

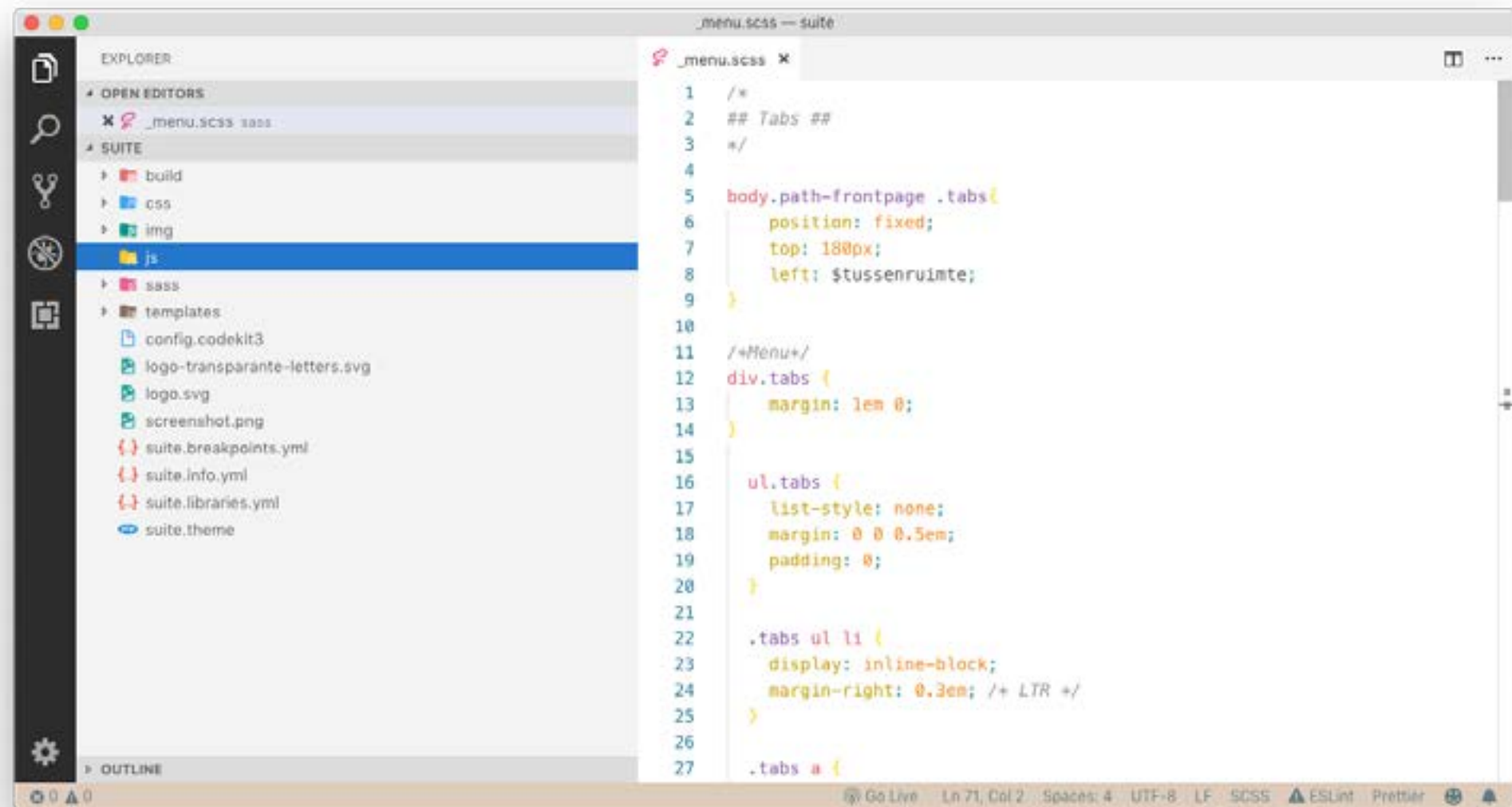
# HTML5 & CSS

*De Basis van HTML & CSS*

---

# CODE EDITOR

Programma om HTML, CSS of JS in te schrijven.



# CONTROLEER JE HTML (EN CSS)

Je kan de W3C validator gebruiken:

- 1 - URL in de website plakken
- 2 - Je HTML bestand uploaden
- 3 - Je code in het text-veld plakken

Je wil altijd een foutloos HTML bestand.

W3C® Markup Validation Service  
Check the markup (HTML, XHTML, ...) of Web documents

Validate by URI   Validate by File Upload   Validate by Direct Input

Validate by URI

Validate a document online:

Address:

► More Options

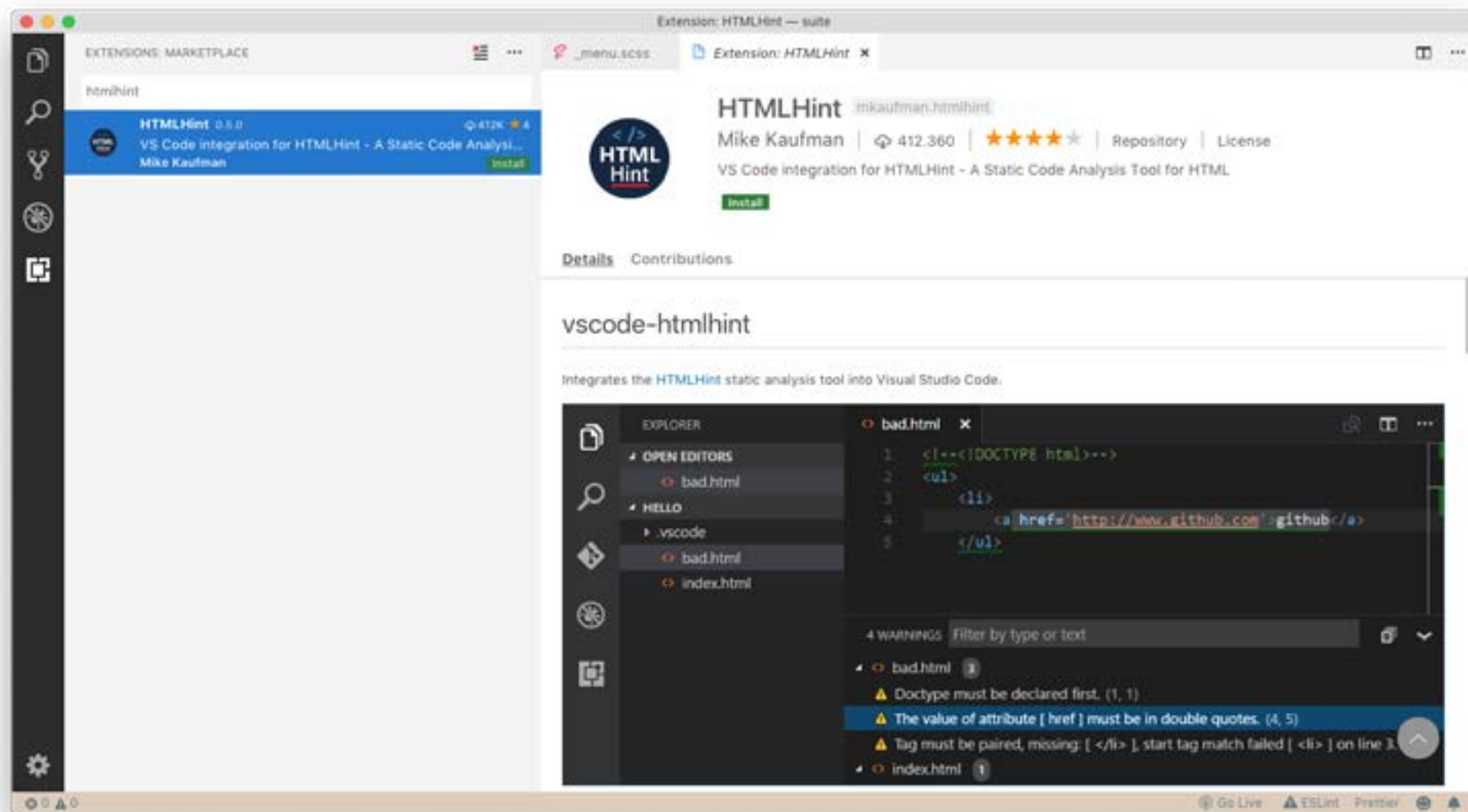
Check

This validator checks the [markup validity](#) of Web documents in HTML, XHTML, SMIL, MathML, etc. If you wish to validate specific content such as [RSS/Atom feeds](#) or [CSS stylesheets](#), [MobileOK content](#), or to [find broken links](#), there are [other validators and tools](#)

# OF GEBRUIK EEN 'LINTER'

Voor Visual Studio Code zijn er diverse plugins die automatisch je HTML, CSS of JS controleren op fouten.

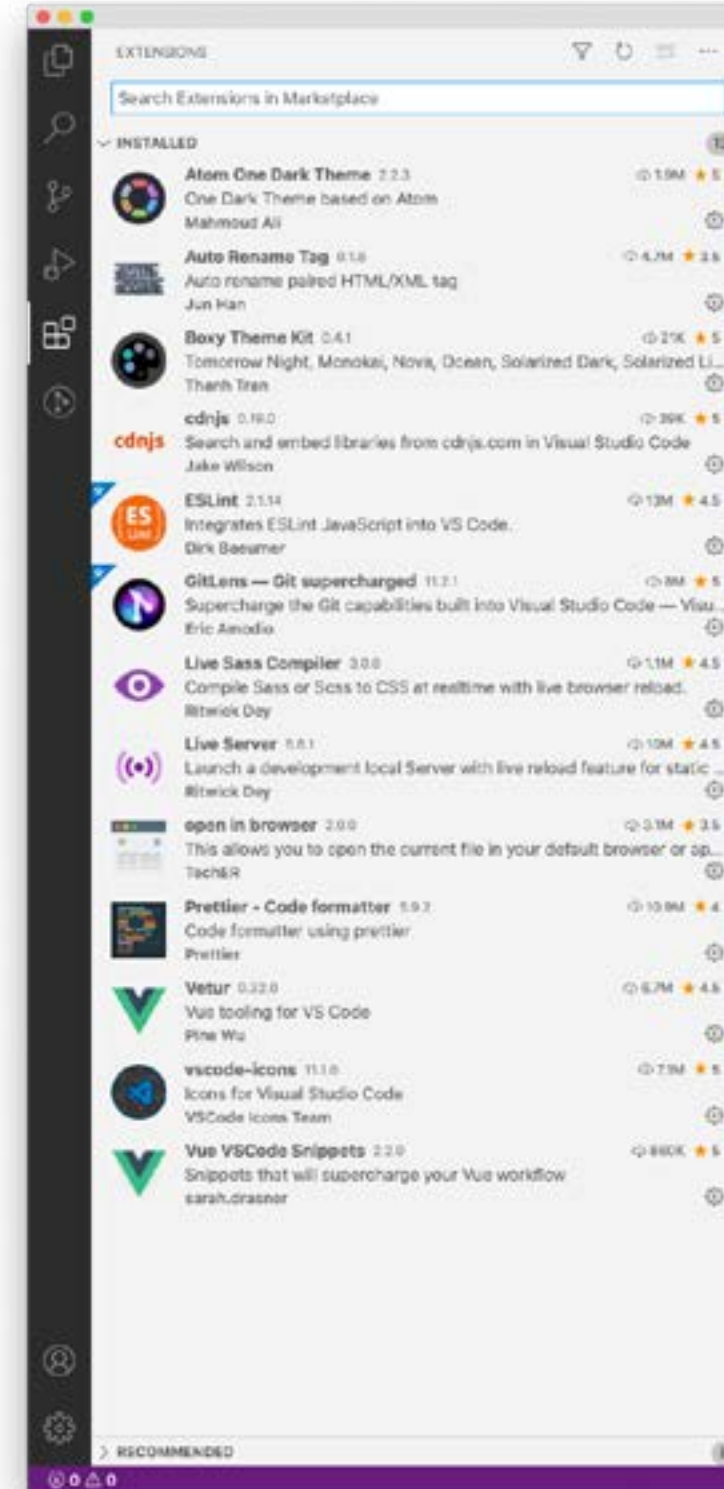
Een van de meest gebruikte is 'HTMLHint'.  
Je kan deze installeren via het plugins tabje.



# ANDERE PLUGINS

Aanraders zijn:

- Open in Browser
- Auto Rename Tag
- Live SASS Compiler
  - Prettier
  - Live Server



# EERSTE PAGINA

---

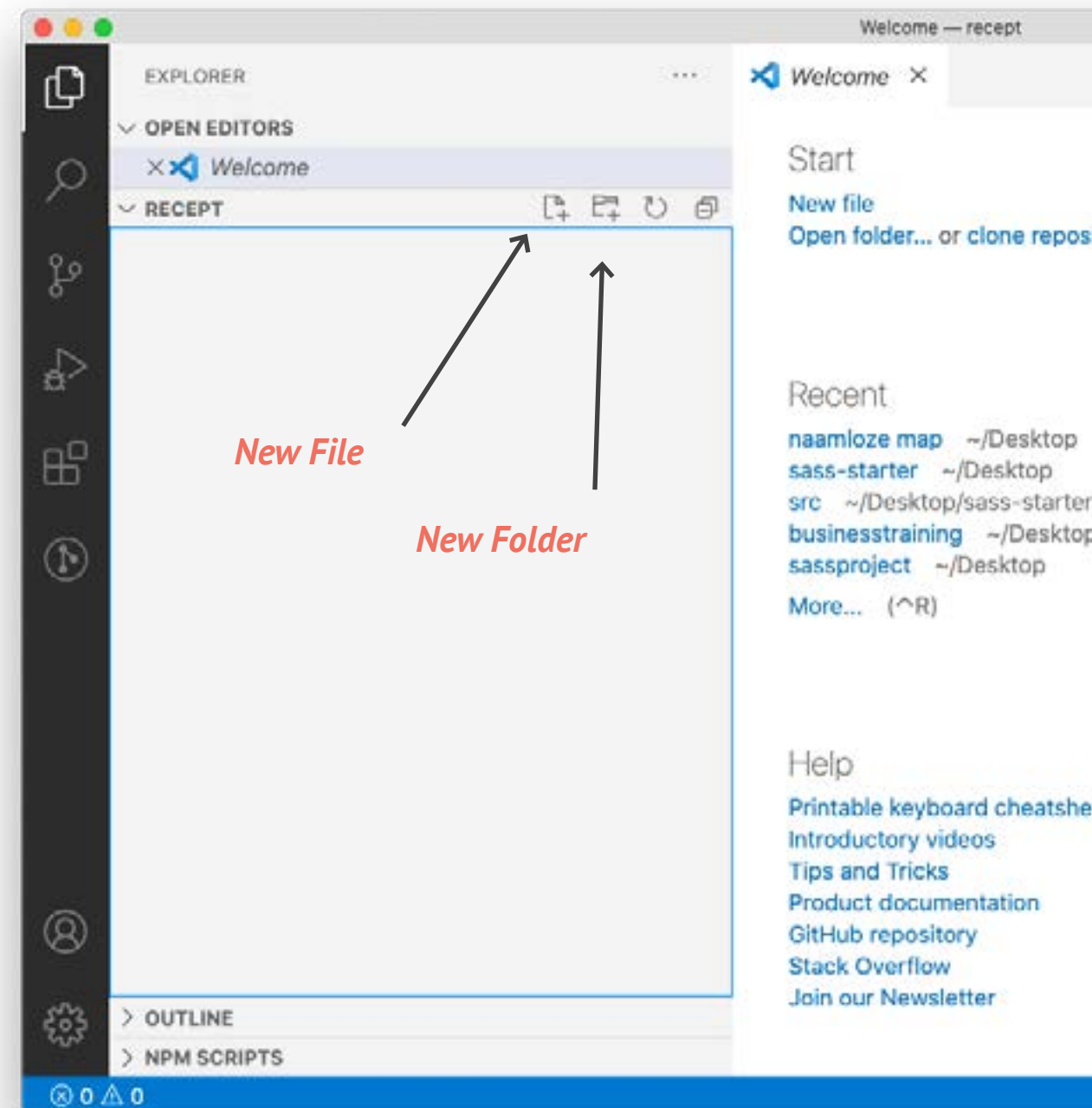
Maak een lege map aan op je bureaublad:

