijnfont.eot?#iefix') format('embedded-opentype'),
5        url("fonts/mijnfont.woff") format("woff"),
6        url("fonts/mijnfont.ttf") format("truetype"),
7        url("fonts/mijnfont.svg#mijnfont") format("svg");
8   font-weight: normal;
9   font-style: normal;
10 }
```

```
<div style="font-family:mijnfont;">Hallo!</div>
```

PICTOGRAMMEN: PICTOGRAMMEN MET WEBFONTS

1. Laad het font bestand in:

```
1 @font-face {  
2   font-family: "icicons";  
3   src: url("fonts/icons.eot");  
4   src: url('fonts/icons.eot#iefix') format('embedded-opentype'),  
5         url("fonts/icons.woff") format("woff"),  
6         url("fonts/icons.ttf") format("truetype"),  
7         url("fonts/icons.svg#icons") format("svg");  
8   font-weight: normal;  
9   font-style: normal;  
10 }
```

1. Voor zoekmachines en screenreaders:

```
1 .hide {  
2   clip: rect(1px, 1px, 1px, 1px);  
3   height: 0;  
4   opacity: 0;  
5   position: absolute;  
6   visibility: hidden;  
7   width: 0;  
8 }
```

2. Selecteer het font als de klasse icon- bevat:

```
1 [class*="icon-"]::before {  
2   content: '+';  
3   display: inline-block;  
4   font-family: "icicons" !important;  
5   line-height: 1;  
6   position: relative;  
7   top: 2px;  
8   -webkit-font-smoothing: antialiased;  
9   font-smoothing: antialiased;  
10 }
```

2. Geef het karakter een zinvolle naam:

```
.icon-twitter::before { content: "t"; }
```

3. Zo voegen we het pictogram toe:

```
<a href="http://www.twitter.com/" class="icon-twitter">  
  <li class="hide">Twitter</li>  
</a>
```

PICTOGRAMMEN: PICTOGRAMMEN MET WEBFONTS

1. Laad het font bestand in:

```
1 @font-face {
2   font-family: "Icons";
3   src: url("fonts/icons.eot");
4   src: url("fonts/icons.eot?#iefix") format("embedded-opentype"),
5        url("fonts/icons.woff") format("woff"),
6        url("fonts/icons.ttf") format("truetype"),
7        url("fonts/icons.svg#Icons") format("svg");
8   font-weight: normal;
9   font-style: normal;
10 }
```

1. Voor zoekmachines en screenreaders:

```
1 .hide {
2   color: transparent;
3   height: 0;
4   opacity: 0;
5   position: absolute;
6   visibility: hidden;
7   width: 0;
8 }
```

2. Selecteer het font als de klasse icon- bevat:

```
1 [class^="icon-"]::before {
2   content: '+';
3   display: inline-block;
4   font-family: "Icons" !important;
5   line-height: 1;
6   position: relative;
7   top: 2px;
8   -webkit-font-smoothing: antialiased;
9   font-smoothing: antialiased;
10 }
```

2. Geef het karakter een zinvolle naam:
3. Zo voegen we het pictogram toe:

```
<a href="http://www.twitter.com/" class="icon-twitter">
<li class="slide">Twitter</li>
</a>
```

PICTOGRAMMEN: PICTOGRAMMEN MET WEBFONTS

1. Laad het font bestand in:

```
1 @font-face {
2   font-family: "Icons";
3   src: url("fonts/icons.eot");
4   src: url("fonts/icons.eot?#iefix") format("embedded-opentype"),
5        url("fonts/icons.woff") format("woff"),
6        url("fonts/icons.ttf") format("truetype"),
7        url("fonts/icons.svg#Icons") format("svg");
8   font-weight: normal;
9   font-style: normal;
10 }
```

1. Voor zoekmachines en screenreaders:

```
1 .hide {
2   clip: rect(1px, 1px, 1px, 1px);
3   height: 0;
4   opacity: 0;
5   position: absolute;
6   visibility: hidden;
7   width: 0;
8 }
```

2. Selecteer het font als de klasse icon- bevat:

```
1 [class^="icon-"]::before {
2   content: '+';
3   display: inline-block;
4   font-family: "Icons" !important;
5   line-height: 1;
6   position: relative;
7   top: 2px;
8   -webkit-font-smoothing: antialiased;
9   font-smoothing: antialiased;
10 }
```

2. Geef het karakter een zinvolle naam:

```
1 .icon-twitter:before { content: "tr"; }
2
3
4 <a href="http://www.twitter.com/" class="icon-twitter">
5   <li class="slide">Twitter</li>
6 </a>
```

3. Zo voegen we het pictogram toe:

PICTOGRAMMEN: PICTOGRAMMEN MET WEBFONTS

1. Laad het font bestand in:

```
1 @font-face {
2   font-family: "Icons";
3   src: url("fonts/icons.eot");
4   src: url("fonts/icons.eot?#iefix") format("embedded-opentype"),
5        url("fonts/icons.woff") format("woff"),
6        url("fonts/icons.ttf") format("truetype"),
7        url("fonts/icons.svg#icon") format("svg");
8   font-weight: normal;
9   font-style: normal;
10 }
```

1. Voor zoekmachines en screenreaders:

```
1 .hide {
2   clip: rect(1px, 1px, 1px, 1px);
3   height: 0;
4   opacity: 0;
5   position: absolute;
6   visibility: hidden;
7   width: 0;
8 }
```

2. Selecteer het font als de klasse icon- bevat:

```
1 [class^="icon-"]::before {
2   content: '+';
3   display: inline-block;
4   font-family: "Icons" !important;
5   line-height: 1;
6   position: relative;
7   top: 2px;
8   -webkit-font-smoothing: antialiased;
9   font-smoothing: antialiased;
10 }
```

2. Geef het karakter een zinvolle naam:

```
1 .icon-twitter::before { content: "tr"; }
2
3 href="http://www.twitter.com/" class="icon-twitter">
4 <li class="slide">Twitter</li>
5 </a>
```

3. Zo voegen we het pictogram toe:

PICTOGRAMMEN: PICTOGRAMMEN MET WEBFONTS

1. Laad het font bestand in:

```
1 @font-face {
2   font-family: "Icons";
3   src: url("fonts/icons.eot");
4   src: url("fonts/icons.eot?#iefix") format("embedded-opentype"),
5        url("fonts/icons.woff") format("woff"),
6        url("fonts/icons.ttf") format("truetype"),
7        url("fonts/icons.svg#Icons") format("svg");
8   font-weight: normal;
9   font-style: normal;
10 }
```

1. Voor zoekmachines en screenreaders:

```
1 .hide {
2   clip: rect(1px, 1px, 1px, 1px);
3   height: 0;
4   opacity: 0;
5   position: absolute;
6   visibility: hidden;
7   width: 0;
8 }
```

2. Selecteer het font als de klasse icon- bevat:

```
1 [class^="icon-"]::before {
2   content: '+';
3   display: inline-block;
4   font-family: "Icons" !important;
5   line-height: 1;
6   position: relative;
7   top: 2px;
8   -webkit-font-smoothing: antialiased;
9   font-smoothing: antialiased;
10 }
```

2. Geef het karakter een zinvolle naam:
3. Zo voegen we het pictogram toe:

