

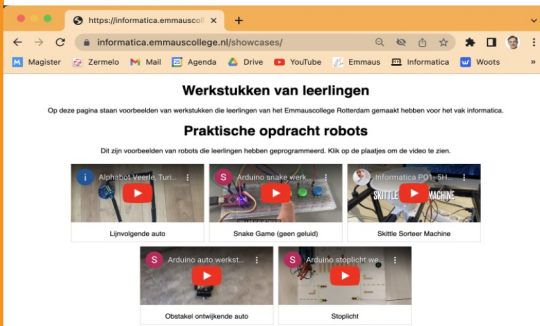
Wat gebeurt er als je de computer opstart?



1. de computer gaat aan.

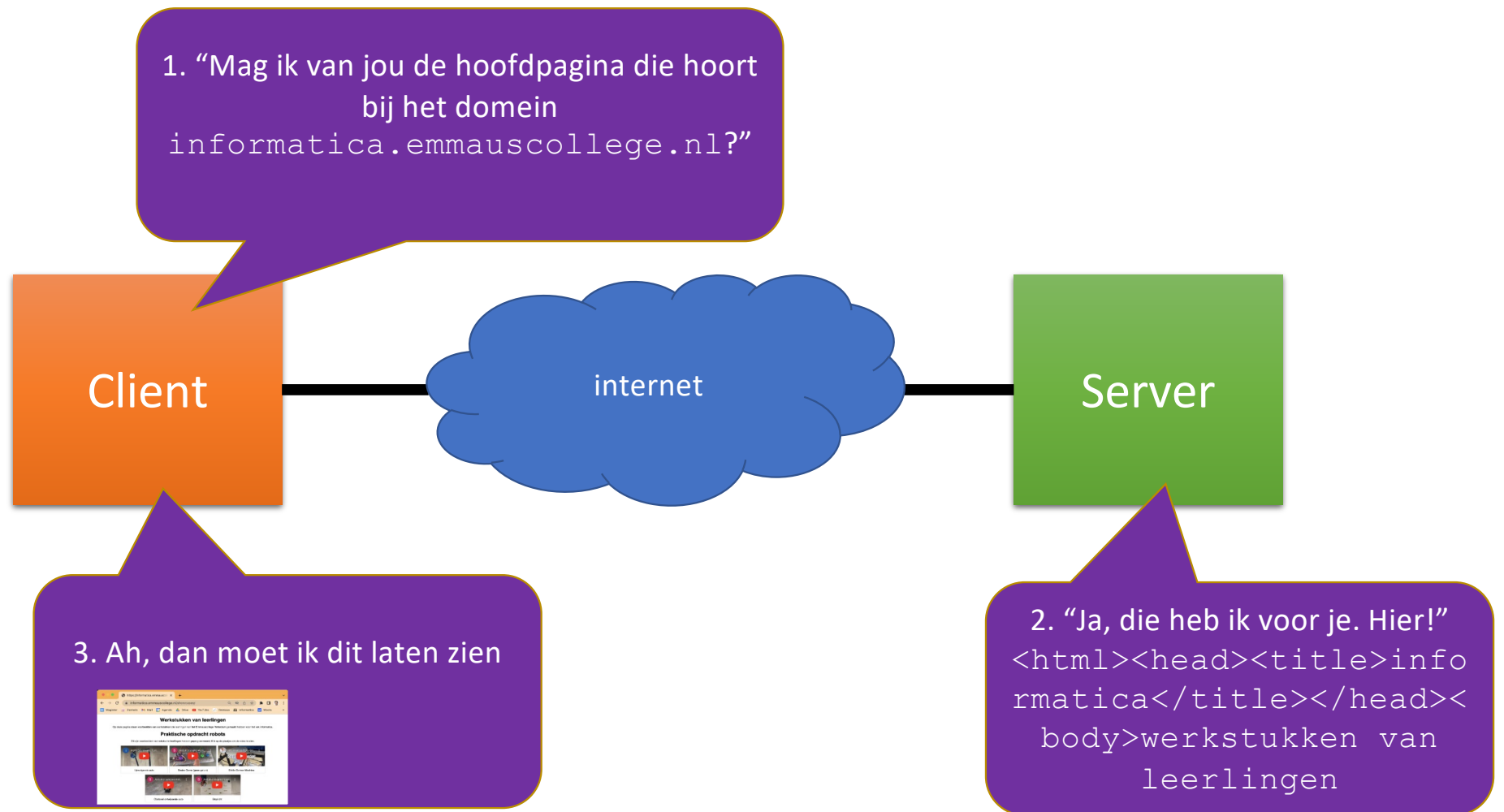


1. het operating system wordt geladen.