‘Recept’

Sleep de map op het icoontje van VSCODE  
of kies:

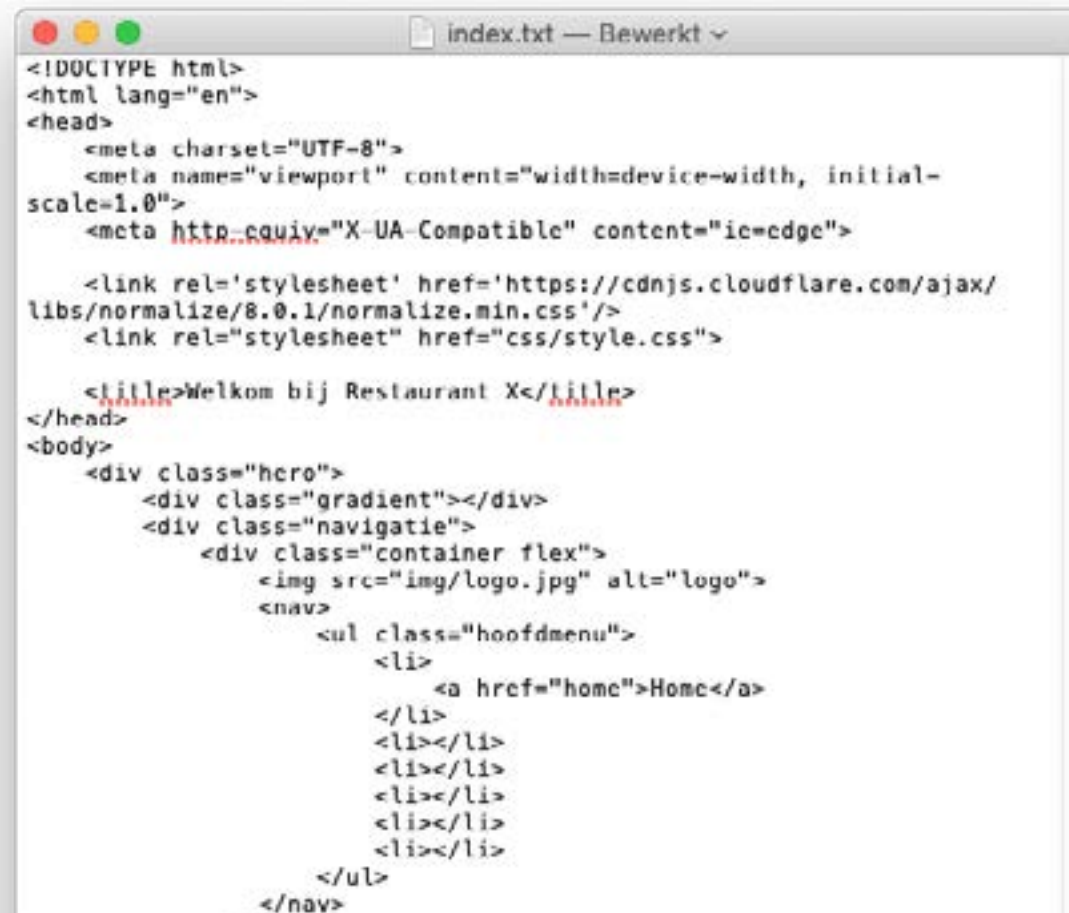
FILE - OPEN - En selecteer de MAP ‘Recept’.

Maak een nieuw bestand aan ‘index.html’.



# WAT IS HTML?

---



```
index.txt — Bewerkt v
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">

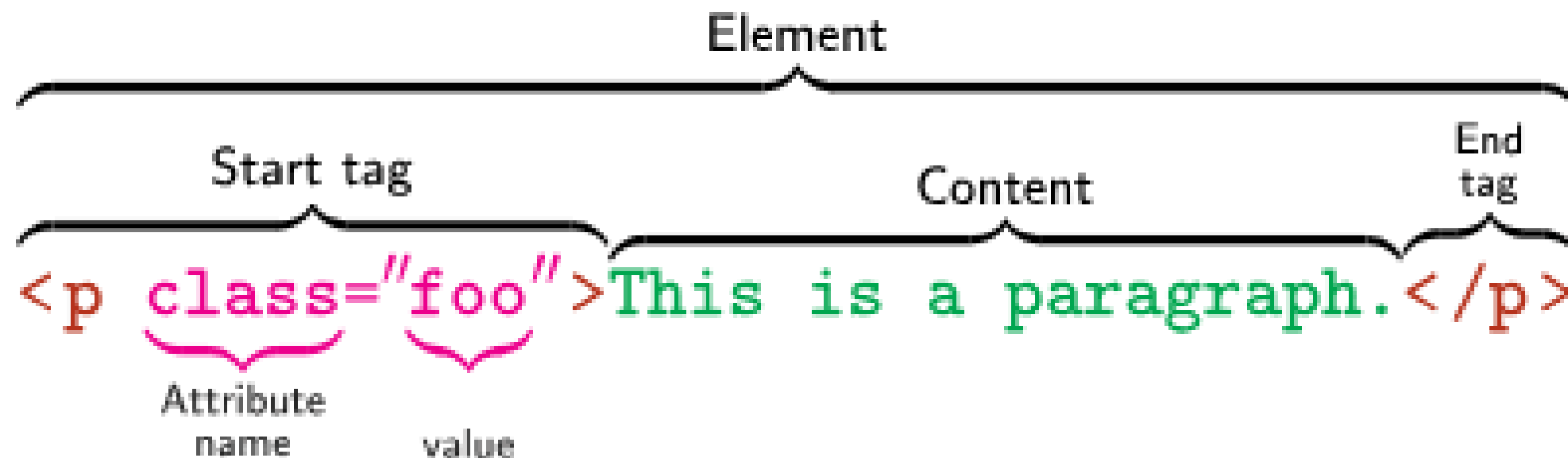
  <link rel='stylesheet' href='https://cdnjs.cloudflare.com/ajax/
libs/normalize/8.0.1/normalize.min.css'/>
  <link rel="stylesheet" href="css/style.css">

  <title>Welkom bij Restaurant X</title>
</head>
<body>
  <div class="hero">
    <div class="gradient"></div>
    <div class="navigatie">
      <div class="container flex">
        
        <nav>
          <ul class="hoofdmenu">
            <li>
              <a href="home">Home</a>
            </li>
            <li></li>
            <li></li>
            <li></li>
            <li></li>
          </ul>
        </nav>
```

Een HTML bestand is een simpel tekstbestand in de ASCII-indeling, opgeslagen met extensie .html of .htm



# ELEMENTEN



Elementen geven tekst een semantische waarde, in andere woorden: het zorgt ervoor dat de computer kan begrijpen wat de betekenis van bepaalde tekst is.

- Een element kan leeg zijn.
- Een element kan uit 1 tag bestaan, of uit een opening- en closingtag
- Een element kan attributen bevatten
- Een element kan andere elementen bevatten

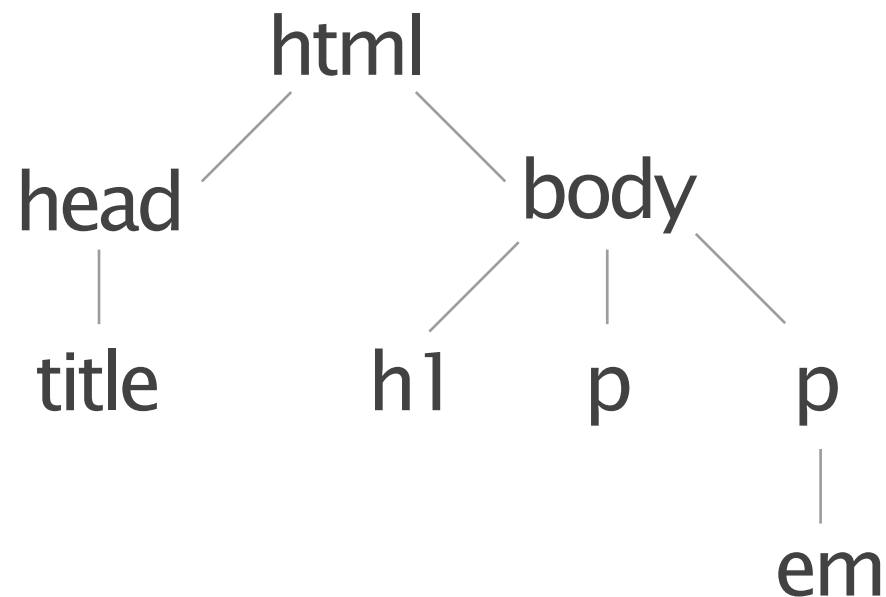


# DOM TREE

---

The document tree  
de DOM (Document Object Model)

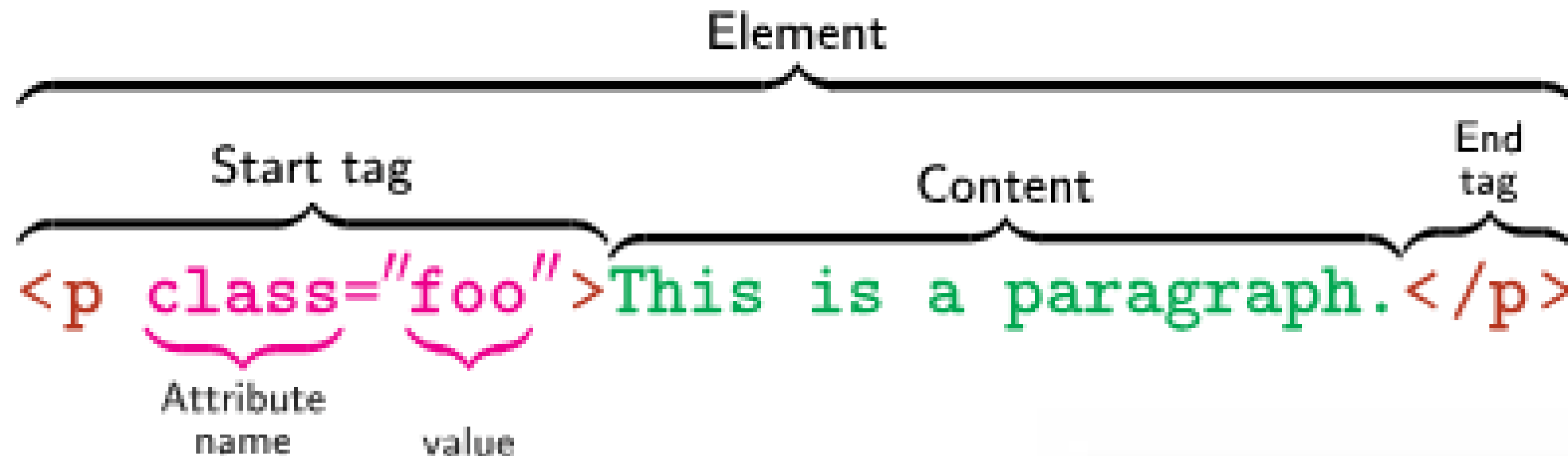
Structuur:



Code:

```
<html>
  <head>
    <title>titel document</title>
  </head>
  <body>
    <h1>hoofdtitel</h1>
    <p>paragraaf</p>
    <p><em>paragraaf</em></p>
  </body>
</html>
```

# ELEMENTEN



Elementen geven tekst een semantische waarde, in andere woorden: het zorgt ervoor dat de computer kan begrijpen wat de betekenis van bepaalde tekst is.

- Een element kan leeg zijn.
- Een element kan uit 1 tag bestaan, of uit een opening- en closingtag
- Een element kan attributen bevatten
- Een element kan andere elementen bevatten

```
<a href="http://www.tweakers.net">Link naar tweakers.net</a>
```

```
<br>  
<hr>  

```

# HEADERS

---

```
<h1>Aan de slag met HTML</h1>
<h2>Koppen markeren</h2>

<!-- Fout: -->
<header>
  <h1>titel</h1>
  <h2>Titel 2</h2> <!-- creert een nieuwe sectie-->
</header>

<!-- Goed: -->
<header>
  <h1>Titel</h1>
  <p>subtitel</p>
</header>

<header>
  <h1>Titel <span>subtitel</span></h1>
</header>
```

Headers gebruik je om titel te maken in een tekst. Van <h1> tot <h6>  
Belangrijk: CSS zorgt voor de opmaak, je kiest enkel op semantiek

# HTML = TEXT MARKEREN

---

```
<body>
  <body>
Bananentaart met chocolade
Verwarm de oven voor op 180 °C.
Vermeng de bloem met wat zout en de suiker. ...
Vet de taartvorm in met boter en bestuif met bloem. ...
Laat op een zacht vuur in het pannetje de chocolade in stukjes smelten met de boter. ...
Klop de eiwitten stijf met de vanillesuiker en breng op smaak met citroensap.
</body>
</body>
```

Platte tekst, zonder betekenis, zonder mogelijkheid voor layout; zelfs TABS werken niet.