```
<a href="http://www.twitter.com/" class="icon-twitter">
<li class="hide">Twitter</li>
</a>
```

FONT ICONS EN ACCESSIBILITY

<https://cloudfour.com/thinks/seriously-dont-use-icon-fonts/>

- Dyslectici gebruiken soms **andere fonts**
- Screenreaders kunnen geen pictogrammen voorlezen

BIJVOORBEELD: FONT AWESOME

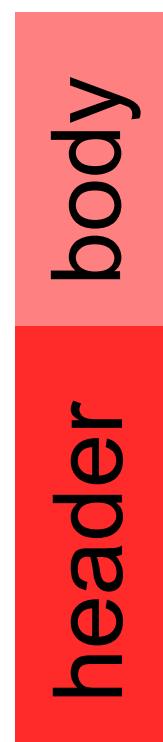
```
<html>
<head>
  <link rel="stylesheet" href="https://netdna.bootstrapcdn.com/font-awesome/4.0.3/css/font-awesome.css">
</head>
<body>
  <i class="fa fa-refresh fa-spin"></i>
</body>
</html>
```

HTTP

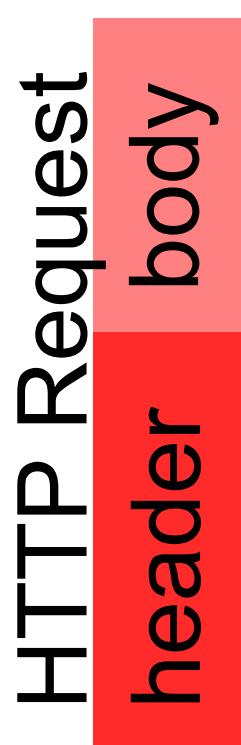
PROTOCOLLEN

body

PROTOCOLLEN



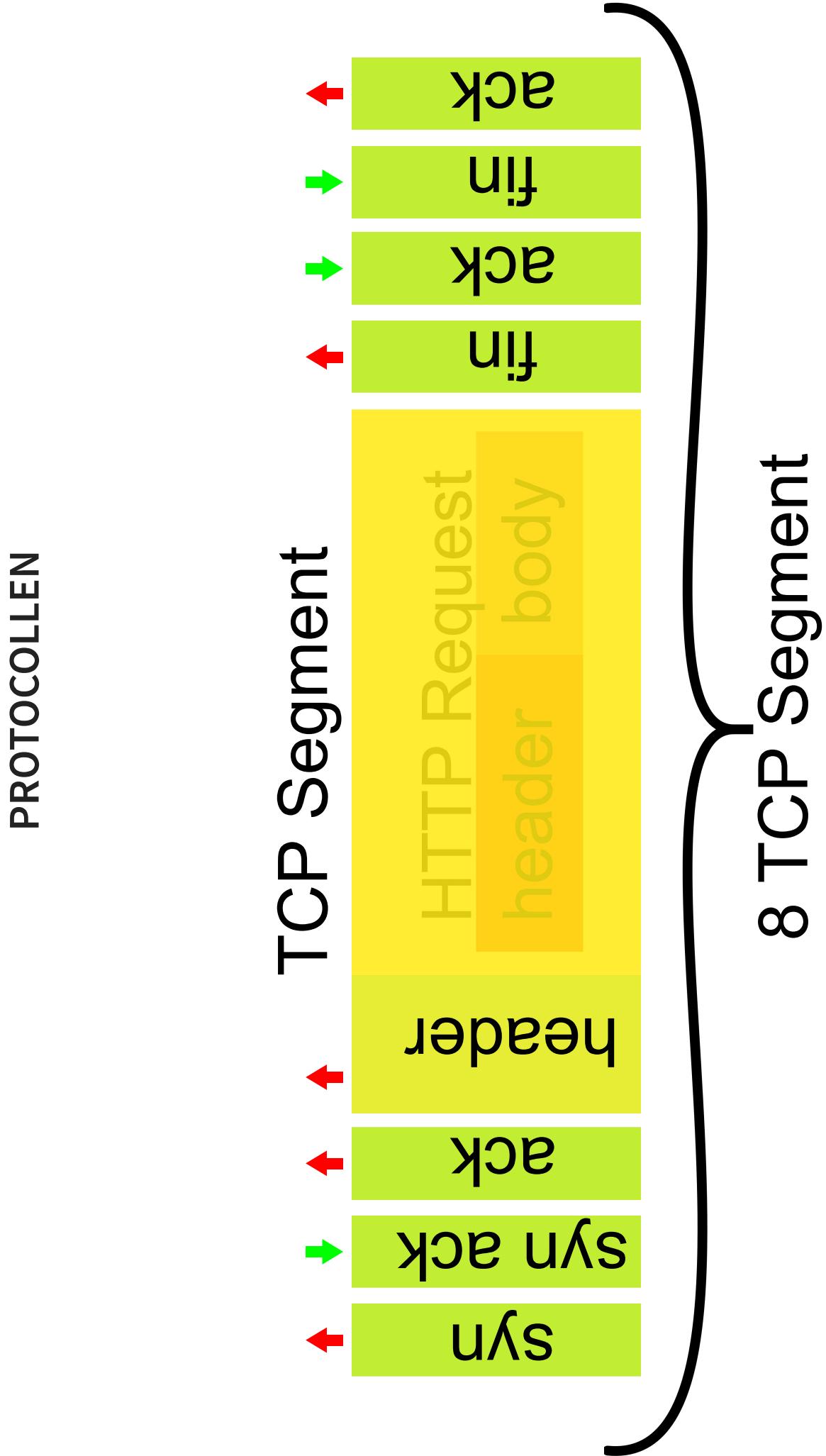
PROTOCOLLEN



PROTOCOLLEN

TCP Segment





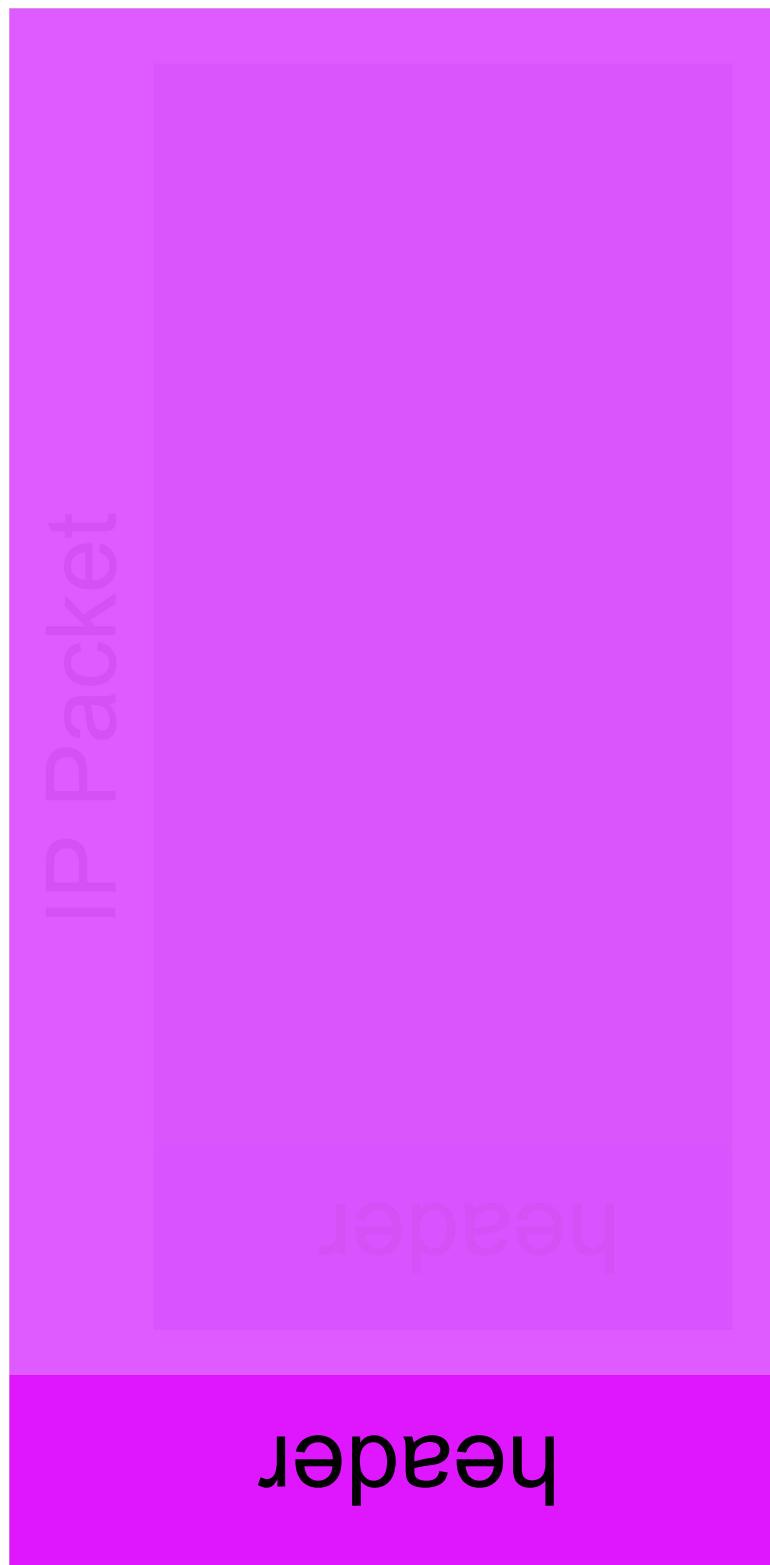
PROTOCOLLEN

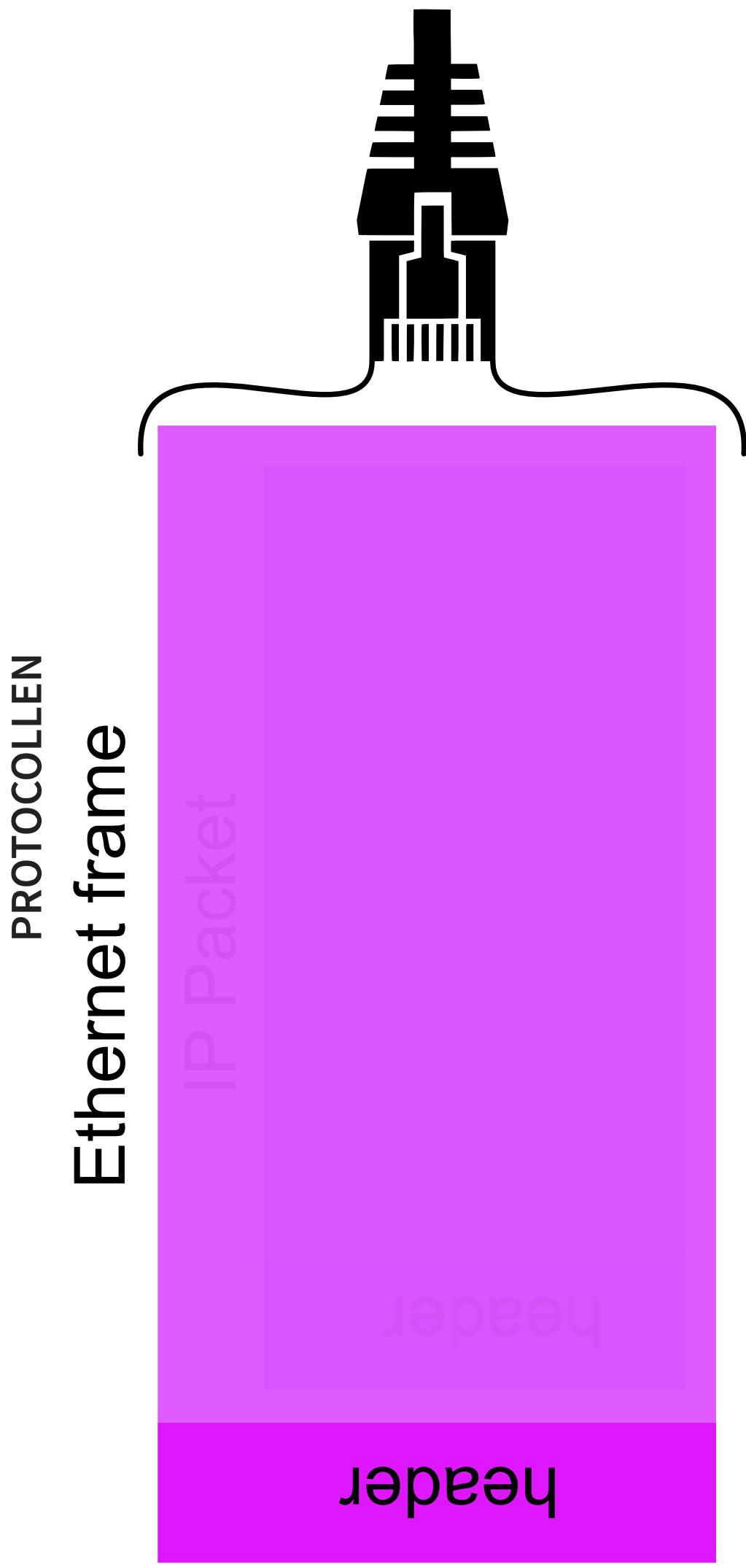
IP Packet



PROTOCOLLEN

Ethernet frame





HTTP VOORBEELDJE