1. je start een browser en gaat naar een website

Wat doet een browser?





HTML tag, attribute, value

HTML

HTML is een computertaal waarmee je de inhoud van een webpagina beschrijft.

- Tag: geeft aan wat voor inhoud een stuk webpagina bevat.
 - Opening tag: geeft begin van stuk aan
 - Closing tag (met /): geeft eind van stuk aan
- Attribute: een eigenschap die je aan een tag kunt geven
- Value: de waarde die je de eigenschap wilt geven, bijvoorbeeld een link.

HTML voorbeeld

```
<A HREF="contact.html">
```

Contactinformatie

```
</A>
```

Uitzondering: van
 en bestaan geen closing tags.

Een tag kan meer dan één attribute hebben.



Parent, Child, Sibling

```
<html>
  <head>
    <title>  Vacatures  </title>
  </head>
  <body>
    <p>  Voordelen als je html kent  </p>
    <ol>
      <li>  Leuk werk  </li>
      <li>  Lekker verdienen  </li>
    </ol>
  </body>
</html>
```

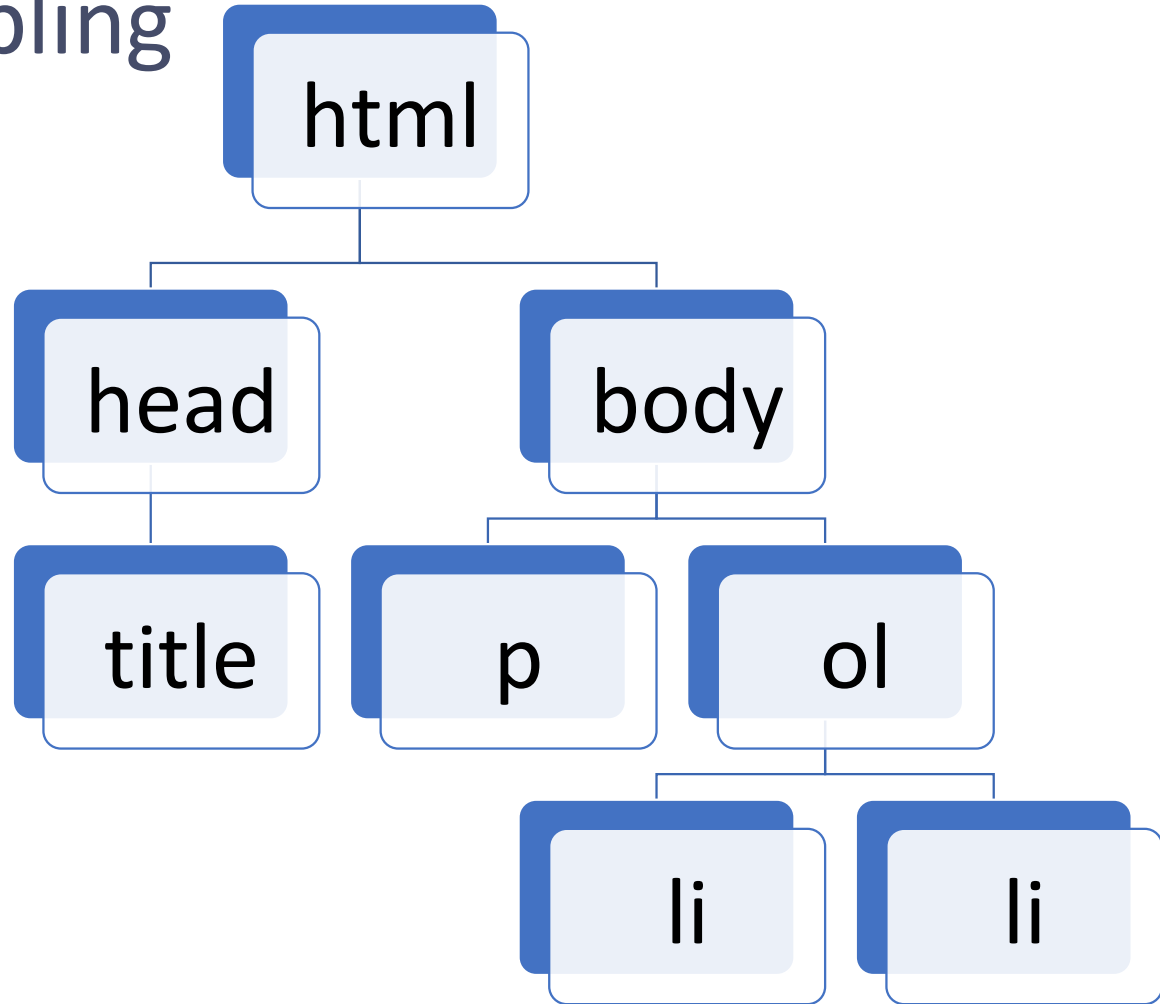
23-09-2022

Parent, Child, Sibling (stamboom)

- Je kunt Parents, Children en Sibling in een boomstructuur (stamboom) beschrijven.
- Schrijf de navolgende tekst in de vorm van een stamboom

```
<html>  <head>  <title>  Vacatures  </title>
</head>  <body>  <p>  Voordelen als je html kent
</p>  <ol>  <li>  Leuk werk  </li>  <li>  Lekker
verdiene  </li>  </ol>  </body>  </html>
```

Parent, Child, Sibling



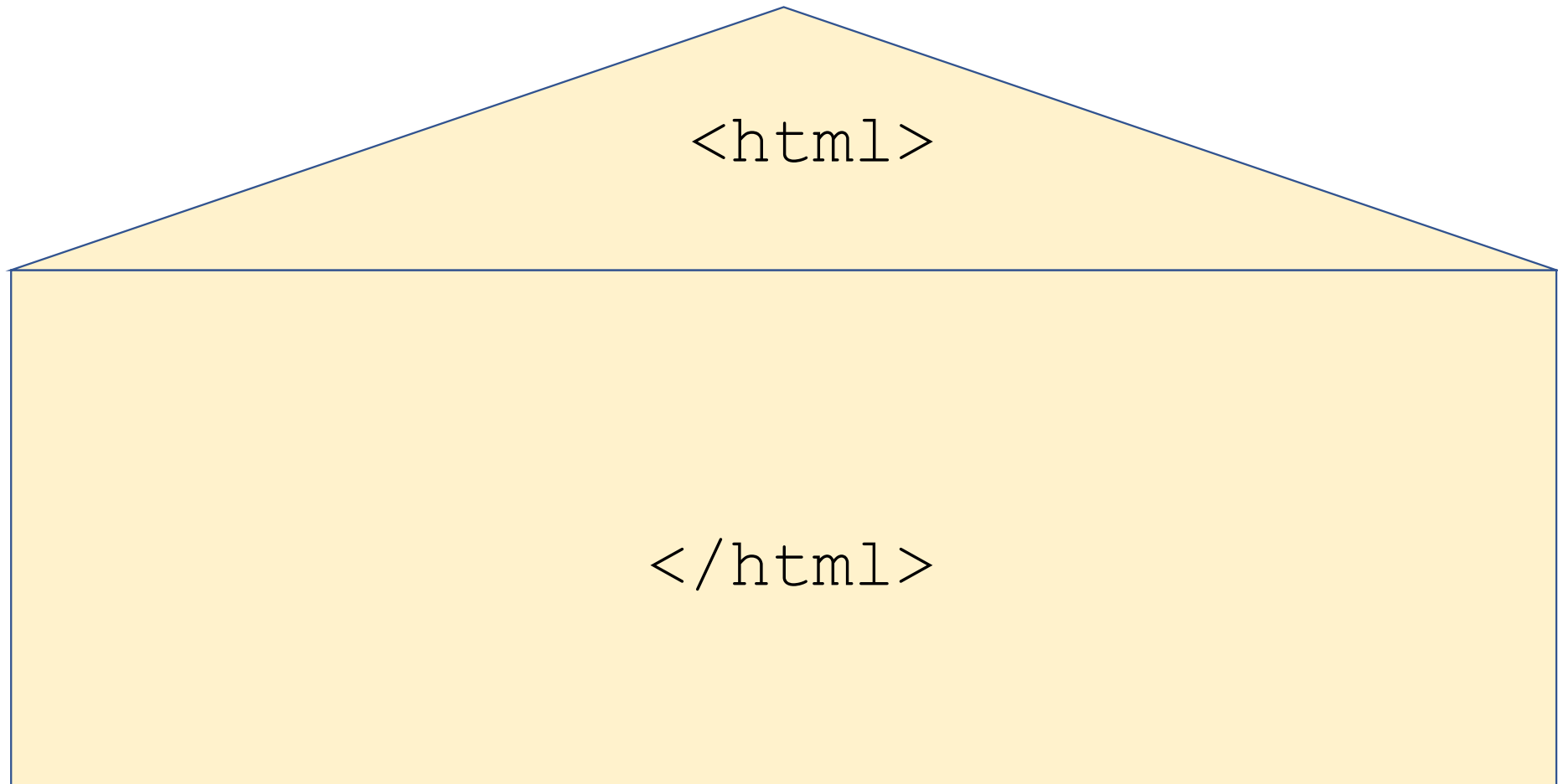
Parent, Child, Sibling (enveloppen)