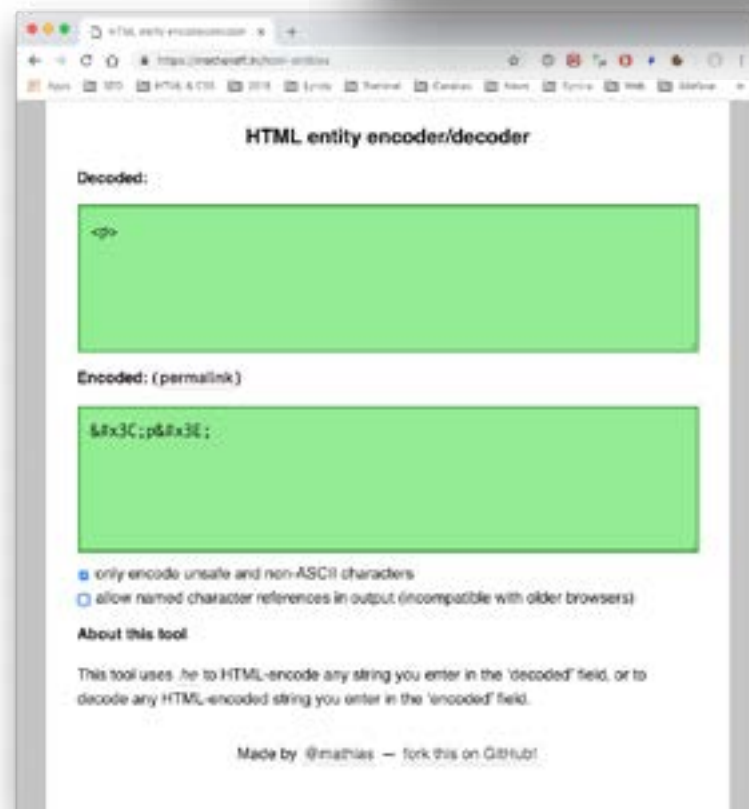
Gebruiks TABS om de hiërarchie te tonen in je code:

```
<body>
  <h1>Kop</h1>
  <main>
    <p>Lorem ipsum dolor
    libero asperiores bea
    <section>
      <h3>tekst</h3>
    </section>
  </main>
</body>
```

# PARAGRAFEN

---


```
<body>
  <body>
    <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Iste debitis praesentium ab inventore odit
tempore sint, ipsa eaque dicta quam porro amet beatae nam laborum ipsum qui cumque autem esse.</p>
    <p>Cumque sapiente deleniti recusandae culpa perferendis ab incidunt sint corporis harum deserunt?
Provident illo, iusto sed eum adipisci commodi blanditiis excepturi deleniti eveniet architecto. Voluptatem
tenetur eligendi repellendus incidunt nisi?</p>
    <p>Hic fuga molestias ratione eius voluptatum perferendis inventore consequatur deleniti quod debitis
dolor id quae soluta laboriosam aspernatur, obcaecati, totam error optio! Amet optio id numquam, libero
minima sequi eius.</p>
  </body>
</body>
```



- `<p>` gebruik je om een nieuwe alinea te beginnen.
- Het einde van een alinea eindigt altijd met een closing-tag: `</p>`
- In de browser default stylesheet zit standaard wat 'margin' tussen de paragrafen.
- Wil je speciale tekens gebruiken dan zal je deze moeten 'escapen', bijvoorbeeld als je een `<` wil gebruiken.

# GIMME A BREAK

---



```
<p>Morgen zal ik dit in orde brengen: <br>
hier volgt een lijstje...</p>
<p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Nam facere nesciunt ut? Fugiat provident
harum dicta libero iste, unde ex, labore eius laborum assumenda impedit alias saepe! Sunt, odio ducimus?</p>
```



In een stuk tekst kan je forceren dat tekst op een nieuwe regel begint.

Het is een 'leeg' element: het heeft geen closing tag.

NIET gebruiken om 'meer witte regels' toe te voegen; dat doe je met CSS.



# NU ZELF

---

Google naar 'Bananentaart Jeroen Meus'

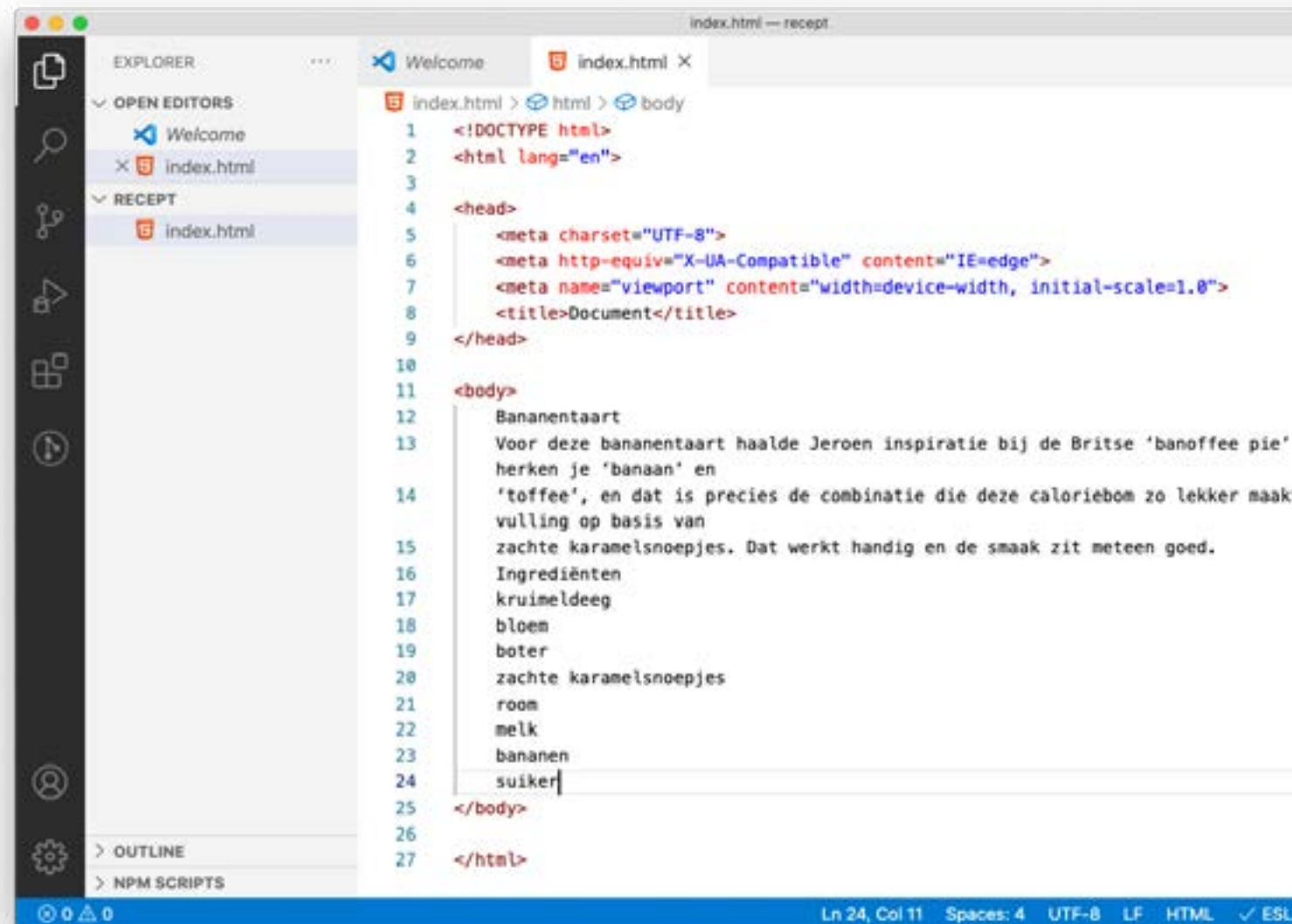
Maak een geldige HTML pagina met:

<h1>

<p>

<ul> of <ol>

en test de pagina in de browser.



```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta http-equiv="X-UA-Compatible" content="IE=edge">
7   <meta name="viewport" content="width=device-width, initial-scale=1.0">
8   <title>Document</title>
9 </head>
10
11 <body>
12   Bananentaart
13   Voor deze bananentaart haalde Jeroen inspiratie bij de Britse 'banoffee pie'
14   herken je 'banaan' en
15   'toffee', en dat is precies de combinatie die deze caloriebom zo lekker maakt
16   vulling op basis van
17   zachte karamelsnoepjes. Dat werkt handig en de smaak zit meteen goed.
18   Ingrediënten
19   kruimeldeeg
20   bloem
21   boter
22   zachte karamelsnoepjes
23   room
24   melk
25   bananen
26   suiker
27 </body>
</html>
```

Bananentaart Voor deze bananentaart haalde Jeroen inspiratie bij de Britse 'banoffee pie'. In dat woord herken je 'banaan' en 'toffee', en dat is precies de combinatie die deze caloriebom zo lekker maakt. Hij maakt de vulling op basis van zachte karamelsnoepjes. Dat werkt handig en de smaak zit meteen goed. Ingrediënten kruimeldeeg bloem boter zachte karamelsnoepjes room melk bananen suiker



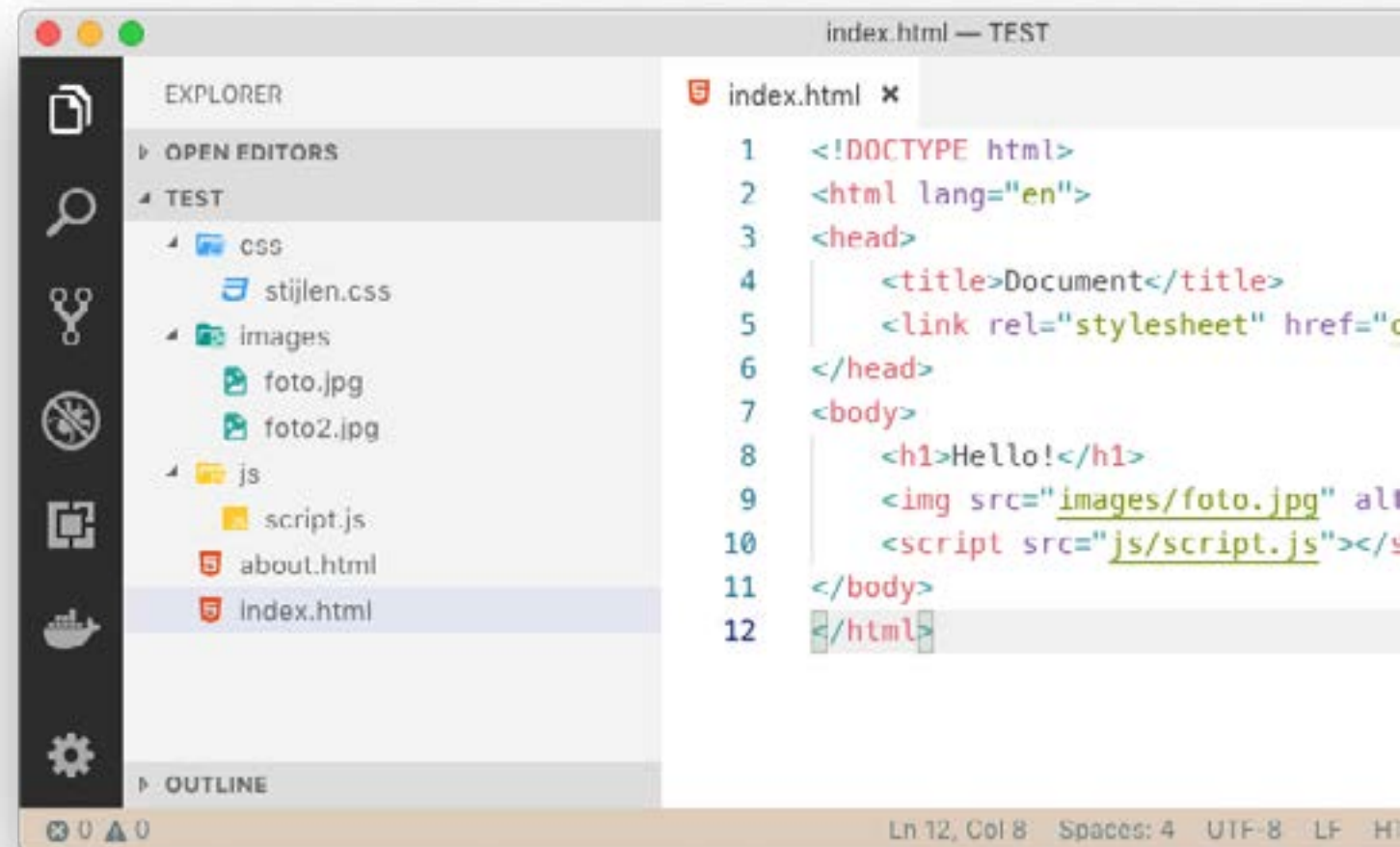
# KOPPELEN VAN EEN AFBEELDING

```
<body>
  <!-- Relative -->
  
  
  

  <!-- Absolute -->
  
```

Als je een afbeelding, video, of geluidsfragment in een HTML pagina plaatst, dan zet je NIET het bestand zelf in de HTML, maar een verwijzing.

Het bestand kan op dezelfde server staan, maar dat hoeft niet.



Koppel nu zelf de afbeelding in het document. Met de 'alt'-attribute geef je de foto een omschrijving.