```
1 POST /adfs/ls/?client-request-id=2f81...mvqawUQd4GQA1 HTTP/1.1
2 Host: sts.hhs.nl
3 Connection: keep-alive
4 Content-Length: 75
5 Pragma: no-cache
6 Cache-Control: no-cache
7 Upgrade-Insecure-Requests: 1
8 Origin: https://sts.hhs.nl
9 Content-Type: application/x-www-form-urlencoded
10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 ...
11 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/...
12 Sec-Fetch-Site: same-origin
13 Sec-Fetch-Mode: navigate
14 Sec-Fetch-User: ?1
15 Sec-Fetch-Dest: document
16 Referer: https://sts.hhs.nl/adfs/ls/?client-request-id=2f81...mvqawUQd4GQA1
17 Accept-Encoding: gzip, deflate, br
18 Accept-Language: en-US,en;q=0.9,nl;q=0.8
19 Cookie: MSISSignout=c21nbm91dENS...IjoxM30=
20 UserName=pvdbosch%40hhs.nl&Password=Voetbal123&AuthMethod=formsAuthentication
21 Body
```

request methode en url
HTTP methoden: GET, POST, PUT, DELETE, HEAD,
CONNECT, OPTIONS, ...
GET hoort niets aan te passen serverside (waarom
niet?)
PUT en DELETE worden niet zoveel meer gebruikt
CONNECT om te switchen naar HTTPS
OPTIONS om CORS te gebruiken
Request Headers

- De Content-Type bepaald hoe de body
geinterpretieerd wordt, volgens de MIME
standaard. Voorbeelden: json, xml,

HTTP VOORBEELDJE

```
1 POST /sts/lsr/2challenge-request-4d-2481... .nvqgatuod4GQA1 HTTP/1.1
2 Host: sts.hhs.nl
3 Connection: keep-alive
4 Content-Length: 75
5 Pragma: no-cache
6 Cache-Control: no-cache
7 Upgrade-Insecure-Requests: 1
8 Origin: https://sts.hhs.nl
9 Content-Type: application/x-www-form-urlencoded
10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36...
11 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/...
12 Sec-Patch-Site: same-origin
13 Sec-Patch-Code: navigate
14 Sec-Patch-User: ...
15 Sec-Patch-Dest: document
16 Referer: https://sts.hhs.nl/sake/1w/7client-request-4d-2481... .nvqgatuod4GQA1
17 Accept-Encoding: gzip, deflate, br
18 Accept-Language: en-US,en;q=0.9
19 Cookie: M3TSSignOutC21nbm9IENs...IjoxM30=
20
21 Username=pvdbasch40hs.nl&Password=vPortals123&AuthMethod=FormsAuthentication
```

- Request methode en url
- HTTP methoden: GET, POST, PUT, DELETE, HEAD, CONNECT, OPTIONS, ...
 - GET hoort niets aan te passen serverside (waarom niet?)
 - PUT en DELETE worden niet zoveel meer gebruikt
 - CONNECT om te switchen naar HTTPS
 - OPTIONS om CORS te gebruiken
 - Request Headers
 - Body
 - De Content-Type bepaald hoe de body geïnterpreteerd wordt, volgens de MIME standaard. Voorbeelden: json, xml,

HTTP VOORBEELDJE

```
1 POST /sts/ls/7client-request-4d2481...nvqabuodGQA HTTP/1.1
2 Host: sts.hhs.nl
3 Connection: keep-alive
4 Content-Length: 75
5 Pragma: no-cache
6 Cache-Control: no-cache
7 Upgrade-Insecure-Requests: 1
8 Origin: https://sts.hhs.nl
9 Content-Type: application/x-www-form-urlencoded
10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36...
11 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/...
12 Sec-Fetch-Dest: same-origin
13 Sec-Fetch-Mode: navigate
14 Sec-Fetch-Site: none
15 Sec-Fetch-User: document
16 Referer: https://sts.hhs.nl/7client-request-4d2481...nvqabuodGQA
17 Accept-Encoding: gzip, deflate, br
18 Accept-Language: en-US,en;q=0.9,fr;q=0.8
19 Cookie: .NETSISignOut=21bmQDENS...; .LcM5j0...
20
21 User-Name=pvdbosch40hhs.nl&Password=Postbaal1234AuthMethod=FormsAuthentication
```

- Request methode en url
- HTTP methoden: GET, POST, PUT, DELETE, HEAD, CONNECT, OPTIONS, ...
 - GET hoort niets aan te passen serverside (waarom niet?)
 - PUT en DELETE worden niet zoveel meer gebruikt
 - CONNECT om te switchen naar HTTPS
 - OPTIONS om CORS te gebruiken
 - Request Headers
 - Body
 - De Content-Type bepaald hoe de body geïnterpreteerd wordt, volgens de MIME standaard. Voorbeelden: json, xml,

DE REFERER HTTP HEADER

Waarom?

- analytics
- logging
- bots

Gevaarlijk: zie [dit](#). Oplossing:

- Referrer-Policy HTTP header
- referrer meta tag

DE REFERER HTTP HEADER

Waarom?

- analytics
- logging
- bots

Gevaarlijk: zie [dit](#). Oplossing:

- Referrer-Policy HTTP header
- referrer meta tag

DE REFERER HTTP HEADER

Waarom?

- analytics
- logging
- bots

Gevaarlijk: zie [dit](#). Oplossing:

- Referrer-Policy HTTP header
- referrer meta tag

DE REFERER HTTP HEADER

Waarom?

- analytics
- logging
- bots

Gevaarlijk: zie [dit](#). Oplossing:

- Referrer-Policy HTTP header
- referrer meta tag

DE REFERER HTTP HEADER

Waarom?

- analytics
- logging
- bots

Gevaarlijk: zie [dit](#). Oplossing:

- Referrer-Policy HTTP header
- referrer meta tag

DE REFERER HTTP HEADER

Waarom?

- analytics
- logging
- bots

Gevaarlijk: zie [dit](#). Oplossing:

- Referrer-Policy HTTP header
- referrer meta tag

DE REFERER HTTP HEADER

Waarom?

- analytics
- logging
- bots

Gevaarlijk: zie [dit](#). Oplossing:

- Referrer-Policy HTTP header
- referrer meta tag

VOORBEELDJE

De [Webplanner](#).

BOOTSTRAP

VOORBEREIDING

- A. Ik heb de voorbereiding geheel gedaan
- B. Ik heb de voorbereiding gedeeltelijk gedaan
- C. Ik ben de voorbereiding niet begonnen

SETUP

- bootstrap.min.js vs bootstrap.bundle.min.js
 - CSS en JS
 - /bootstrap.min.js vs
<https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/js/bootstrap.min.js>

SETUP

- bootstrap.min.js vs bootstrap.bundle.min.js
- CSS en JS
- /bootstrap.min.js vs
<https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/js/bootstrap.min.js>

SETUP

- bootstrap.min.js vs bootstrap.bundle.min.js
- CSS en JS
- /bootstrap.min.js vs
<https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/js/bootstrap.min.js>

CODE GOLF

- Wie kan de code op <https://wpfw.imfast.io/vieropeenrij.html> zo klein mogelijk maken?

MINIFICATION

- De werking van het programma is **hetzelfde**
- De grootte van het bestand wordt kleiner
- Soms: obfuscatie

SVG MINIFIERS