Je kunt Parents, Children en Sibling met enveloppen beschrijven.

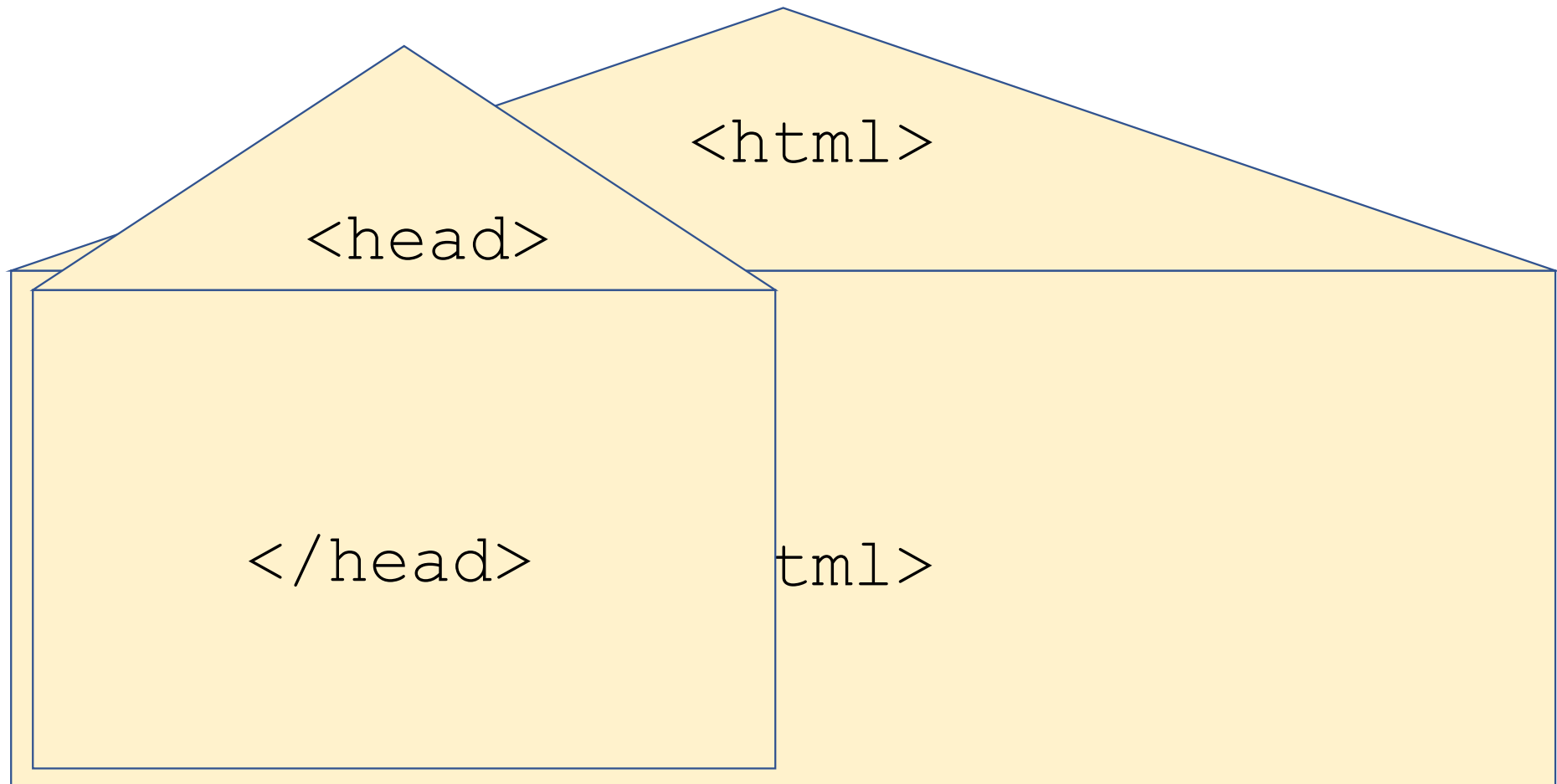
1. Knip de html-tags en de stukjes tekst hieronder los. Je heb dan 20 stukjes.
2. Pak een enveloppe: schijf de eerste opening-tag “<html>” op de voorkant en de bijbehorende closing-tag “</html>” op de achterkant. Neem een grote enveloppe.
3. Zoek de eerste tag na “<html>”, dit is het eerste child van “<html>”. Kijk of er nog meer children zijn van “<html>”. Maak voor elk child een kleine enveloppe en stop die in de grote enveloppe van hun parent.
4. Ga zo door, met steeds kleinere enveloppen, tot je bij children bent die geen parent meer zijn. Alle tags zijn dan op. De 4 losse teksten kun je in de enveloppe stoppen met de tag waar ze tussen staan.

```
<html> <head> <title> Vacatures </title> </head> <body> <p>  
Voordelen als je html kent </p> <ol> <li> Leuk werk </li> <li>  
Lekker verdienen </li> </ol> </body> </html>
```