# RESULTAAT

---



# CSS

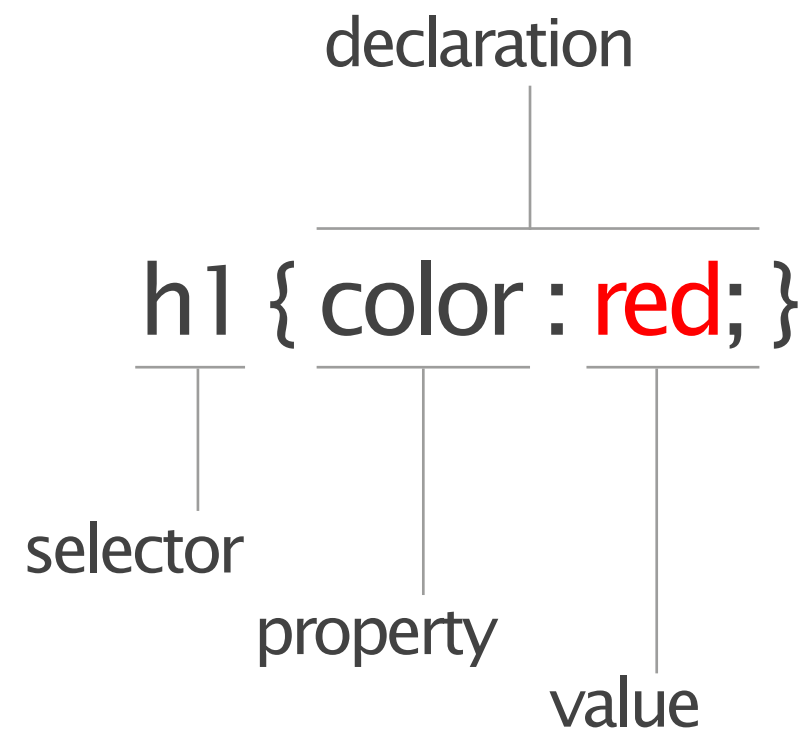
*de basis*

---

*waarom nodig?*

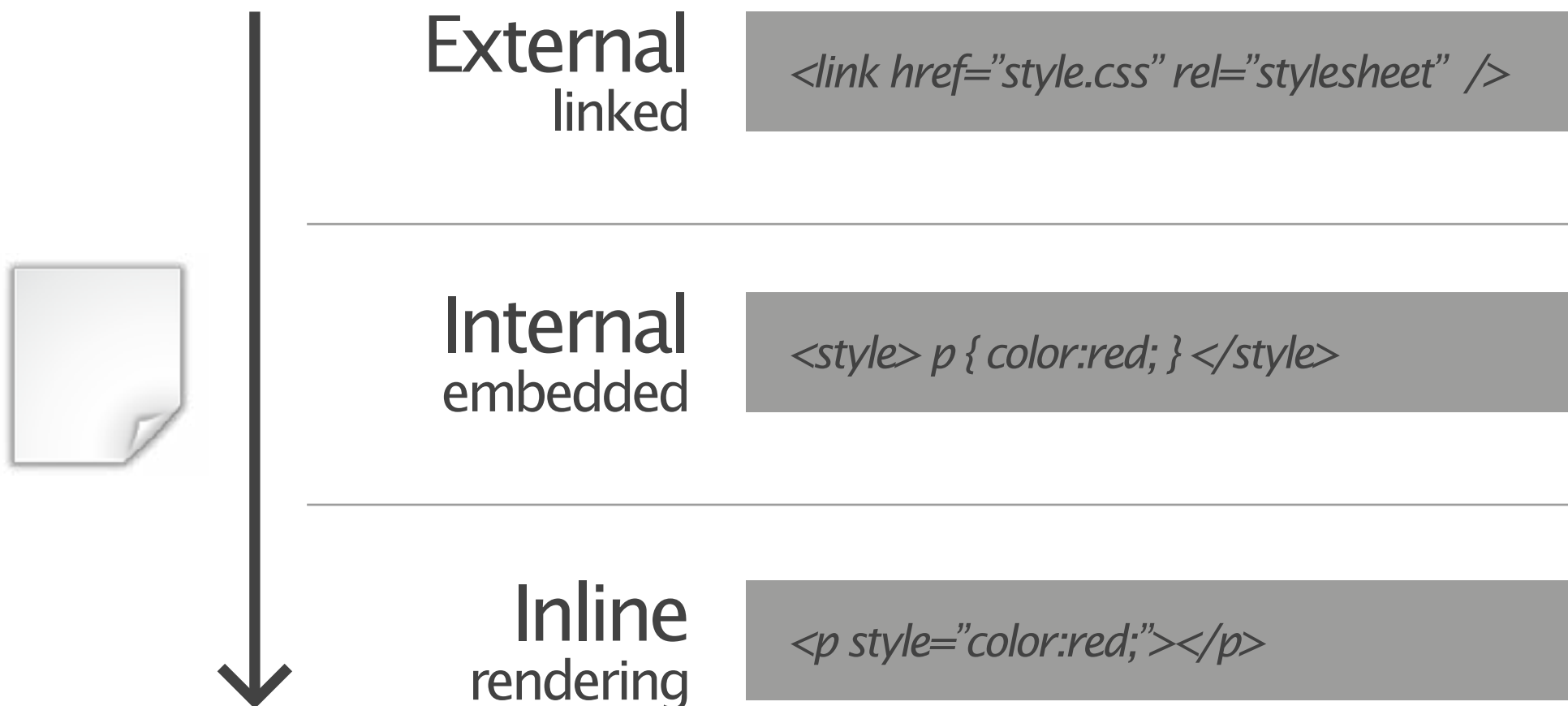
# De anatomie van een css regel

## selectors en declarations



# De 3 locaties voor CSS

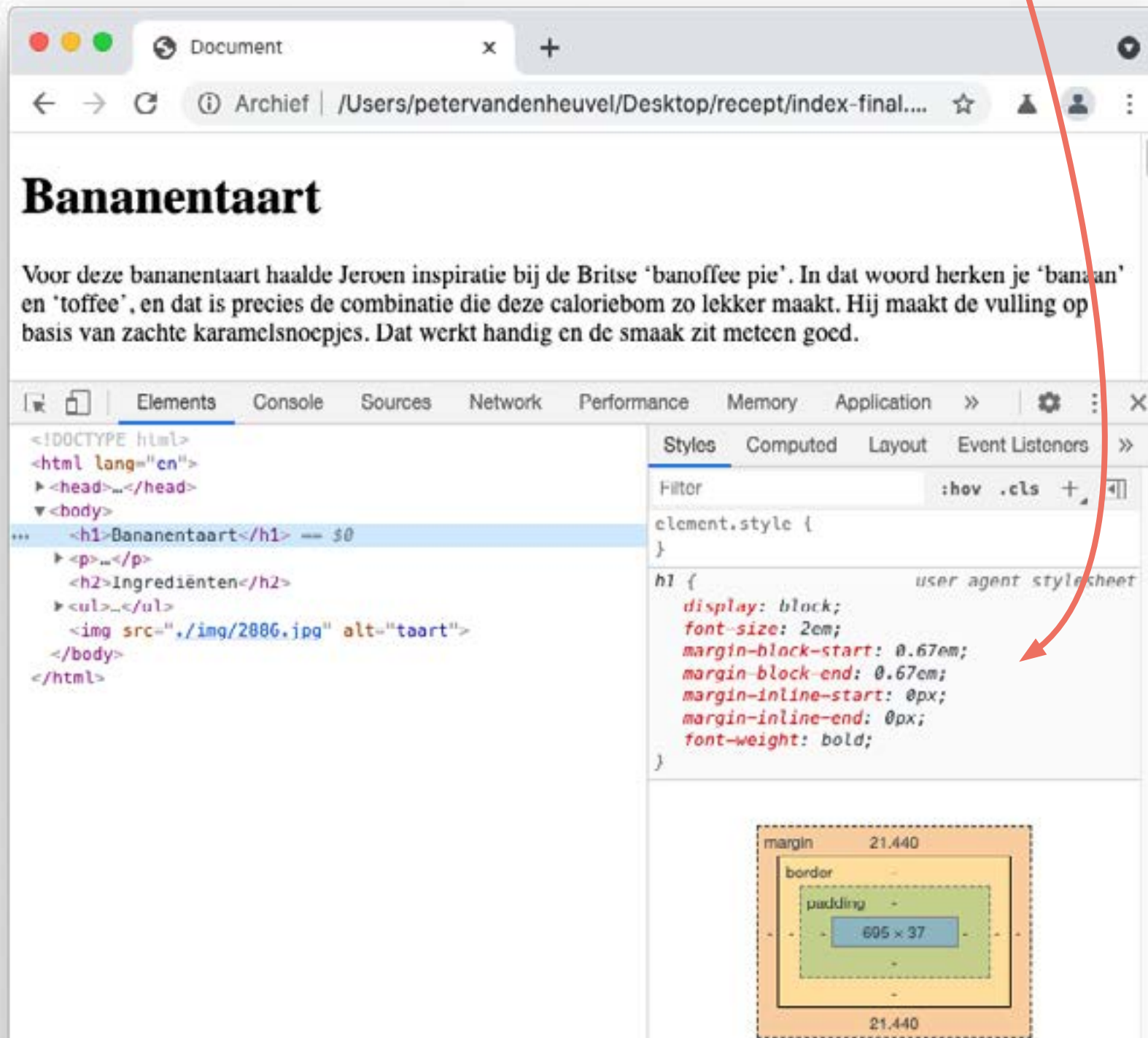
## a cascade of style sheets



waarom is er al styling zonder css?

# CSS reset technieken

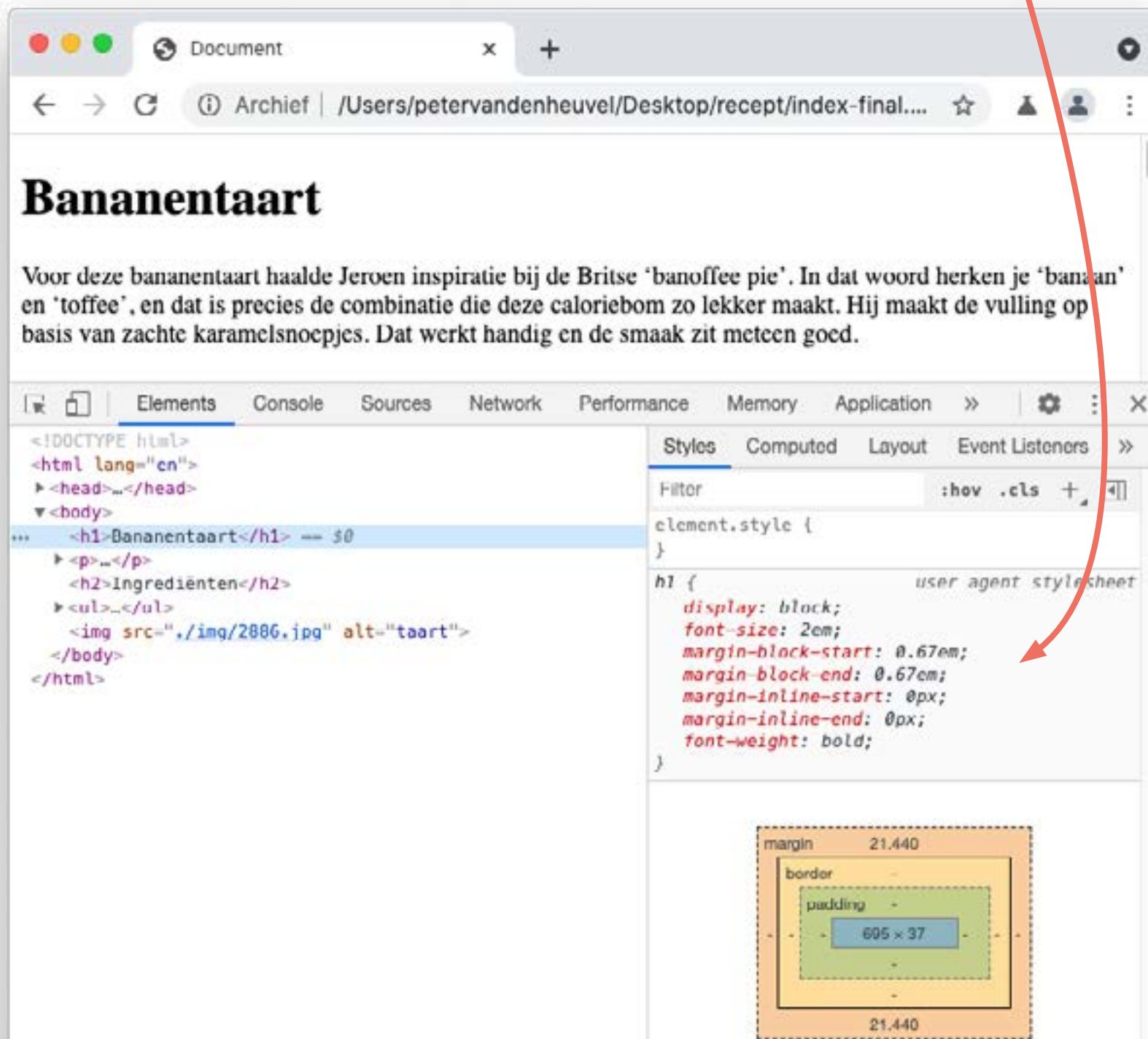
## de vierde locatie voor CSS is 'default'



waarom is er al styling zonder css?

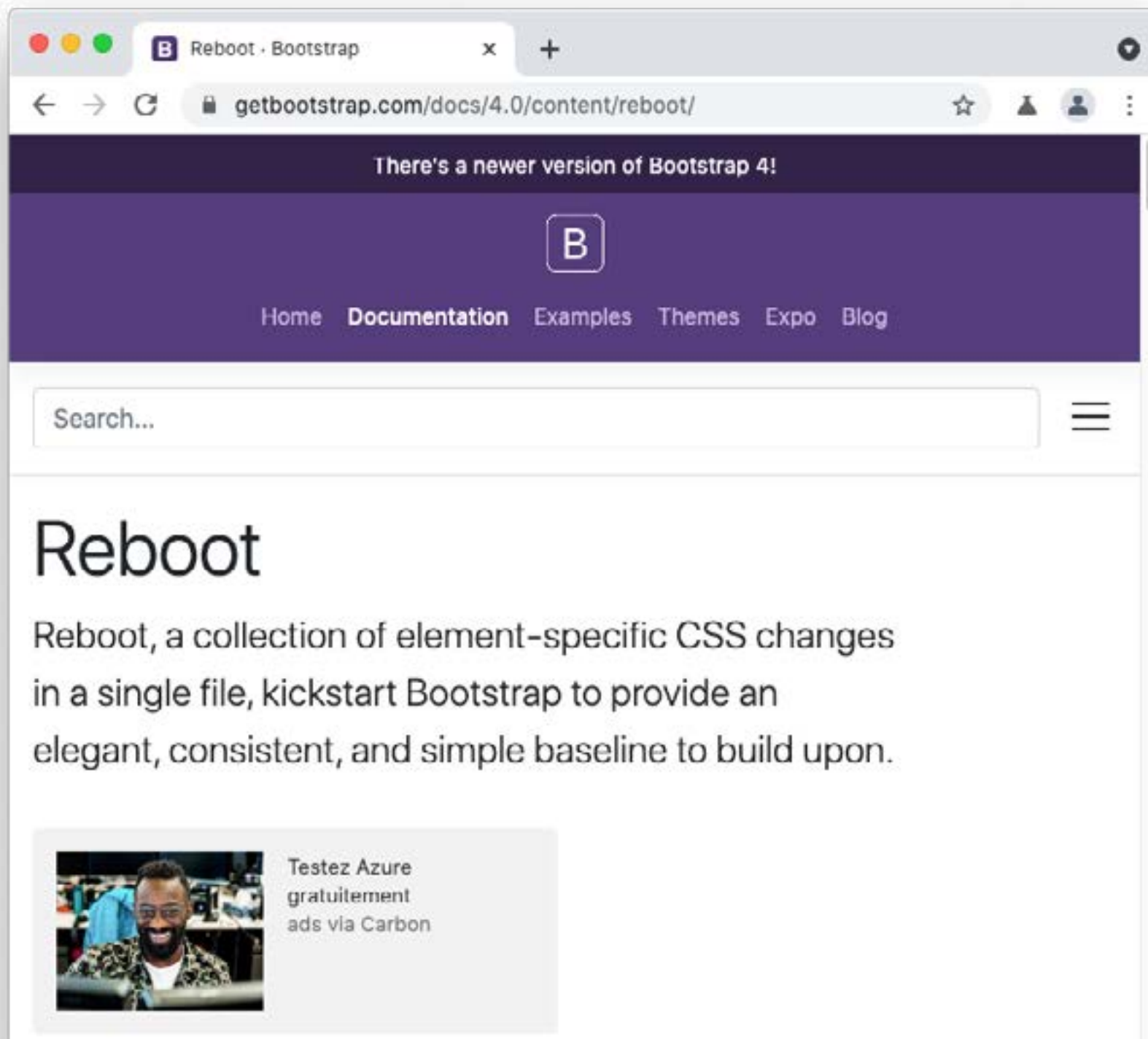
# CSS reset technieken

## de vierde locatie voor CSS is 'default'





# Alles gelijk zetten: Reboot.css (of normalize, of...)



Schrijf nu zelf 4 CSS regels in de recepten pagina.