```
<?xml version="1.0" encoding="UTF-8"
standalone="no"?>
<!-- Created with Inkscape
(http://www.inkscape.org/) -->
<svg
xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:cc="http://creativecommons.org/ns#"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#"
xmlns:svg="http://www.w3.org/2000/svg"
xmlns="http://www.w3.org/2000/svg"
xmlns:sodipodi="http://sodipodi.sourceforge.net/D
0.dtd">
<!--:-->
<!--:-->
<!--:-->
<!--:-->
<!--:-->
```

```
<svg xmlns="http://www.w3.org/2000/svg"
width="425.488" height="126.928" viewBox="0 0
112.577 33.583" xmlns:v="https://vecta.io/nano">
<path d="M42.713 13.123h-1.125l-.708
2.016h2.4131.718 2.127h2.955l-3.564-9.322-1.236
3.547zm42.37-6.79h7.674v8.28h-
5.055v2.225h4.09v1.898h-4.09v2.912h5.1041.278
1.947h-8zm-11.093 8.05c.884.655 2.078 1.21 3.534
1.21 1.145 0 1.637-.605 1.637-1.31 0-.736-.737-
1.014-1.997-1.49-2.274-.867-3.452-1.75-3.452-
3.42 0-1.964 1.342-3.305 4.042-3.305 1.734 0
3.223.72 3.845 1.211-.982 1.75c-.638-.524-1.898-
1.014-2.863-1.014-1.13 0-1.49.524-1.49 1.13 0
.573.59.85 2.127 1.473 1.964.8 3.322 1.57 3.322
.510 0 2.004 1.807 2.207 2.256 2.256 1.000 0
.......
```

CSS: USER AGENT STYLESHEETS EN NORMALIZE.CSS

- Elke browser/OS heeft haar eigen default, bekijk deze website.
 - Deze defaults staan in de broncode van de browser.
 - Twee mogelijke oplossingen
 - CSS Reset: voorbeeld.
 - CSS Normalize: normalize.css of reboot (in Bootstrap).
- Bekijk normalize.scss in de website van de HHS.

CSS: USER AGENT STYLESHEETS EN NORMALIZE.CSS

- Elke browser/OS heeft haar eigen default, bekijk [d deze website.](#)
- Deze defaults staan in de [broncode van de browser.](#)
- Twee mogelijke oplossingen
 - CSS Reset: voorbeeld.
 - CSS Normalize: normalize.css of reboot (in Bootstrap).
- Bekijk [normalize.scss](#) in de website van de HHS.

CSS: USER AGENT STYLESHEETS EN NORMALIZE.CSS

- Elke browser/OS heeft haar eigen default, bekijk [d deze website.](#)
- Deze defaults staan in de [broncode van de browser.](#)
- Twee mogelijke oplossingen
 - CSS Reset: [voorbeeld.](#)
 - CSS Normalize: normalize.css of reboot (in Bootstrap).
- Bekijk [normalize.scss](#) in de website van de HHS.

CSS: USER AGENT STYLESHEETS EN NORMALIZE.CSS

- Elke browser/OS heeft haar eigen default, bekijk [deze](#) website.
- Deze defaults staan in de [broncode](#) van de browser.
- Twee mogelijke oplossingen
 - CSS Reset: [voorbeeld](#).
 - CSS Normalize: [normalize.css](#) of [reboot](#) (in Bootstrap).
- Bekijk [normalize.scss](#) in de website van de HHS.

CSS: USER AGENT STYLESHEETS EN NORMALIZE.CSS

- Elke browser/OS heeft haar eigen default, bekijk [deze](#) website.
- Deze defaults staan in de [broncode](#) van de browser.
- Twee mogelijke oplossingen
 - CSS Reset: [voorbeeld](#).
 - CSS Normalize: [normalize.css](#) of [reboot](#) (in Bootstrap).
- Bekijk [normalize.scss](#) in de website van de HHS.

CSS: USER AGENT STYLESHEETS EN NORMALIZE.CSS

- Elke browser/OS heeft haar eigen default, bekijk [deze](#) website.
- Deze defaults staan in de [broncode](#) van de browser.
- Twee mogelijke oplossingen
 - CSS Reset: [voorbeeld](#).
 - CSS Normalize: [normalize.css](#) of [reboot](#) (in Bootstrap).
- Bekijk [normalize.scss](#) in de website van de HHS.

CSS WAT GAAT VOOR?

Volgorde (cascading order):

- User Agent stylesheet
- Extern: <link ...> of @import ...
- Embedded: <style>...</style>
- Inline: <p style=" ... ">

Specificiteit:

- De selector met de meeste ID selectors gaat voor, bij gelijkspel:
- de selector met de meeste [(pseudo-)class selectors + attribuut selectors] gaat voor, bij gelijkspel:
- de selector met de meeste (pseudo-)element selectors gaat voor, bij gelijkspel:
- de laatst-gedefinieerde stijl gaat voor.

Dit is een 'specificiteit calculator'.

CSS WAT GAAT VOOR?

Volgorde (cascading order):

- User Agent stylesheet
- Extern: <link ...> of @import ...
- Embedded: <style>...</style>
- Inline: <p style=" ... ">

Specificiteit:

- De selector met de meeste ID selectors gaat voor, bij gelijkspel:
- de selector met de meeste [(pseudo-)class selectors + attribuut selectors] gaat voor, bij gelijkspel:
- de selector met de meeste (pseudo-)element selectors gaat voor, bij gelijkspel:
- de laatst-gedefinieerde stijl gaat voor.

Dit is een 'specificiteit calculator'.

CSS WAT GAAT VOOR?

Volgorde (cascading order):

- User Agent stylesheet
- Extern: <link ...> of @import ...
- Embedded: <style>...</style>
- Inline: <p style=" ... ">

Specificiteit:

- De selector met de meeste ID selectors gaat voor, bij gelijkspel:
- de selector met de meeste [(pseudo-)class selectors + attribuut selectors] gaat voor, bij gelijkspel:
- de selector met de meeste (pseudo-)element selectors gaat voor, bij gelijkspel:
- de laatst-gedefinieerde stijl gaat voor.

Dit is een 'specificiteit calculator'.

CSS WAT GAAT VOOR?

Volgorde (cascading order):

- User Agent stylesheet
- Extern: <link ...> of @import ...
- Embedded: <style>...</style>
- Inline: <p style=" ... ">

Specificiteit:

- De selector met de meeste ID selectors gaat voor, bij gelijkspel:
- de selector met de meeste [(pseudo-)class selectors + attribuut selectors] gaat voor, bij gelijkspel:
- de selector met de meeste (pseudo-)element selectors gaat voor, bij gelijkspel:
- de laatst-gedefineerde stijl gaat voor.

Dit is een 'specificiteit calculator'.

CSS WAT GAAT VOOR?

Volgorde (cascading order):

- User Agent stylesheet
- Extern: <link ...> of @import ...
- Embedded: <style>...</style>
- Inline: <p style="...">

Specificiteit:

- De selector met de meeste ID selectors gaat voor, bij gelijkspel:
- de selector met de meeste [(pseudo-)class selectors + attribuut selectors] gaat voor, bij gelijkspel:
- de selector met de meeste (pseudo-)element selectors gaat voor, bij gelijkspel:
- de laatst-gedefinieerde stijl gaat voor.

Dit is een 'specificiteit calculator'.

CSS WAT GAAT VOOR?

Volgorde (cascading order):

- User Agent stylesheet
- Extern: <link ...> of @import ...
- Embedded: <style>...</style>
- Inline: <p style="...">

Specificiteit:

- De selector met de meeste ID selectors gaat voor, bij gelijkspel:
- de selector met de meeste [(pseudo-)class selectors + attribuut selectors] gaat voor, bij gelijkspel:
- de selector met de meeste (pseudo-)element selectors gaat voor, bij gelijkspel:
- de laatst-gedefineerde stijl gaat voor.

Dit is een 'specificiteit calculator'.

CSS WAT GAAT VOOR?

Volgorde (cascading order):

- User Agent stylesheet
- Extern: <link ...> of @import ...
- Embedded: <style>...</style>
- Inline: <p style="...">

Specificiteit:

- De selector met de meeste ID selectors gaat voor, bij gelijkspel:
 - de selector met de meeste [(pseudo-)class selectors + attribuut selectors] gaat voor, bij gelijkspel:
 - de selector met de meeste (pseudo-)element selectors gaat voor, bij gelijkspel:
 - de laatst-gedefinieerde stijl gaat voor.

Dit is een 'specificiteit calculator'.

CSS WAT GAAT VOOR?

Volgorde (cascading order):

- User Agent stylesheet
- Extern: <link ...> of @import ...
- Embedded: <style>...</style>
- Inline: <p style=" ... ">

Specificiteit:

- De selector met de meeste ID selectors gaat voor, bij gelijkspel:
- de selector met de meeste [(pseudo-)class selectors + attribuut selectors] gaat voor, bij gelijkspel:
 - de selector met de meeste (pseudo-)element selectors gaat voor, bij gelijkspel:
 - de laatst-gedefinieerde stijl gaat voor.

Dit is een 'specificiteit calculator'.

CSS WAT GAAT VOOR?

Volgorde (cascading order):

- User Agent stylesheet
- Extern: <link ...> of @import ...
- Embedded: <style>...</style>
- Inline: <p style=" ... ">

Specificiteit:

- De selector met de meeste ID selectors gaat voor, bij gelijkspel:
- de selector met de meeste [(pseudo-)class selectors + attribuut selectors] gaat voor, bij gelijkspel:
- de selector met de meeste (pseudo-)element selectors gaat voor, bij gelijkspel:
- de laatst-gedefineerde stijl gaat voor.

Dit is een 'specificiteit calculator'.

CSS WAT GAAT VOOR?

Volgorde (cascading order):

- User Agent stylesheet
- Extern: <link ...> of @import ...
- Embedded: <style>...</style>
- Inline: <p style=" ... ">

Specificiteit:

- De selector met de meeste ID selectors gaat voor, bij gelijkspel:
- de selector met de meeste [(pseudo-)class selectors + attribuut selectors] gaat voor, bij gelijkspel:
- de selector met de meeste (pseudo-)element selectors gaat voor, bij gelijkspel:
- de laatst-gedefineerde stijl gaat voor.

Dit is een 'specificiteit calculator'.

CSS WAT GAAT VOOR?

Volgorde (cascading order):

- User Agent stylesheet
- Extern: <link ...> of @import ...
- Embedded: <style>...</style>
- Inline: <p style=" ... ">

Specificiteit:

- De selector met de meeste ID selectors gaat voor, bij gelijkspel:
- de selector met de meeste [(pseudo-)class selectors + attribuut selectors] gaat voor, bij gelijkspel:
- de selector met de meeste (pseudo-)element selectors gaat voor, bij gelijkspel:
- de laatst-gedefineerde stijl gaat voor.

Dit is een 'specificiteit calculator'.

FRAMEWORKS

Wat voor soort frameworks zijn er?

Front-end frameworks/technologieen:

- **Bootstrap**: puur CSS of CSS + JS (+ jQuery!).
- **Foundation**: puur CSS of CSS + JS.
- **Bulma**: puur CSS
- **Razor / Blazor**
- **React**: slim render(), zie JSX en **babel**.
- **Vue**
- **Angular**

Waarom? Leesbare onderhoudbare code, responsiveness. Soms: herbruikbaarheid componenten.

FRAMEWORKS

Wat voor soort frameworks zijn er?