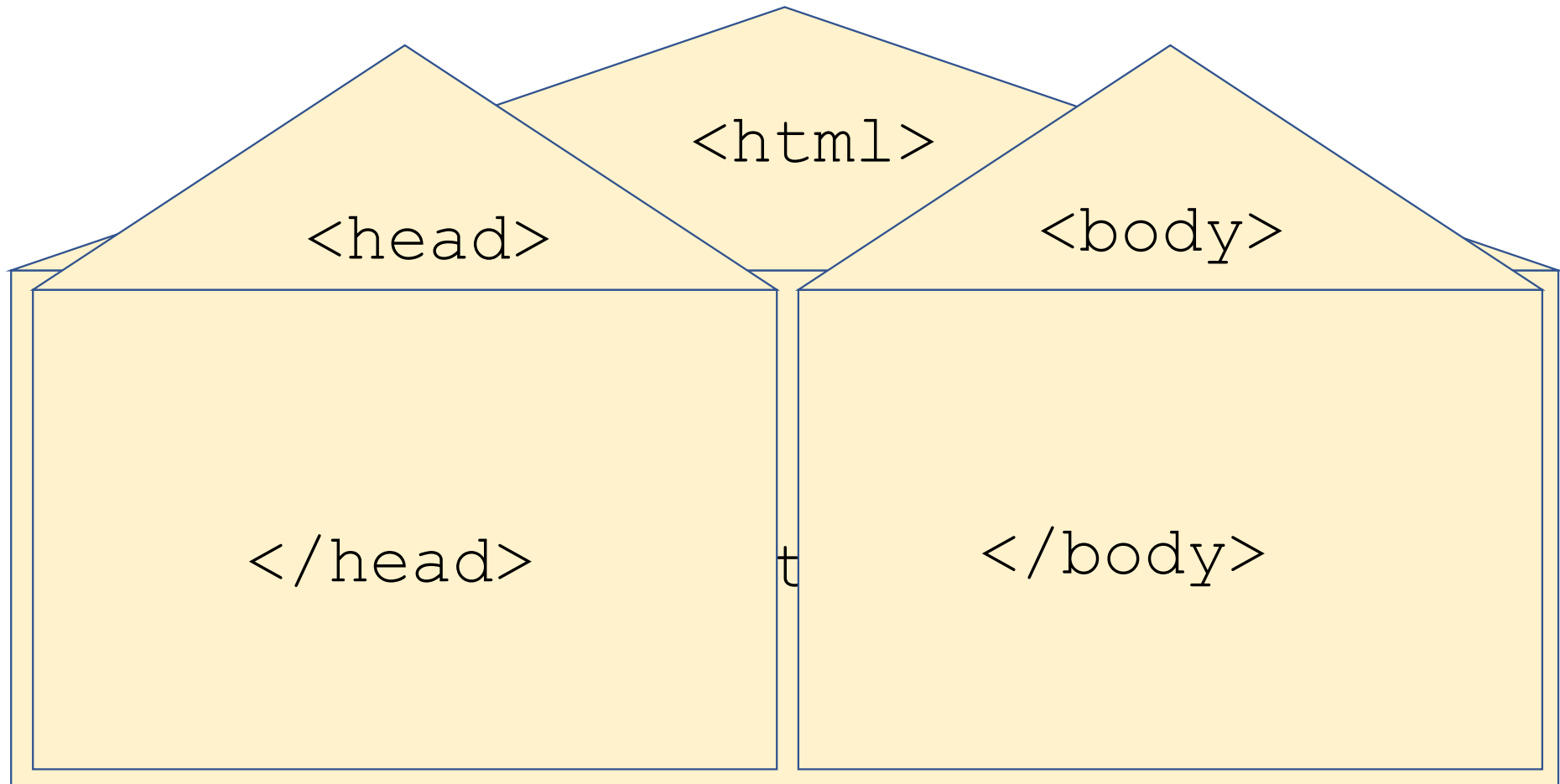
Parent, Child, Sibling