## De 4 selectors en hun afgeleiden

### HTML elementen <tags>

`body { font-size: 12px; }`

Alle tekst op de pagina

`p { color: red; }`

Elke paragraaf

`h1 { color: black; }`

Elke hoofding 1

`<body>`

`<h1>hoofdtitel</h1>`

`<p>paragraaf</p>`

`</body>`

“

hoofdtitel

paragraaf

# De 4 selectors en hun afgeleiden

## ID's (elke id max 1 x per pagina)

```
#content { color: red; }
```

Elke Id is uniek

```
<body>
```

```
<div id="content">
```

*Een identifier is een selector om 1  
specifiek element te benaderen in de  
html-code.*

```
</div>
```

```
</body>
```

“

**Een identifier is een selector  
om 1 specifiek element te be-  
naderen in de html-code.**

”

# De 4 selectors en hun afgeleiden

## Klasse (class)

```
.first { color: red; }  
.description { font-weight: bold; }  
  
<body>  
  <h2 class="first">eerste titel</h2>  
  <p class="first description">eerste tekst</p>  
  <p>geen klasse</p>  
</body>
```

Elke klasse krijgt hier  
een eigen naam mee

**eerste titel**  
**eerste tekst**  
geen klasse

# De 4 selectors en hun afgeleiden

Voeg een link to naar 'dagelijkse kost'  
en style de link kleur. (en de hover)

## Pseudoklasse

```
a:link { color: black; }  
a:visited { color: blue; }  
a:hover { color: red; }  
a:active { color: green; }  
a:focus { color: red; }
```

```
<body>
```

```
  <a href="#">www.carabas.be</a>
```

```
</body>
```

De 5 statussen van een link

“  
www.carabas.be  
”

“  
www.carabas.be  
”



# Decendent selectors

de span in de link in de paragraaf

Alle combinaties van selectors zijn mogelijk

```
p span { color: red; }  
p span .italic { font-style: italic; }  
  
<body>  
  <h1>the <span>color</span></h1>  
  <p>the <span>color</span></p>  
  <p>the <span class="italic">color</span></p>  
</body>
```

Hoe langer de combinatie, hoe specifieker

“  
the color  
the **color**  
the *color*  
”

# Inheritance - overerving

blauwe div tags zijn erfelijk

Test dit uit! Op de body, met  
kleur of font family

## Stap 1

maak alle body tekst  
blauw

```
<body>
  <p>een paragraaf</p>
  <div>tekst in een div tag</div>
</body>
```

## Stap 2

maak alle tekst in een  
paragraaf rood

```
body { color: blue; }
```

“  
een paragraaf  
tekst in een div tag  
”



```
body { color: blue; }
```

```
p { color: red; }
```

“  
een paragraaf  
tekst in een div tag  
”



# Specificity

## the biggest number wins



	Inline	ID	Class	Element	Specificity
<code>&lt;p style="..."&gt;</code>	1	0	0	0	1000
<code>h1, h2</code>	0	0	0	1	1
<code>h1.first</code>	0	0	1	1	11
<code>#main</code>	0	1	0	0	100
<code>#main h2</code>	0	1	0	1	101
<code>#main #sidebar p</code>	0	2	0	1	201
<code>h1 a</code>	0	0	0	2	2
<code>.author</code>	0	0	1	0	10

!important  
krijgt altijd voorrang



Stap 1  
zonder important

```
<body>  
  <div><p>tekst</p></div>  
</body>
```

Stap 2  
met important

```
p { color: green; }  
body div p { color: red; }
```

1  
3

```
p { color: green !important; }  
body div p { color: red; }
```



“  
tekst  
”

“  
tekst  
”

# DE BASICS: NORMAL FLOW LEARN THE RULES

*so you can break them*



# NORMAL FLOW<sup>(of: static)</sup>

*alle elementen op elkaar*

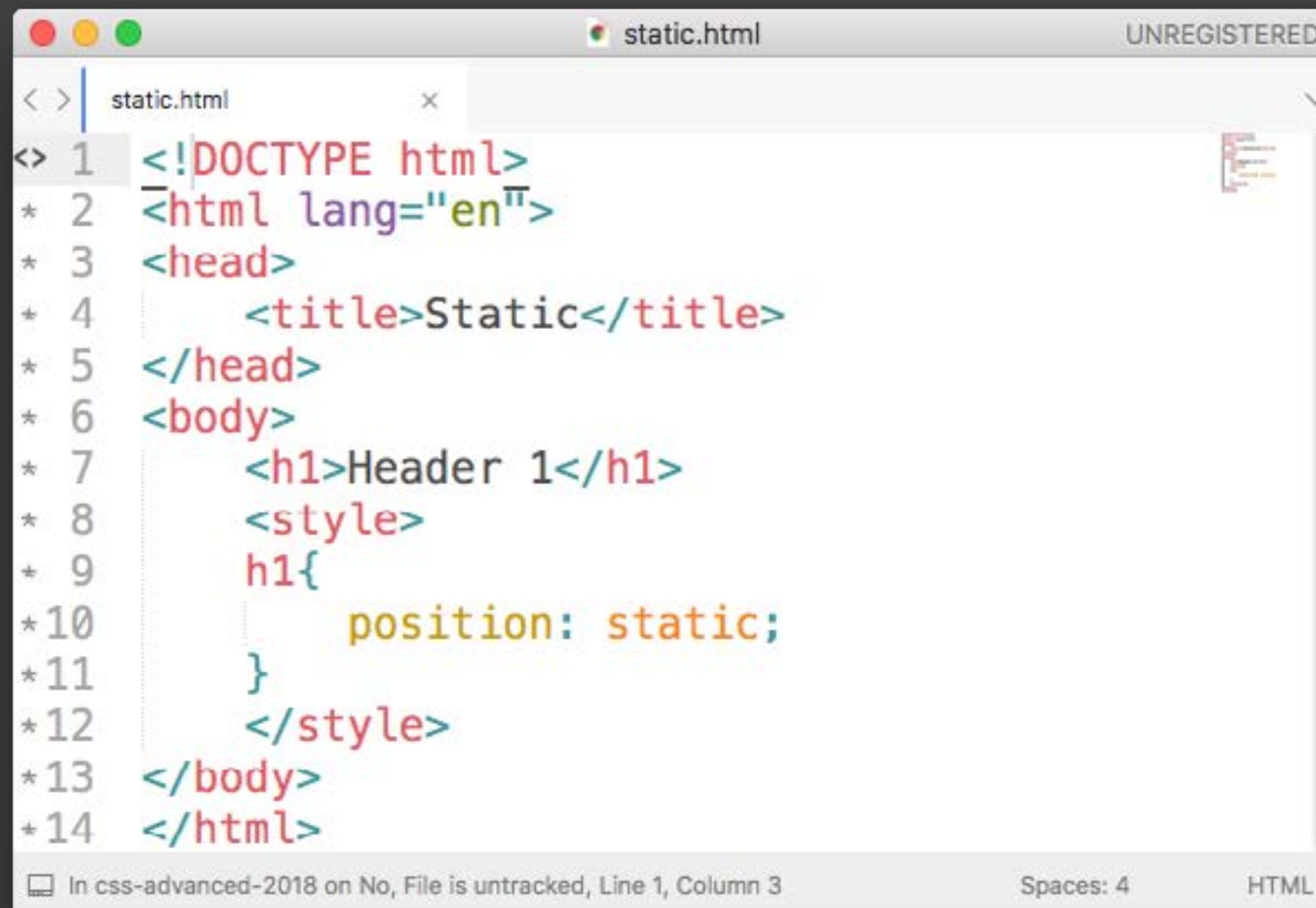
---

- Volgt de *volgorde* van schrijven in je HTML
- Het *eerste element* komt bovenaan *links* in de browser
- Het volgende element komt *ernaast* of *eronder*

<https://codepen.io/petervandenheuvel/pen/xyYXww?editors=1100>

# GEEN POSITIONING

*normal flow of position: static*



```
<> static.html x
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <title>Static</title>
5 </head>
6 <body>
7     <h1>Header 1</h1>
8     <style>
9     h1{
10         position: static;
11     }
12 </style>
13 </body>
14 </html>
```

In css-advanced-2018 on No, File is untracked, Line 1, Column 3      Spaces: 4      HTML



# NORMAL FLOW

*‘verschuiven’ binnen normal flow:*

```
div.element-in-normal-flow{  
  margin-left: 20px; /* Move left outside box */  
  padding-left: 40px; /* Move left inside box */  
  
  display: block; /* Change to block-level */  
  display: inline; /* Change to inline-level */  
  display: inline-block; /* Change to hybrid */  
}
```

Let op: - Gebruik kleine waarden om elementen te verplaatsen:  
(een margin-top: 500px of margin-left: -250px is te vermijden)