- **Front-end en back-end is scheiding op basis van separation of concerns**
- Vooruitblik 3-lagen architectuur: presentatie laag, applicatie laag, database laag
- **Client-side en server-side** is fysieke scheiding. Server-side frameworks genereren HTML.
- **Een puur CSS / CSS + JS / gedeeltelijk server-side** framework.

Front-end frameworks/technologieen:

- Bootstrap: puur CSS of CSS + JS (+ jQuery!).
- Foundation: puur CSS of CSS + JS.
- Bulma: puur CSS
- Razor / Blazor
- React: slim render(), zie JSX en babel.
- Vue
- Angular

Waarom? Leesbare onderhoudbare code, responsiveness. Soms: herbruikbaarheid componenten.

FRAMEWORKS

Wat voor soort frameworks zijn er?

- **Front-end en back-end** is scheiding op basis van **separation of concerns**
- Vooruitblik **3-lagen architectuur**: presentatie laag, applicatie laag, database laag
- Client-side en server-side is fysieke scheiding. Server-side frameworks genereren HTML.
- Een puur CSS / CSS + JS / **gedeeltelijk server-side** framework.

Front-end frameworks/technologieen:

- Bootstrap: puur CSS of CSS + JS (+ jQuery!).
- Foundation: puur CSS of CSS + JS.
- Bulma: puur CSS
- Razor / Blazor
- React: slim render(), zie JSX en babel.
- Vue
- Angular

Waarom? Leesbare onderhoudbare code, responsiveness. Soms: herbruikbaarheid componenten.

FRAMEWORKS

Wat voor soort frameworks zijn er?

- **Front-end en back-end** is scheiding op basis van **separation of concerns**
- Vooruitblik **3-lagen architectuur**: presentatie laag, applicatie laag, database laag
- **Client-side en server-side** is fysieke scheiding. Server-side frameworks genereren HTML.
- Een puur CSS / CSS+JS / **gedeeltelijk server-side** framework.

Front-end frameworks/technologieen:

- Bootstrap: puur CSS of CSS + JS (+ jQuery!).
- Foundation: puur CSS of CSS + JS.
- Bulma: puur CSS
- Razor / Blazor
- React: slim render(), zie JSX en babel.
- Vue
- Angular

Waarom? Leesbare onderhoudbare code, responsiveness. Soms: herbruikbaarheid componenten.

FRAMEWORKS

Wat voor soort frameworks zijn er?

- **Front-end en back-end** is scheiding op basis van **separation of concerns**
- Vooruitblik **3-lagen architectuur**: presentatie laag, applicatie laag, database laag
- **Client-side en server-side** is fysieke scheiding. Server-side frameworks genereren HTML.
- Een puur CSS / CSS + JS / **gedeeltelijk server-side** framework.

Front-end frameworks/technologieen:

- Bootstrap: puur CSS of CSS + JS (+ jQuery!).
- Foundation: puur CSS of CSS + JS.
- Bulma: puur CSS
- Razor / Blazor
- React: slim render(), zie JSX en babel.
- Vue
- Angular

Waarom? Leesbare onderhoudbare code, responsiveness. Soms: herbruikbaarheid componenten.

FRAMEWORKS

Wat voor soort frameworks zijn er?

- **Front-end en back-end** is scheiding op basis van **separation of concerns**
- Vooruitblik **3-lagen architectuur**: presentatie laag, applicatie laag, database laag
- **Client-side en server-side** is fysieke scheiding. Server-side frameworks genereren HTML.
- Een puur CSS / CSS + JS / **gedeeltelijk server-side** framework.

Front-end frameworks/technologieen:

- Bootstrap: puur CSS of CSS + JS (+ jQuery!).
- Foundation: puur CSS of CSS + JS.
- Bulma: puur CSS
- Razor / Blazor
- React: slim render(), zie JSX en babel.
- Vue
- Angular

Waarom? Leesbare onderhoudbare code, responsiveness. Soms: herbruikbaarheid componenten.

FRAMEWORKS

Wat voor soort frameworks zijn er?

- **Front-end en back-end** is scheiding op basis van **separation of concerns**
- Vooruitblik **3-lagen architectuur**: presentatie laag, applicatie laag, database laag
- **Client-side en server-side** is fysieke scheiding. Server-side frameworks genereren HTML.
- Een puur CSS / CSS + JS / **gedeeltelijk server-side** framework.

Front-end frameworks/technologieen:

- **Bootstrap**: puur CSS of CSS + JS (+ jQuery!).
- **Foundation**: puur CSS of CSS + JS
- **Bulma**: puur CSS
- **Razor / Blazor**
- **React**: slim render(), zie JSX en **babel**.
- **Vue**
- **Angular**

Waarom? Leesbare onderhoudbare code, responsiveness. Soms: herbruikbaarheid componenten.

FRAMEWORKS

Wat voor soort frameworks zijn er?

- **Front-end en back-end** is scheiding op basis van **separation of concerns**
- Vooruitblik **3-lagen architectuur**: presentatie laag, applicatie laag, database laag
- **Client-side en server-side** is fysieke scheiding. Server-side frameworks genereren HTML.
- Een puur CSS / CSS + JS / **gedeeltelijk server-side** framework.

Front-end frameworks/technologieen:

- **Bootstrap**: puur CSS of CSS + JS (+ jQuery!).
- **Foundation**: puur CSS of CSS + JS.
- Bulma: puur CSS
- Razor / Blazor
- React: slim render(), zie JSX en babel.
- Vue
- **Angular**

Waarom? Leesbare onderhoudbare code, responsiveness. Soms: herbruikbaarheid componenten.

FRAMEWORKS

Wat voor soort frameworks zijn er?

- **Front-end en back-end** is scheiding op basis van **separation of concerns**
- Vooruitblik **3-lagen architectuur**: presentatie laag, applicatie laag, database laag
- **Client-side en server-side** is fysieke scheiding. Server-side frameworks genereren HTML.
- Een puur **CSS + JS / gedeeltelijk server-side** framework.

Front-end frameworks/technologieen:

- **Bootstrap**: puur CSS of CSS + JS (+ jQuery!).
- **Foundation**: puur CSS of CSS + JS.
- **Bulma**: puur CSS
- **Razor / Blazor**
- React: slim render(), zie JSX en **babel**.
- Vue
- **Angular**

Waarom? Leesbare onderhoudbare code, responsiveness. Soms: herbruikbaarheid componenten.

FRAMEWORKS

Wat voor soort frameworks zijn er?

- **Front-end en back-end** is scheiding op basis van **separation of concerns**
- Vooruitblik **3-lagen architectuur**: presentatie laag, applicatie laag, database laag
- **Client-side en server-side** is fysieke scheiding. Server-side frameworks genereren HTML.
- Een puur CSS / CSS + JS / **gedeeltelijk server-side** framework.

Front-end frameworks/technologieen:

- **Bootstrap**: puur CSS of CSS + JS (+ jQuery!).
- **Foundation**: puur CSS **of** CSS + JS.
- **Bulma**: puur CSS
- **Razor / Blazor**
- React: **slim render()**, zie JSX en **babel**.
- Vue
- **Angular**

Waarom? Leesbare onderhoudbare code, responsiveness. Soms: herbruikbaarheid componenten.

FRAMEWORKS

Wat voor soort frameworks zijn er?

- **Front-end en back-end** is scheiding op basis van **separation of concerns**
- Vooruitblik **3-lagen architectuur**: presentatie laag, applicatie laag, database laag
- **Client-side en server-side** is fysieke scheiding. Server-side frameworks genereren HTML.
- Een **puur CSS / CSS + JS / gedeeltelijk server-side** framework.

Front-end frameworks/technologieen:

- **Bootstrap**: puur CSS of CSS + JS (+ jQuery!).
- **Foundation**: puur CSS **of** CSS + JS.
- **Bulma**: puur CSS
- **Razor / Blazor**
- **React**: **slim render()**, zie JSX en **babel**.
- **Vue**
- **Angular**

Waarom? Leesbare onderhoudbare code, responsiveness. Soms: herbruikbaarheid componenten.

FRAMEWORKS

Wat voor soort frameworks zijn er?

- **Front-end en back-end** is scheiding op basis van **separation of concerns**
- Vooruitblik **3-lagen architectuur**: presentatie laag, applicatie laag, database laag
- **Client-side en server-side** is fysieke scheiding. Server-side frameworks genereren HTML.
- Een puur CSS / CSS + JS / **gedeeltelijk server-side** framework.

Front-end frameworks/technologieen:

- **Bootstrap**: puur CSS of CSS + JS (+ jQuery!).
- **Foundation**: puur CSS **of** CSS + JS.
- **Bulma**: puur CSS
- **Razor / Blazor**
- **React**: slim render(), zie JSX en **babel**.
- **Vue**
- **Angular**

Waarom? Leesbare onderhoudbare code, responsiveness. Soms: herbruikbaarheid componenten.

FRAMEWORKS

Wat voor soort frameworks zijn er?

- **Front-end en back-end** is scheiding op basis van **separation of concerns**
- Vooruitblik **3-lagen architectuur**: presentatie laag, applicatie laag, database laag
- **Client-side en server-side** is fysieke scheiding. Server-side frameworks genereren HTML.
- Een puur CSS / CSS + JS / **gedeeltelijk server-side** framework.

Front-end frameworks/technologieen:

- **Bootstrap**: puur CSS of CSS + JS (+ jQuery!).
- **Foundation**: puur CSS **of** CSS + JS.
- **Bulma**: puur CSS
- **Razor** / **Blazor**
- React: **slim render()**, zie JSX en **babel**.
- **Vue**
- **Angular**

Waarom? Leesbare onderhoudbare code, responsiveness. Soms: herbruikbaarheid componenten.

FRAMEWORKS

Wat voor soort frameworks zijn er?

- **Front-end en back-end** is scheiding op basis van **separation of concerns**
- Vooruitblik **3-lagen architectuur**: presentatie laag, applicatie laag, database laag
- **Client-side en server-side** is fysieke scheiding. Server-side frameworks genereren HTML.
- Een **puur CSS / CSS + JS / gedeeltelijk server-side** framework.

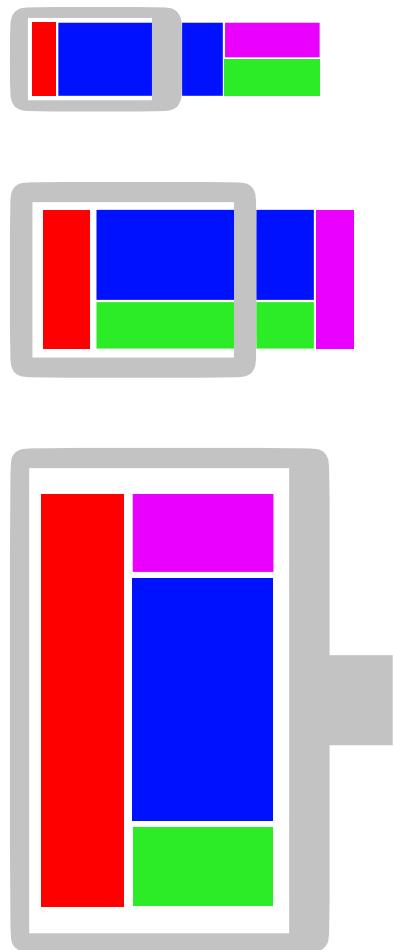
Front-end frameworks/technologieen:

- **Bootstrap**: puur CSS of CSS + JS (+ jQuery!).
- **Foundation**: puur CSS **of** CSS + JS.
- **Bulma**: puur CSS
- **Razor / Blazor**
- React: **slim render()**, zie JSX en **babel**.
- **Vue**
- **Angular**

Waarom? Leesbare onderhoudbare code, responsiveness. Soms: herbruikbaarheid componenten.

RESPONSIVENESS: HET PROBLEEM

Probleem:



Zie ook [dit](#).

RESPONSIVENESS: DE OPLOSSINGEN

Oplossingen:

- CSS eenheden: absoluut (cm, in, px, pt) vs relatief (em, vw, %)
 - [Viewport](#):

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```
 - Flexbox
 - Media Queries: in CSS