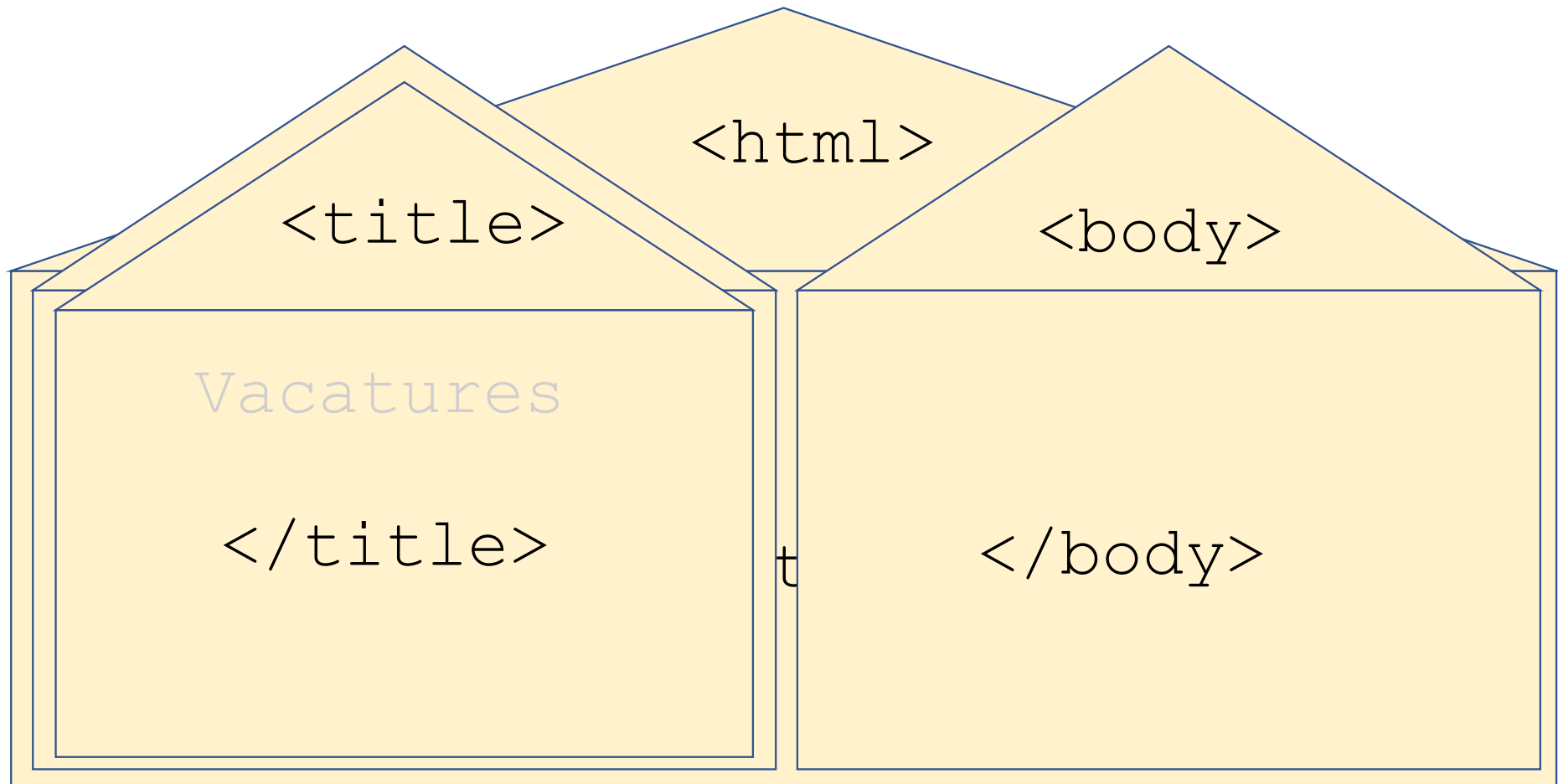
Parent, Child, Sibling