<https://codepen.io/petervandenheuvel/pen/xyYXww>

# BLOCK VS INLINE

*dozen, overal dozen!*

**<H1>**

**<H2>**

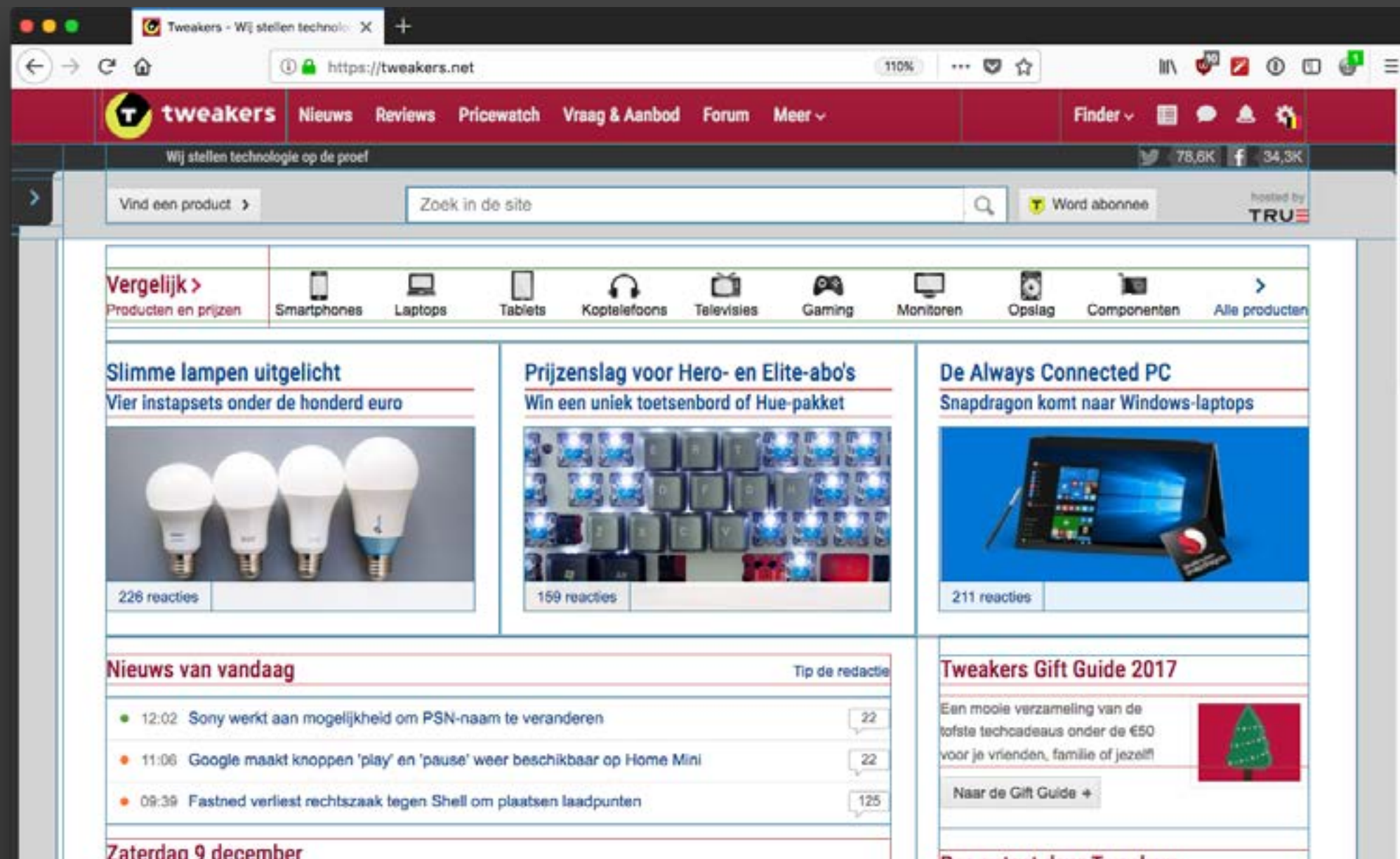
**<SPAN>**

**<EM>**

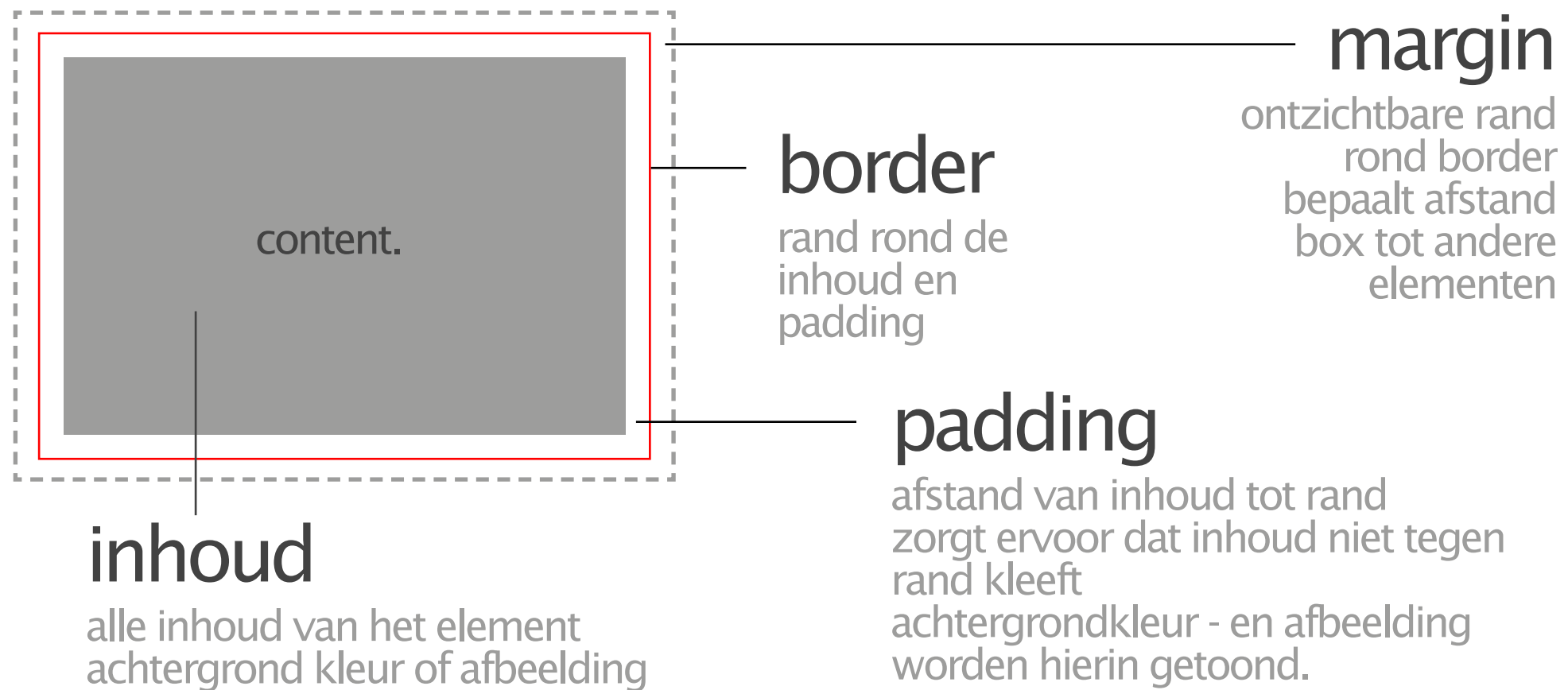


# MAAK DOZEN ZICHTBAAR

*gebruik pesticide.io of web-dev toolbar*



# THE BOX MODEL



# THE BOX MODEL

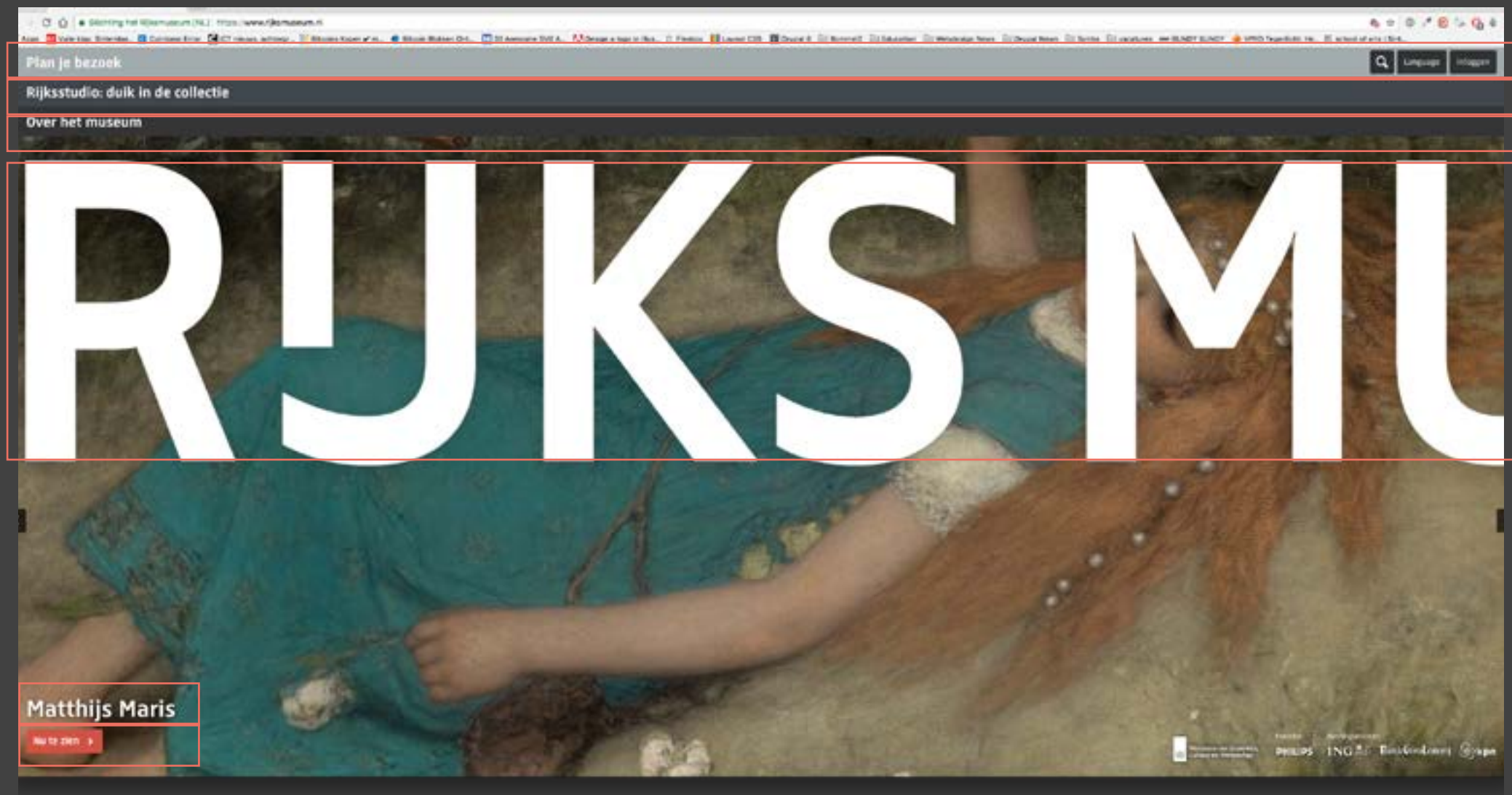


```
.doos{  
    box-sizing: content-box;  
}  
  
.doos{  
    box-sizing: border-box;  
}
```



# SOMS MAG HET COMPLEXER

*Alle complexe css layouts zijn ook 'dozen op elkaar'*



# DE BASICS:

*Hoe verschuif je de elementen uit hun natuurlijke flow?*

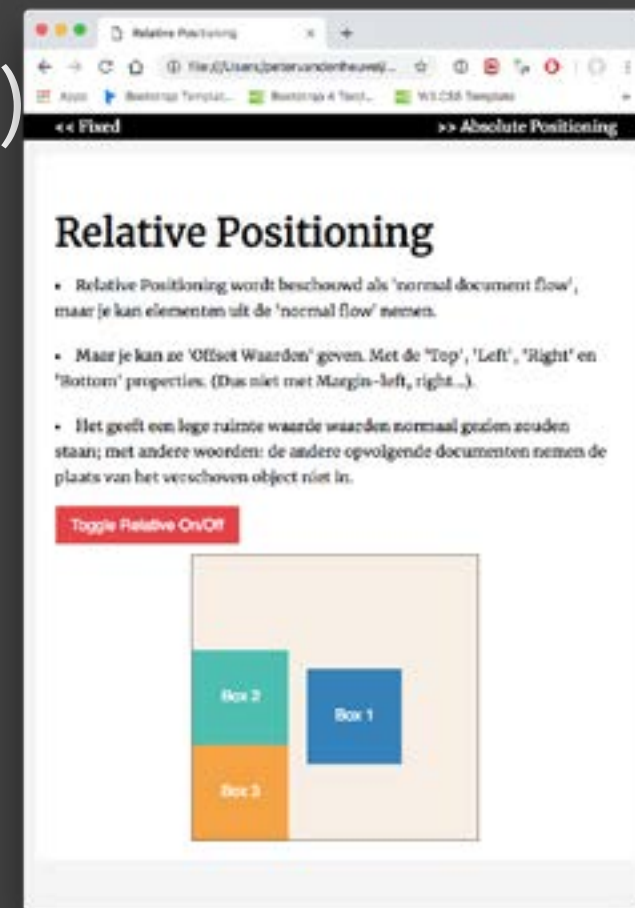
- ① RELATIEF POSITIONEREN
- ② ABSOLUUT POSITIONEREN
- ③ FIXED POSITIONEREN
- ④ POSITION 'STICKY'
- ⑤ FLOATEN

# 1) RELATIVE POSITIONING

*h1{ position: relative;}*

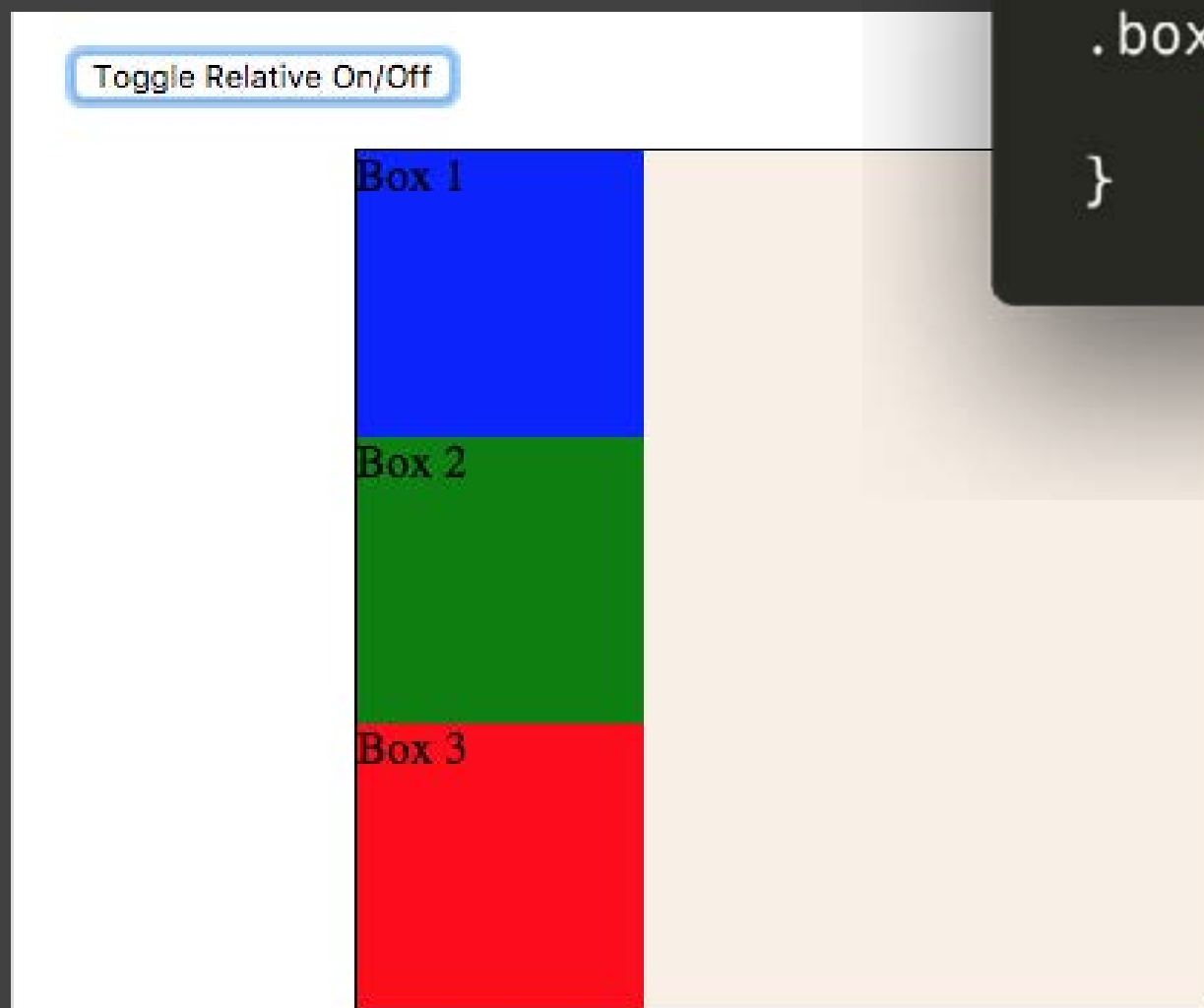
- 1) Het **geselecteerde** element gedraagt zich als een **static** (niet-positioneerd) element.
- 2) Totdat je het een **positie** geeft. (Offset waarden)
- 3) Ofwel: het element blijft in de **normal flow**.

[HTTPS://CODEPEN.IO/PETERVANDENHEUVEL/PEN/GBVEYB](https://codepen.io/petervandenheuvel/pen/GBVEYB)



# VOORBEELD 1

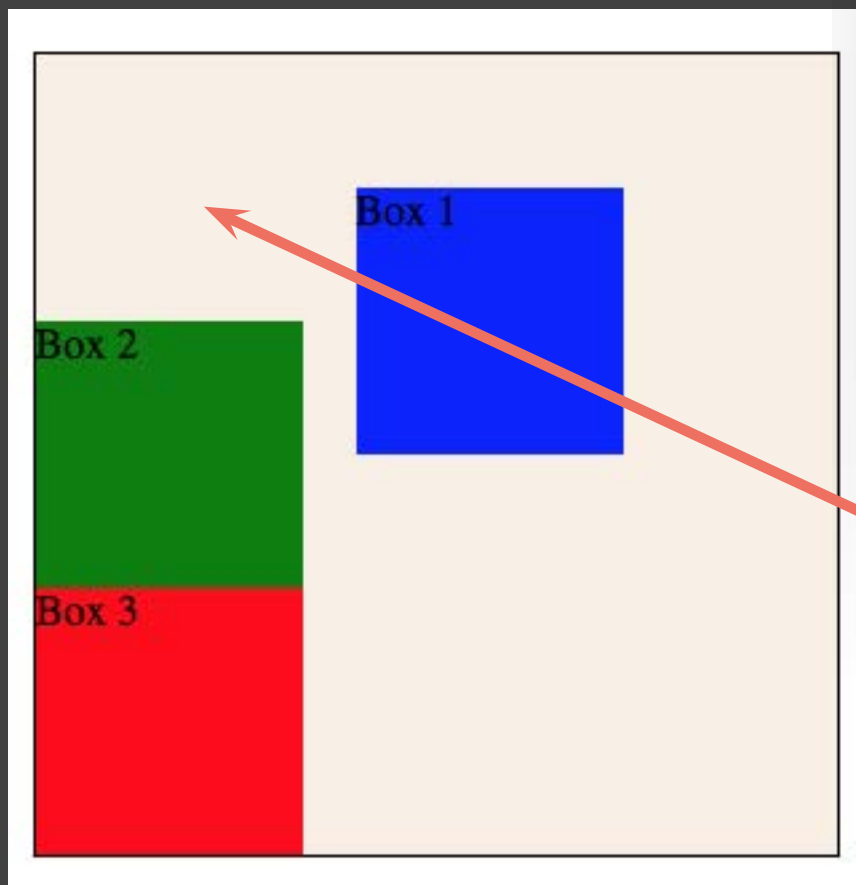
*er lijkt niets veranderd:  
maar doos 1 heeft nu 'position:relative'.*



```
.box1 {  
    position: relative;  
}
```

# VOORBEELD 2

*totdat je offset waarden geeft:*



```
.box1 {  
    position: relative;  
    left: 10px;  
    top: 10px;  
}
```

De 'geest' van .box1 blijft staan op de normale plek.

<https://codepen.io/petervandenheuvel/pen/gBveyb>



# 1) RELATIVE POSITIONING

Het **volgende** element wordt gepositioneerd op de plaats waar het **normaal gezien zou voorkomen** in normal flow.

De **normal flow** is exact **hetzelfde** voor de verplaatsing, ookal is het **element visueel verplaatst**.

Gevolg: **JE MOET WETEN WAAR HET ELEMENT IN NORMAL FLOW ZOU STAAN!**  
Extra MARGIN toevoegen maakt dit nog complexer, pas daar mee op.