```
@media only screen and (max-width: 600px) {  
    body {  
        background-color: lightblue;  
    }  
}
```
- of bij het laden van CSS
- ```
<link rel="stylesheet" media="screen and (min-width: 900px)" href="widescreen.css">
<link rel="stylesheet" media="screen and (max-width: 600px)" href="smallscreen.css">
```
- Zie ook [light-level en inverted-colors en scripting](#).
- Bootstrap, of een ander front-end framework

# RESPONSIVENESS: DE OPLOSSINGEN

## Oplossingen:

- CSS eenheden: absoluut (cm, in, px, pt) vs relatief (em, vw, %)
- **Viewport:**

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

- Flexbox
- Media Queries: in CSS

```
@media only screen and (max-width: 600px) {
 body {
 background-color: lightblue;
 }
}
```

of bij het laden van CSS

```
<link rel="stylesheet" media="screen and (min-width: 900px)" href="widescreen.css"
<link rel="stylesheet" media="screen and (max-width: 600px)" href="smallscreen.css"
```

Zie ook **light-level** en **inverted-colors** en **scripting**.

- Bootstrap, of een ander front-end framework

# RESPONSIVENESS: DE OPLOSSINGEN

## Oplossingen:

- CSS eenheden: absoluut (cm, in, px, pt) vs relatief (em, vw, %)
- [\*\*Viewport:\*\*](#)

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```
- [\*\*Flexbox\*\*](#)
- Media Queries: in CSS

```
@media only screen and (max-width: 600px) {
 body {
 background-color: lightblue;
 }
}
```
- of bij het laden van CSS

```
<link rel="stylesheet" media="screen and (min-width: 900px)" href="widescreen.css">
<link rel="stylesheet" media="screen and (max-width: 600px)" href="smallscreen.css">
```

Zie ook [light-level en inverted-colors](#) en [scripting](#).
- Bootstrap, of een ander front-end framework

# RESPONSIVENESS: DE OPLOSSINGEN

## Oplossingen:

- CSS eenheden: absoluut (cm, in, px, pt) vs relatief (em, vw, %)
- [Viewport](#):
- `<meta name="viewport" content="width=device-width, initial-scale=1.0">`
- Flexbox
- Media Queries: in CSS

```
@media only screen and (max-width: 600px) {
 body {
 background-color: lightblue;
 }
}
```

of bij het laden van CSS