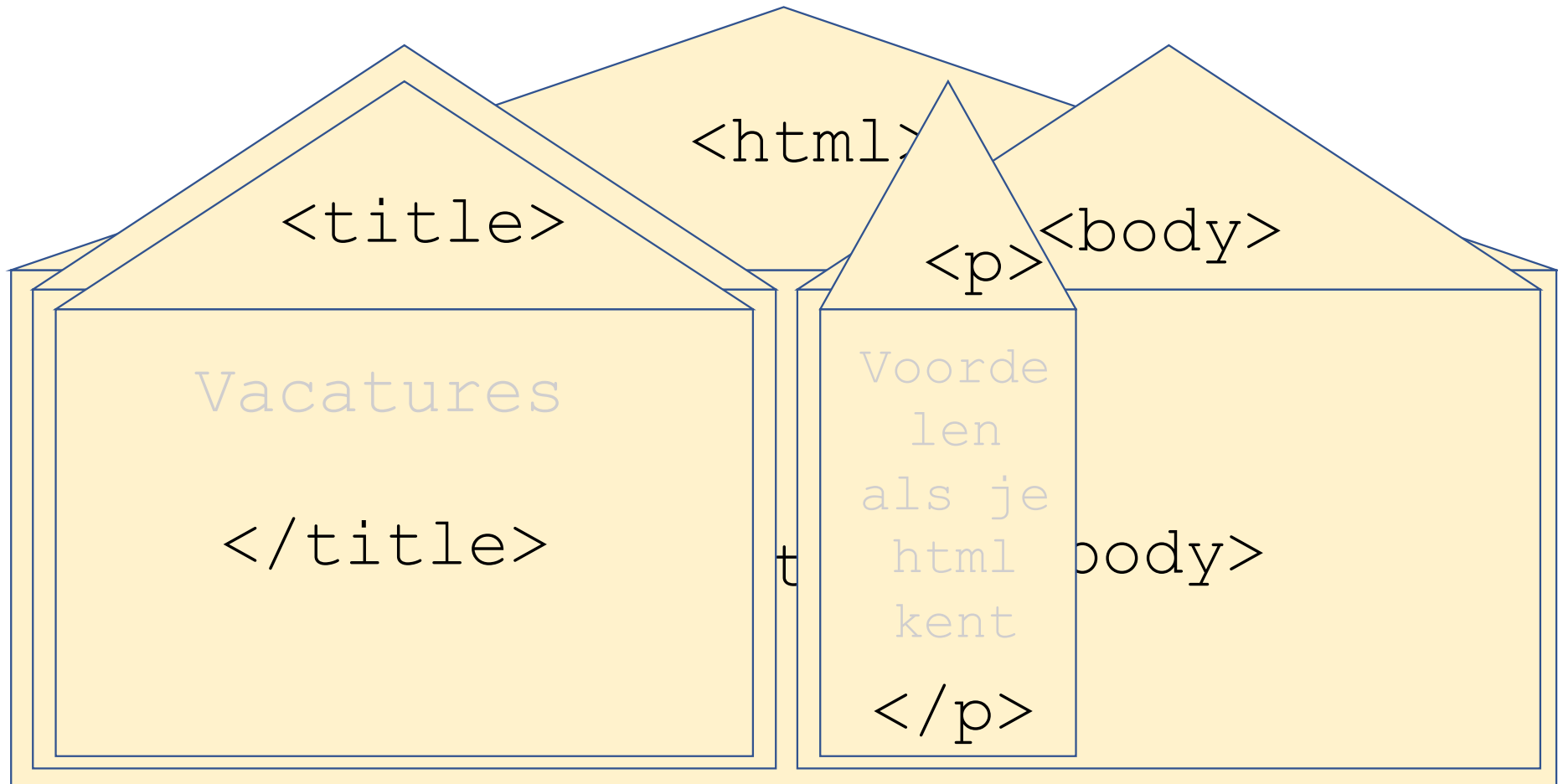
Parent, Child, Sibling