# OPDRACHT

*maak het voorbeeld na en test de gevolgen van:  
'position: relative'*

- 1) Maak het kader: een **div.container**: width & margin: 0 auto
- 2) Maak de drie kubussen (.box) daarin (block level-elementen..)
- 3) Gebruik een klasse **.box** om hem te verkleinen: 100px op 100px.
- 4) Test het toepassen van **pos: relative** op de eerste box:
  - Wat gebeurt er met .box1 ?
  - Wat gebeurt er als je top en left-values toevoegd?

```
.container{  
  background: lightgrey;  
  width: 50%;  
  margin: 0 auto;  
}  
  
.box{  
  width: 100px;  
  height: 100px;  
}  
  
.one{background:blue;}  
.two{background:green;}  
.three{background:red;}  
  
.one{  
  position: relative;  
  left: 125px;  
  top: 50px;  
}
```

## 2 ABSOLUUT POSITIONEREN

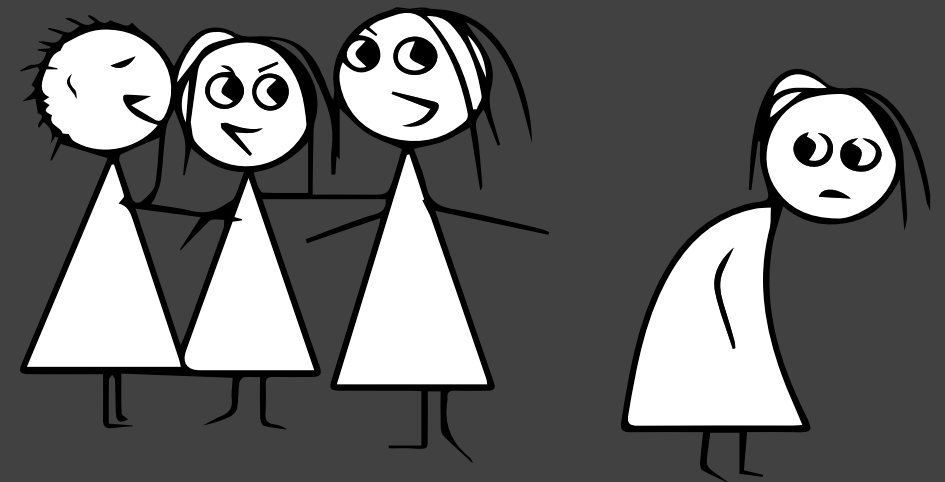
---

# ABSOLUTE POSITIONING

```
.box1{  
  position: absolute;  
}
```

- Het 'x' element wordt **UIT** de 'natuurlijke flow' van de pagina gehaald.
- Het volgende element in de normal flow doet dus net alsof x niet bestaat.

Als je iets **position: absolute** geeft, doen alle andere elementen alsof dit element niet meer bestaat op de pagina: 'uit de flow'-nemen.



# ABSOLUTE POSITIONING

*Opdracht: pas je (vorige) voorbeeld aan:*

[HTTPS://CODEPEN.IO/PETERVANDENHEUVEL/PEN/ZMRJOM](https://codepen.io/petervandenheuvel/pen/zmrjom)

- *Het verplaatst zich ten opzichte van het **eerste (parent) element** dat **positioning** heeft, als dat er niet is neemt hij de **<body>**-tag.*

*(Dit werkt alleen als je element 'left, right, top, bottom'-waarden heeft...)*

*Het **lijkt** dan alsof het element **position: fixed** heeft...*

```
.container{
  background: lightgrey;
  width: 50%;
  margin: 0 auto;
}

.box{
  width: 100px;
  height: 100px;
}

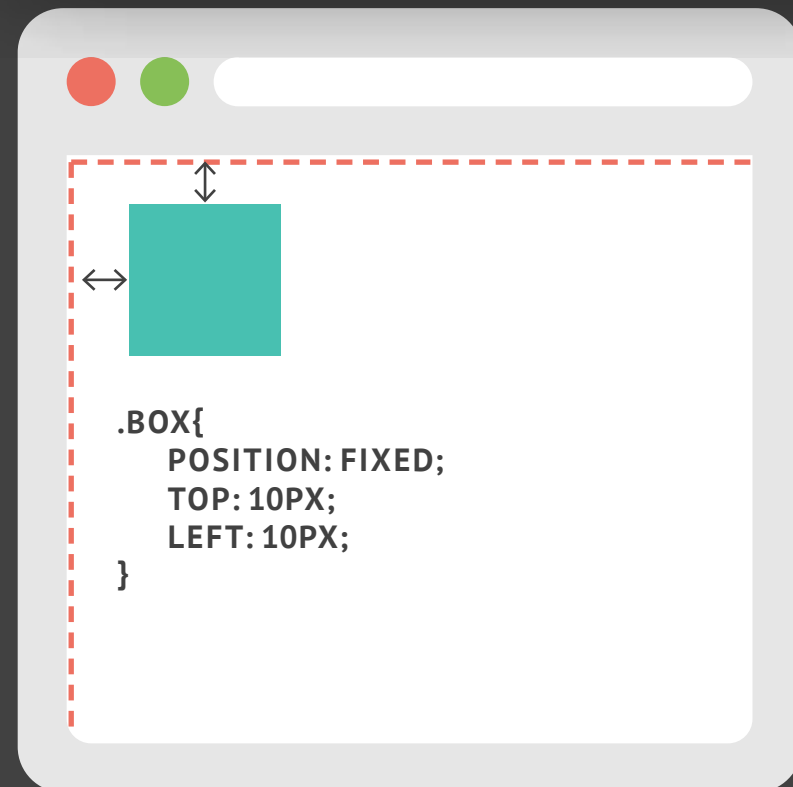
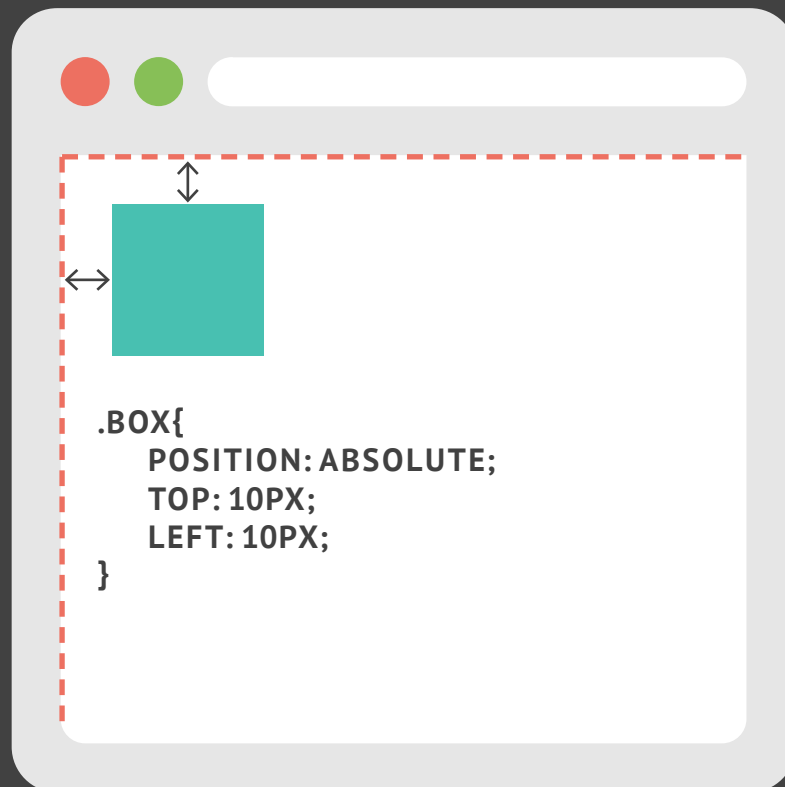
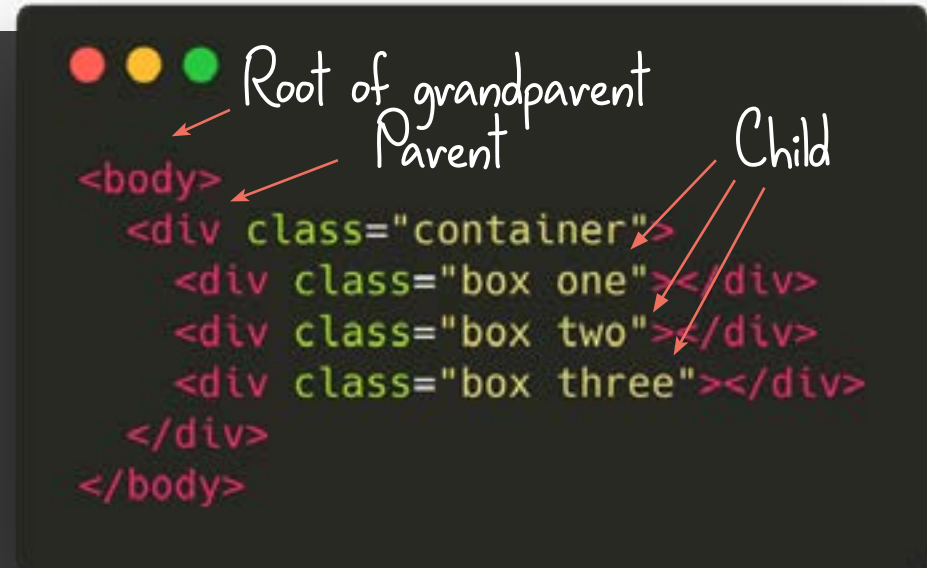
.one{background:blue;}
.two{background:green;}
.three{background:red;}

.one{
  position: absolute;
  left: 125px;
  top: 50px;
}
```

# ABSOLUTE POSITIONING

- Het verplaatst zich ten opzichte van het *eerste (parent) element* dat *positioning* heeft, als dat er niet is neemt hij de `<body>`-tag.

Het lijkt dan alsof het element *position: fixed* heeft...

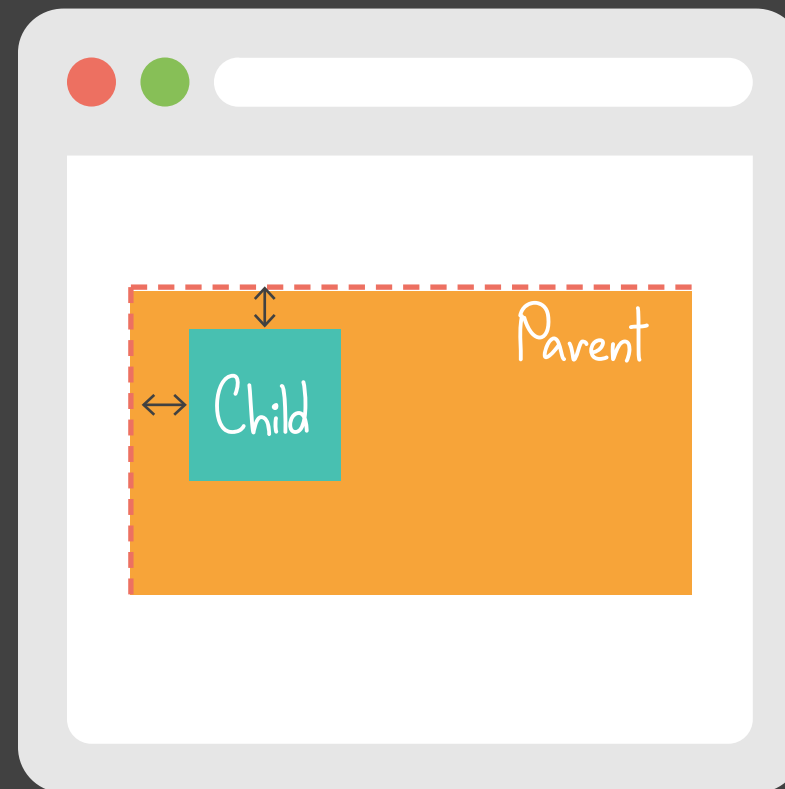


# ABSOLUTE POSITIONING

- Als er wel een *parent* is met positioning (bijv. 'relative');  
dan plaatst het *absoluut* gepositioneerde *kind* zich t.o.v. de parent.

```
Parent
.orange {
  position: relative;
}

Child
.green {
  position: absolute;
  top: 10px;
  left: 10px;
}
```





# ABSOLUTE POSITIONING

*De breedte van het element*

*Vanaf het moment dat je een element '**Position: absolute**' geeft, gaat deze naar de **minimale breedte** van de zijn **inhoud**.*

*Is er geen inhoud, dan wordt deze '0'.*

*Je kan dat eens testen in het vorige voorbeeld door de 'width' weg te halen, en de inhoud (text) te verwijderen.*

# PRAKTISCH VOORBEELD

*absolute positioning*



```
.BLAUW-BUTTON {  
  POSITION: ABSOLUTE;  
  BOTTOM: 10PX;  
  RIGHT: 10PX;  
}
```

```
.KADER {  
  POSITION: RELATIVE;  
}
```

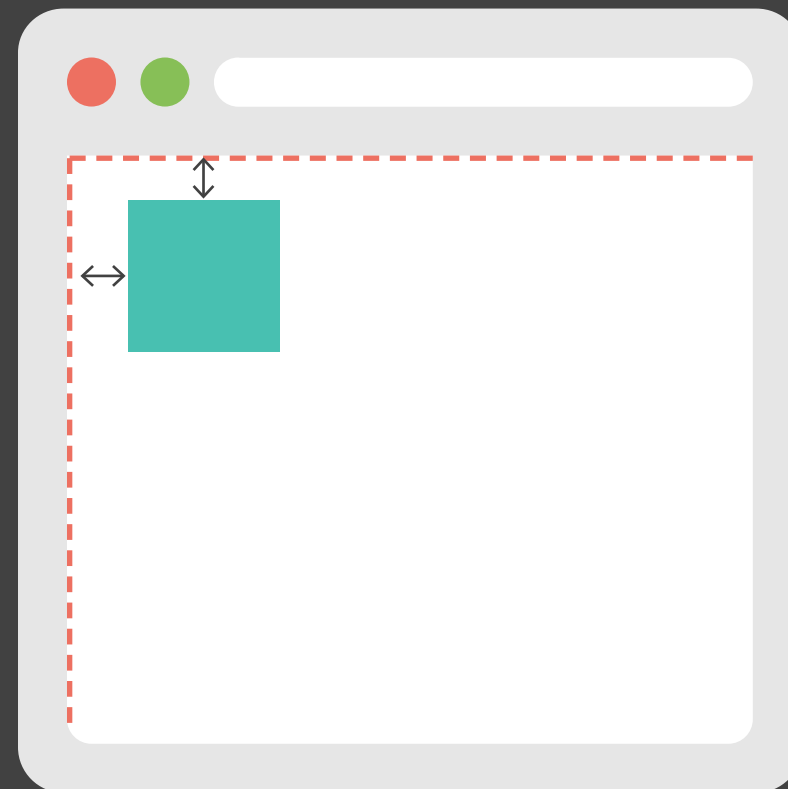
# 3 FIXED POSITIONEREN

---

# FIXED POSITIONING

*het element wordt geplaatst t.o.v. de <body>*

```
.GREEN {  
  POSITION: FIXED;  
  TOP: 10PX;  
  LEFT: 10PX;  
}
```



<https://codepen.io/petervandenheuvel/pen/zVpXjp>

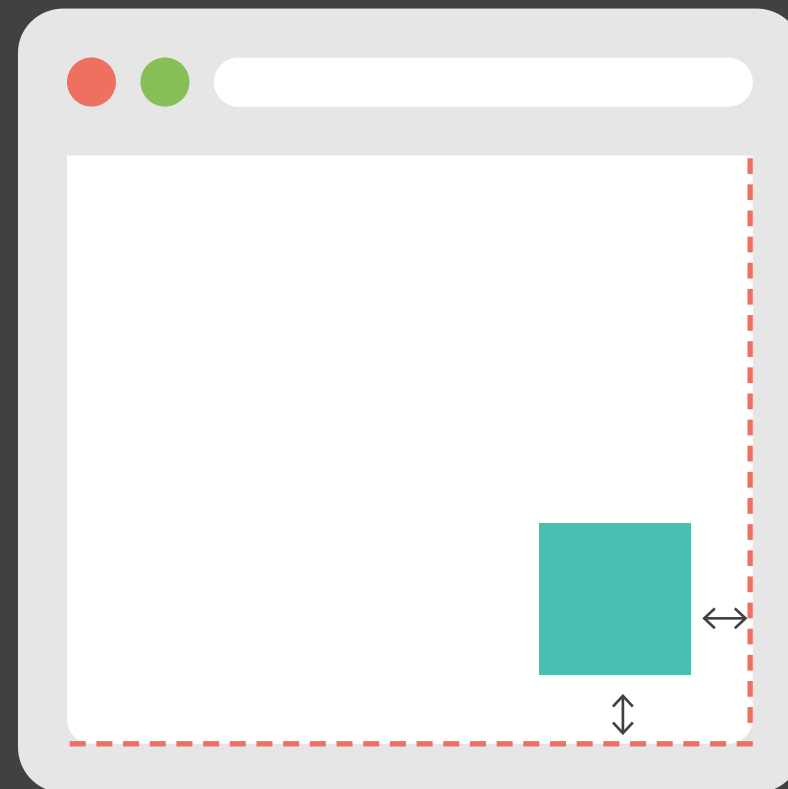
# FIXED POSITIONING

*het element wordt geplaatst t.o.v. de <body>*

```
.GREEN {  
  POSITION: FIXED;  
  BOTTOM: 10PX;  
  RIGHT: 10PX;  
}
```

HET ELEMENT BLIJFT ALTIJD  
ZICHTBAAR IN HET SCHERM,  
OOK ALS JE SCROLLED.