```
<link rel="stylesheet" media="screen and (min-width: 900px)" href="widescreen.css">
<link rel="stylesheet" media="screen and (max-width: 600px)" href="smallscreen.css">
```

Zie ook **light-level en inverted-colors en scripting**.

- Bootstrap, of een ander front-end framework

# RESPONSIVENESS: DE OPLOSSINGEN

## Oplossingen:

- CSS eenheden: absoluut (cm, in, px, pt) vs relatief (em, vw, %)
- [Viewport](#):  

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```
- Flexbox
- Media Queries: in CSS  

```
@media only screen and (max-width: 600px) {
 body {
 background-color: lightblue;
 }
}
```
- of bij het laden van CSS  

```
<link rel="stylesheet" media="screen and (min-width: 900px)" href="widescreen.css">
<link rel="stylesheet" media="screen and (max-width: 600px)" href="smallscreen.css">
```

Zie ook [light-level en inverted-colors en scripting](#).
- Bootstrap, of een ander front-end framework

## WAT KAN JE MET BOOTSTRAP?

- Een responsive layout
  - Extra voor-gedefinieerde componenten
  - Syntactic sugar: korte namen
  - Content: tabellen, code, figuren met onderschriften

## WAT KAN JE MET BOOTSTRAP?

- Een responsive layout
- Extra voor-gedefinieerde componenten
- Syntactic sugar: korte namen
- Content: tabellen, code, figuren met onderschriften

## WAT KAN JE MET BOOTSTRAP?

- Een responsive layout
- Extra voor-gedefinieerde componenten
- Syntactic sugar: korte namen
- Content: tabellen, code, figuren met onderschriften

## WAT KAN JE MET BOOTSTRAP?

- Een responsive layout
- Extra voor-gedefinieerde componenten
- Syntactic sugar: korte namen
- Content: tabellen, code, figuren met onderschriften

# LAYOUT

We kennen al:

- de CSS stijl eigenschap **position**
- de CSS stijl eigenschap **display: grid, flexbox**

In bootstrap:

- De container zorgt voor padding en voor een 'vaste' **breedte**, gecentered.
- Het grid-systeem met  
ROW  
en col-...-.

# LAYOUT

We kennen al:

- de CSS stijl eigenschap **position**
- de CSS stijl eigenschap **display**: grid, flexbox

In bootstrap:

- De container zorgt voor padding en voor een 'vaste' **breedte**, gecentred.
- Het grid-systeem met  
`row`  
en `col-...-`.

# LAYOUT

We kennen al:

- de CSS stijl eigenschap `position`
- de CSS stijl eigenschap `display: grid, flexbox`

In bootstrap:

- De `container` zorgt voor padding en voor een 'vaste' `breedte`, gecentered.
- Het grid-systeem met  
`row`  
en `col-...-`.

# LAYOUT

We kennen al:

- de CSS stijl eigenschap `position`
- de CSS stijl eigenschap `display: grid, flexbox`

In bootstrap:

- De container zorgt voor padding en voor een 'vaste' `breedte`, gecentered.
- Het grid-systeem met  
`row`  
en `col-...-`.

## EXTRA COMPONENTEN

Niet zelf het wiel opnieuw uitwinden!

- Alerts
- Jumbotron
- Navbar
- Carousel
- Button group
- Etc.

## EXTRA COMPONENTEN

Niet zelf het wiel opnieuw uitwinden!

- *Alerts*
- Jumbotron
- Navbar
- Carousel
- Button group
- Etc.

## EXTRA COMPONENTEN

Niet zelf het wiel opnieuw uitwinden!

- *Alerts*
- Jumbotron
- Navbar
- Carousel
- Button group
- Etc.

## EXTRA COMPONENTEN

Niet zelf het wiel opnieuw uitwinden!

- *Alerts*
- Jumbotron
- Navbar
- Carousel
- Button group
- Etc.

## EXTRA COMPONENTEN

Niet zelf het wiel opnieuw uitwinden!

- *Alerts*
- Jumbotron
- Navbar
- *Carousel*
- Button group
- Etc.

## EXTRA COMPONENTEN

Niet zelf het wiel opnieuw uitwinden!

- *Alerts*
- Jumbotron
- Navbar
- Carousel
- Button group
- Etc.

## EXTRA COMPONENTEN

Niet zelf het wiel opnieuw uitwinden!

- *Alerts*
- Jumbotron
- Navbar
- Carousel
- Button group
- Etc.

# ACCESIBILITY

Houd **continu** rekening met slechtzienden:

- gebruik semantische HTML elementen
- gebruik headers
- aria-attributen
- alternatieven voor iconen/plaatjes

## ACCESIBILITY

Houd **continu** rekening met slechtzienden:

- gebruik semantische HTML elementen
  - gebruik headers
  - aria-attributen
  - alternatieven voor iconen/plaatjes

## ACCESIBILITY

Houd **continu** rekening met slechtzienden:

- gebruik semantische HTML elementen
- gebruik headers
- aria-attributen
- alternatieven voor iconen/plaatjes

## ACCESIBILITY

Houd **continu** rekening met slechtzienden:

- gebruik semantische HTML elementen
- gebruik headers
- aria-attributen
- alternatieven voor iconen/plaatjes

## ACCESIBILITY

Houd **continu** rekening met slechtzienden:

- gebruik semantische HTML elementen
- gebruik headers
- aria-attributen
- alternatieven voor iconen/plaatjes

## ANIMATIES

- met Javascript en setTimeout?
- SMIL
  - **“Although Chrome 45 deprecated SMIL in favor of CSS animations and Web animations, the Chrome developers have since suspended that deprecation.**
    - [MDN web docs](#)
  - Edge tot 2020 geen ondersteuning, nu gedeeltelijk
    - Polyfillers als [fakesmile](#).

**GOED OPGELET?**

## GOED OPGELET?

Welk attribuut is ingesteld voor child 3?

- A. padding: 10px;
- B. margin: 10px;
- C. border-width: 10px;
- D. margin: -10px;



## GOED OPGELET?

Welk attribuut is ingesteld voor child 2?

- A. padding: 10px;
- B. margin: 10px;
- C. border-width: 10px;
- D. margin: -10px;



## GOED OPGELET?

Stel: tekst past niet in de ruimte die beschikbaar is voor een div. Hoe maak je div scrollable?

- A. border-width: 10px;
- B. margin: 10px;
- C. padding: 10px;
- D. overflow: auto;

## GOED OPGELET?

Hoe wordt div met class parent op een telefoon met normale oriëntatie en 400px breed getoond?

- A. Met witte achtergrond.
- B. Met default achtergrond.
- C. Met blauwe achtergrond.
- D. Dat kan niet worden bepaald.

```
33 .parent {
34 background-color: white;
35 }
36 @media only screen and (min-width: 500px)
37 and (orientation: landscape) {
38 .parent {
39 background-color: blue;
40 }
41 }
42 }
```

## GOED OPGELET?

Hoe wordt div met class parent op een gekantelde telefoon en 400px breed getoond?

- A. Met witte achtergrond.
- B. Met default achtergrond.
- C. Met blauwe achtergrond.
- D. Dat kan niet worden bepaald.