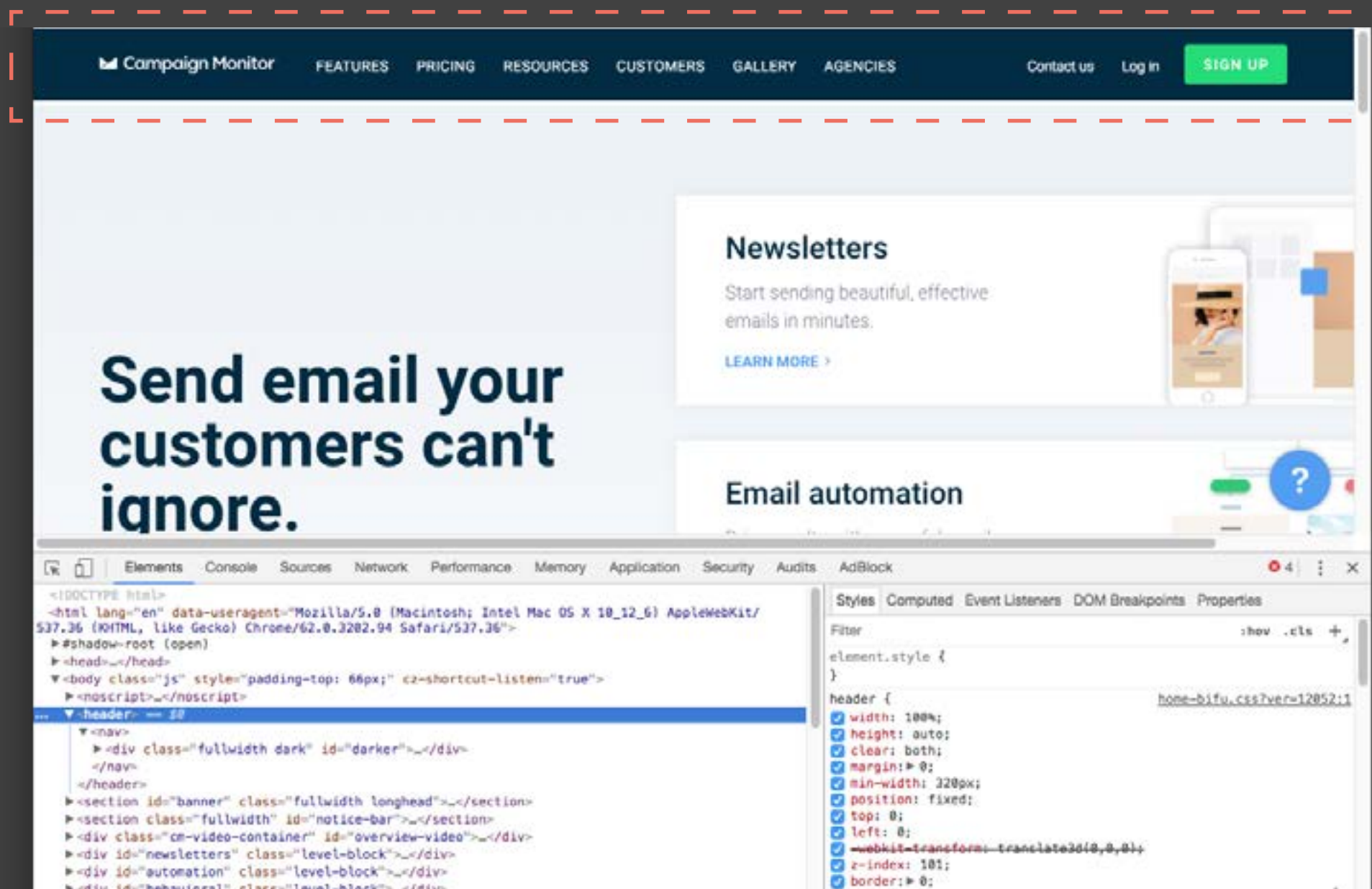
IDEAAL VOOR IETS DAT ALTIJD  
ZICHTBAAR MOET ZIJN: HEADER.



# POSITION: FIXED

## *Praktisch voorbeeld*

Voorbeeld van een website: een fixed header die plakt

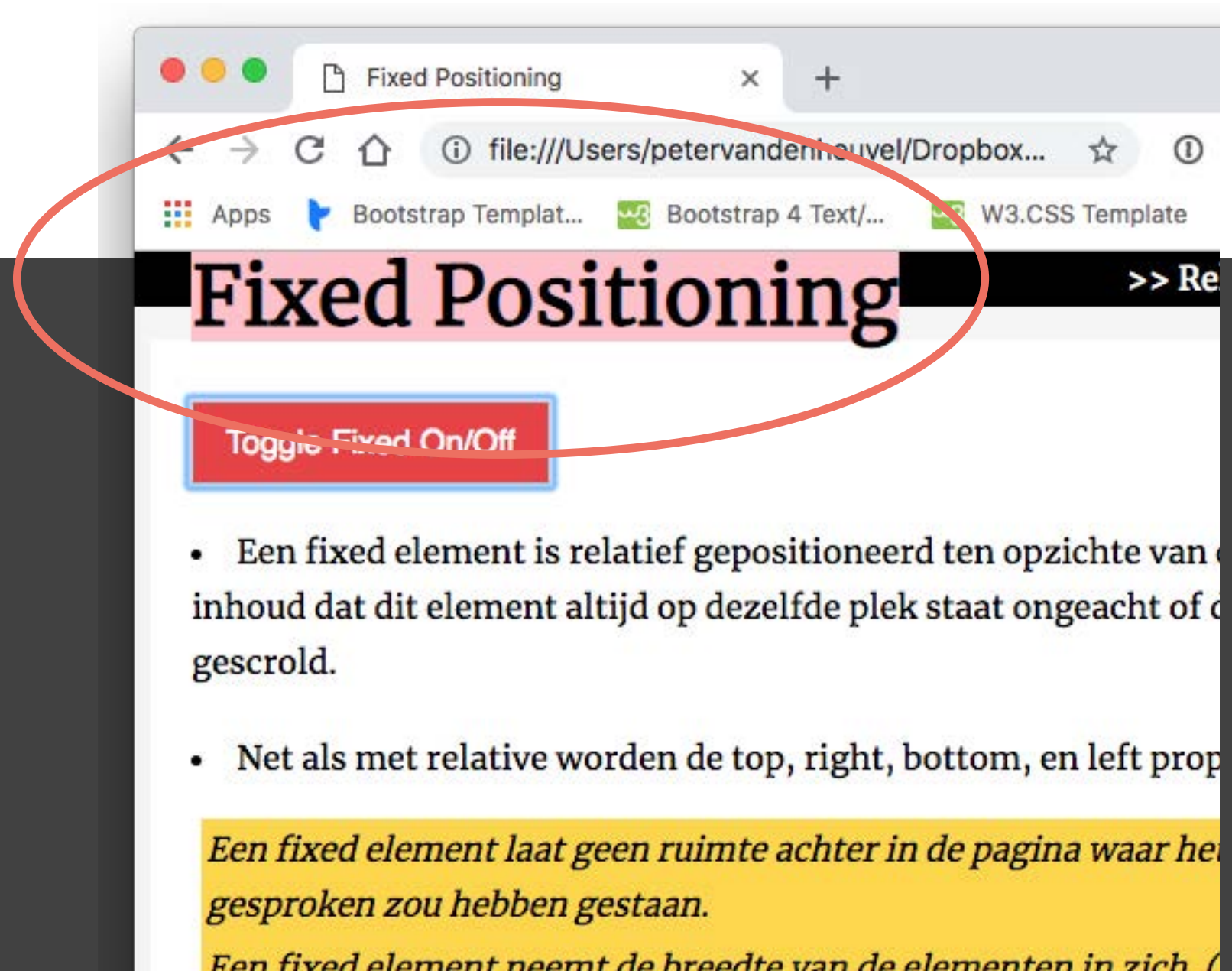


# FIXED POSITIONING

*het element wordt **uit** de flow gehaald.*

*Het volgende element gedraagt zich alsof het niet bestaat.  
(Net als bij absolute positioning)*

```
.HEADER {  
  POSITION: FIXED;  
  TOP: 0PX;  
  LEFT: 0PX;  
}
```





# FIXED POSITIONING

*De breedte van het fixed-element wordt  
**net zo breed** als de **inhoud**.*

*Bij het voorbeeld van de header, geef hem een width van 100%:*

```
.HEADER {  
  POSITION: FIXED;  
  TOP: 0PX;  
  LEFT: 0PX;  
  WIDTH: 100VW OF 100%;  
}
```

# OVERLAPPING

*beïnvloeden*

---

```
.TOP {  
  POSITION: RELATIVE;  
  Z-INDEX: 1;  
}
```

```
.BOTTOM {  
  POSITION: RELATIVE;  
  Z-INDEX: 0;  
}
```

# Z-INDEX

- 1) *Gebruik z-index om de stapelvolgorde van gepositioneerde elementen te beïnvloeden.*
- 2) *Let op: 'gepositioneerde elementen', anders werkt het niet.  
Dat wil zeggen; alles behalve 'static' dus.*
- 3) *Hoe hoger de waarde; hoe dichterbij de kijker.  
999 is bovenop de stapel, 0 is onderop. Of -2, nog dieper.*



```
.GREEN {  
  POSITION: RELATIVE;  
  Z-INDEX: 1;  
}
```

```
.RED {  
  POSITION: RELATIVE;  
  Z-INDEX: 2;  
}
```

# Z-INDEX

*alle elementen op elkaar*

- *Niet-gepositioneerde elementen (position:static) hebben **GEEN** z-index, ofwel een z-index van 0.*
- *Bij een **gelijke** z-index komt het **laatste** element in de broncode (HTML) hoger te liggen in de stapelvolgorde.*

<https://codepen.io/petervandenheuvel/pen/KjZYbW>

## 4 FLOATS

---



# FLOATS

Werden

- *~~Floats~~ worden nog het meest gebruikt om kolommen-layouts te maken.*  
(Bijvoorbeeld in Bootstrap 3, of iedere website van voor 2016)
- *Maar daar zijn ze oorspronkelijk niet voor bedoeld.*



# KLASSIEKE FLOAT:

*een afbeelding in een stuk tekst*

```
img {  
  float: right;  
}
```

Historisch gezien zijn float voor deze toepassing in CSS geschreven.

## What are floats for anyway?



Hello there! Come here my little friend. Don't be afraid. Don't worry, he'll be all right. What happened? Rest easy, son, you've had a busy day. You're fortunate you're still in one piece. Ben? Ben Kenobi? Boy, am I glad to see you! The Jundland wastes are not to be traveled lightly. Tell me young Luke, what brings you out this far? Oh, this little droid! I think he's searching for his former master. You must learn the ways of the Force if you're to come with me to Alderaan. Alderaan! I'm not going to Alderaan. I've got to go home. It's late, I'm in for it as it is. I need your help, Luke. She needs your help. I'm getting too old for this sort of thing. I can't get involved. I've got work to do. It's not that I like the Empire. I hate it! But there's nothing I can do about it right now. It's such a long way from here. That's your uncle talking. Oh, God, my uncle. How am I ever going to explain this? I've never seen such devotion in a droid before. There seems to be no stopping him. He claims to be the property of an Obi-Wan Kenobi. Is he a relative of yours? Do you know who he's talking about?

De afbeelding staat in de tekst (inline)

## What are floats for anyway?

Hello there! Come here my little friend. Don't be afraid. Don't worry, he'll be all right. What happened? Rest easy, son, you've had a busy day. You're fortunate you're still in one piece. Ben? Ben Kenobi? Boy, am I glad to see you! The Jundland wastes are not to be traveled lightly. Tell me young Luke, what brings you out this far? Oh, this little droid! I think he's searching for his former master. You must learn the ways of the Force if you're to come with me to Alderaan. Alderaan! I'm not going to Alderaan. I've got to go home. It's late, I'm in for it as it is. I need your help, Luke. She needs your help. I'm getting too old for this sort of thing. I can't get involved. I've got work to do. It's not that I like the Empire. I hate it! But there's nothing I can do about it right now. It's such a long way from here. That's your uncle talking. Oh, God, my uncle. How am I ever going to explain this? I've never seen such devotion in a droid before. There seems to be no stopping him. He claims to be the property of an Obi-Wan Kenobi. Is he a relative of yours? Do you know who he's talking about?



De tekst vloeit langs de afbeelding.

<https://codepen.io/petervandenheuvel/pen/gqxoRZ?editors=1100>

# FLOATS:

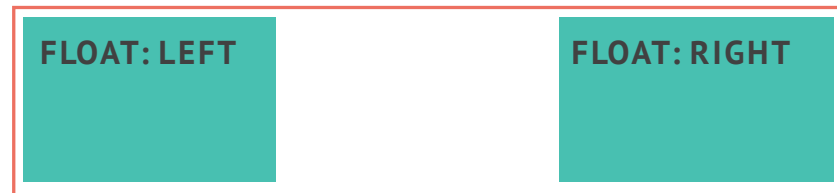
*Parent elements herkennen de hoogte van floated elementen niet, en respecteren enkel de hoogte van non-floated child elements*

**FLOAT: LEFT**

**NOT FLOATED TEXT, LO-  
REM IPSUM LOREM IPSUM  
LOREM IPSUM LOREM**

# CLEARING FLOATS:

- 1) Gebruik 'overflow' met de waarden 'auto' of 'hidden'
- 2) Gebruik één van de *clearfix-hacks* of



```
1 <div class="parent clearfix">
2   <p>floated element</p>
3   <p>floated element</p>
4 </div>
5
6 <style>
7   .clearfix:after {
8     content: "";
9     display: table;
10    clear: both;
11  }
12 </style>
```

ALLES OVER CLEARFIX, EN AL ZIJN VARIANTEN:

<https://css-tricks.com/snippets/css/clear-fix/>