```
33 .parent {
34 background-color: white;
35 }
36
37 @media only screen and (min-width: 500px)
38 and (orientation: landscape) {
39 .parent {
40 background-color: blue;
41 }
42 }
```

## GOED OPGELET?

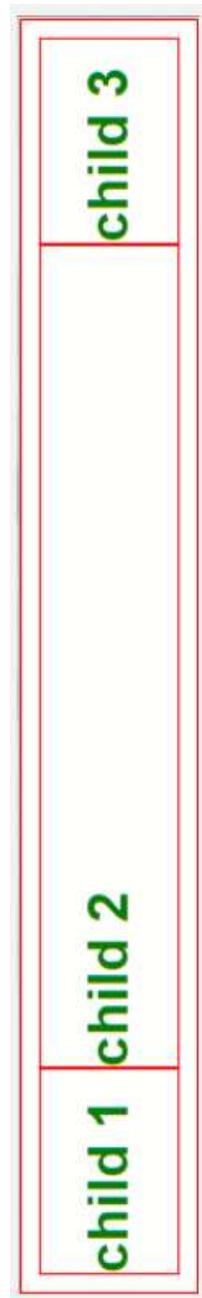
- Wat is het effect van het opnemen van een link in je HTML naar 'normalize.css'?
- A. HTML wordt in alle browsers op alle apparaten gelijk getoond
  - B. Alle plaatjes worden even groot afgebeeld.
  - C. De verhoudingen van plaatjes worden genormaliseerd.
  - D. HTML wordt kleiner gemaakt, zodat het helemaal wordt getoond



## GOED OPGELET?

Welke class in 12-koloms-framework (met col-1 t/m col-12 en row) is van toepassing op child 1?

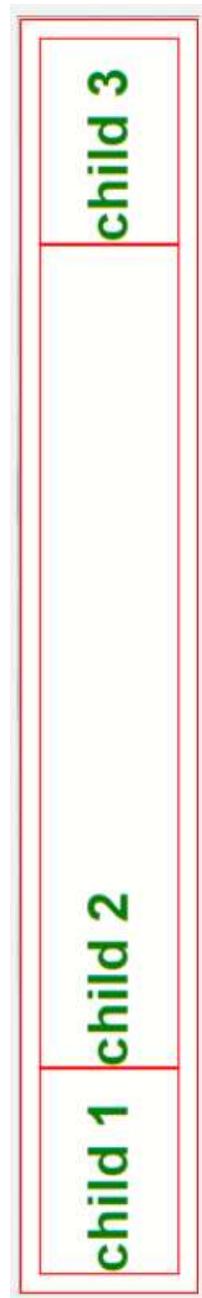
- A. col-2
- B. col-8
- C. row
- D. col-12



## GOED OPGELET?

Welke class in 12-koloms-framework is van toepassing op de parent van child 1 t/m 3?

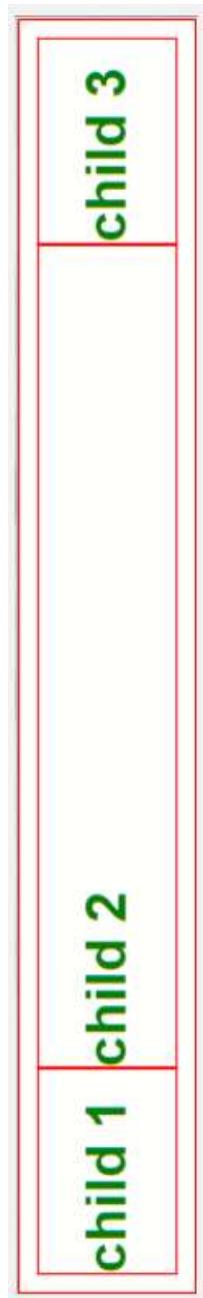
- A. col-2
- B. col-8
- C. row
- D. col-12



## GOED OPGELET?

Welke class in Bootstrap is op een normale telefoon toegepast op de parent van child 1 t/m 3?

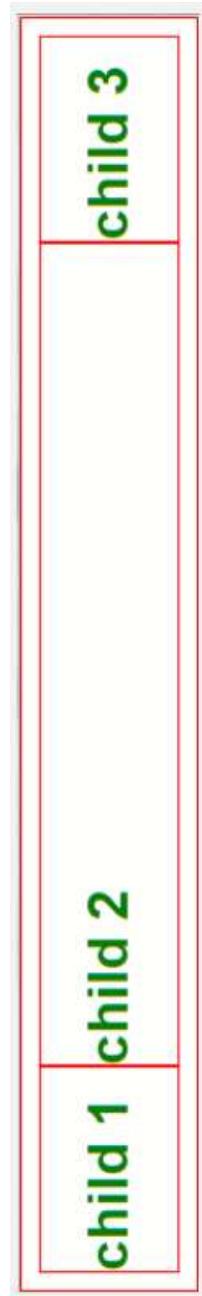
- A. row
- B. col-md
- C. col-sm
- D. col-xs



## GOED OPGELET?

Welke class in Bootstrap is op een normale telefoon toegepast op div met tekst child 1?

- A. col-xs-2
- B. col-md-2
- C. col-sm-2
- D. col-lg-2



## GOED OPGELET?

Wat is het effect van het minimized opslaan van CSS of JavaScript?

- A. Dat bestand wordt gezipped.
- B. Dat veel moeilijker leesbaar wordt.
- C. Dat bestand met CSS of JavaScript groter wordt.
- D. Dat file met CSS/JavaScript meer downloadtijd nodig heeft.

## GOED OPGELET?

Hoe refereer je in HTML aan een minimized version van je CSS-file /css/bootstrap.css?

- A. <link rel="stylesheet" href="/css/bootstrap.min.css">
- B. <link rel="minimized" href="/css/bootstrap.css">
- C. <link rel="minimized" href="/css/bootstrap.min.css">
- D. <link rel="stylesheet" href="/css/minimized.css">

## GOED OPGELET?

Wat ontbreekt er in de volgende code in de body (lees classes als verwijzing naar Bootstrap)?

- A. De verwijzing in <body> naar Bootstrap.
- B. <link rel="minimized" href="/css/boostrap.css">
- C. div met class 'container' (in de DOM direct onder <body>).
- D. <link rel="stylesheet" href="/css/minimized.css">

```
8 <body>
9 <div class="row">
10 <div class="col-md-2">kolom 1</div>
11 <div class="col-md-10">kolom 2</div>
12 </div>
13 </body>
```

## GOED OPGELET?

Hoe zorg ik dat plaatje hieronder zich m.b.v. Bootstrap responsive aanpast aan de breedte?

- A. Door aan <img...> 'style="max-width: 400px;"" toe te voegen.
- B. Door aan <img...> 'class="img-responsive"" toe te voegen.
- C. Door aan <img...> 'style="display: flex;"" toe te voegen.
- D. Door aan <img...> 'style="height: auto;" toe te voegen.

```
19 <div class="col-md-12">
20
21 </div>
```

## GOED OPGELET?

Hoe zorg ik dat alle tekst hieronder met één wijziging rechts wordt uitgelijnd?

- A. Door '<div class="col-md-6 text-right">' uit te breiden.
- B. Door '<h1 class="text-right">' uit te breiden.
- C. Door aan '<div...>' 'class="text-right"' toe te voegen.
- D. Door '<h3 class="text-right">' uit te breiden.

```
13 <div class="col-md-6">
14 <h1>Jane Doette</h1>
15 <h3>Front-end Ninja</h3>
16 </div>
```

## GOED OPGELET?

Als je div met class 'eigen-opmaak' op wilt maken, welk attribuut definieer je dan als eerste?

- A. background-color
- B. De volgorde van attributen maakt nauwelijks iets uit.
- C. margin en padding
- D. font styles

```
15 <div class="col-md-6 eigen-opmaak">
16 <h1>Jane Doette</h1>
17 <h3>Front-end Ninja</h3>
18 </div>
```

## GOED OPGELET?

Wat is de achtergrondkleur van een div met class 'tweede'?

- A. black
- B. red
- C. yellow
- D. blue

```
15 * {background-color: black; }
16 body { background-color: yellow; }
17 div { background-color: red; }
18 .tweede { background-color: blue; }
```

## GOED OPGELET?

Wat is de achtergrondkleur van een div met class 'tweede'?

- A. Dat kan op basis van deze code niet worden bepaald.
- B. white
- C. De default color.
- D. red

```
15 .tweede {
16 background-color: white;
17 }
18 .tweede {
19 background-color: red;
20 }
```

## GOED OPGELET?

In welke file leg je opmaak van de div met class 'eigen-opmaak' vast (kies het beste antwoord)?

- A. In de file 'css/bootstrap.css'
- B. In de file 'css/bootstrap.min.css'
- C. In een eigen file (bijv. 'style.css') in directory 'css'.
- D. in HTML-file in head met tussen tags voor '<style>'.

```
15 <div class="col-md-6 eigen-opmaak">
16 <h1>Jane Doette</h1>
17 <h3>Front-end Ninja</h3>
18 </div>
```

## GOED OPGELET?

Als je Bootstrap-classes wilt overladen in style.css. In welke volgorde neem je <link ...> op?

- A. <link href="style.css"> na <link href="bootstrap.css">.
- B. <link href="bootstrap.css"> na <link href="style.css">.
- C. De volgorde maakt niet uit.
- D. Je kunt classes in bootstrap.css niet overladen.

## GOED OPGELET?

Welke extra file heb je nodig om Modal Popup van Bootstrap te kunnen tonen?

- A. Bootstrap maakt daarvoor gebruik van jquery.min.js.
- B. De css van Bootstrap is noodzakelijk.
- C. Je moet zelf nog een regel code schrijven voor modal popup.
- D. Er zijn geen extra files nodig om deze popup te tonen.

# INLEIDING OPDRACHT

De [Webplanner](#) van de Haagse Hogeschool is een goed voorbeeld van een erg slechte front-end. Herontwerp de website eerst in een wireframe, en daarna in Bootstrap.

Wat is er bijvoorbeeld allemaal mis?

- de website is nu niet responsive (ooit op je mobiel het rooster proberen te bekijken?). Gebruik de Chrome DevTools om de website te bekijken vanaf je mobiel.
- de website vrijwel onmogelijk met een screenreader te gebruiken.
- het stappenplan in de gele balk doet waar hij zin in heeft
- het ontwerp is onnodig ingewikkeld: waarom zit er onder de gele tips een slider waarmee je de grootte van de banner kunt aanpassen? Het aanmaken van een filter (groene knop naast "Geen filter toepassen") is onnodig ingewikkeld.

Gebruik namaak-data als voorbeeld in de website, dus je hoeft niet gebruik te maken van de API van WebPlanner. Maak gebruik van een container, rows met cols en zorg dat er van de volgende componenten tenminste 1 wordt gebruikt:

- ...

Uitdagende mogelijkheden tot uitbreiding...

## **VOORUITBLIK**

Volgende week: (eindelijk) C#