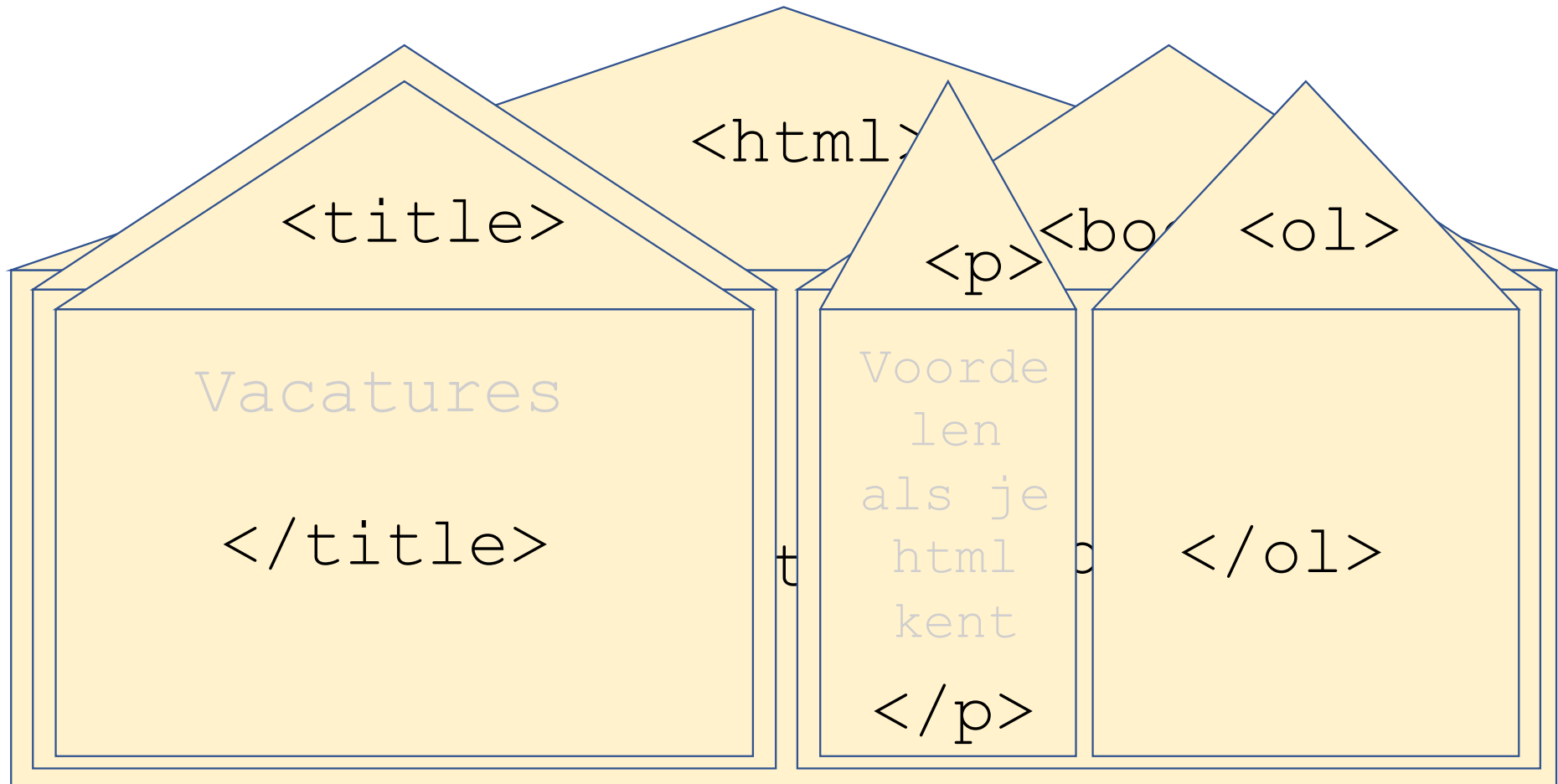
Parent, Child, Sibling



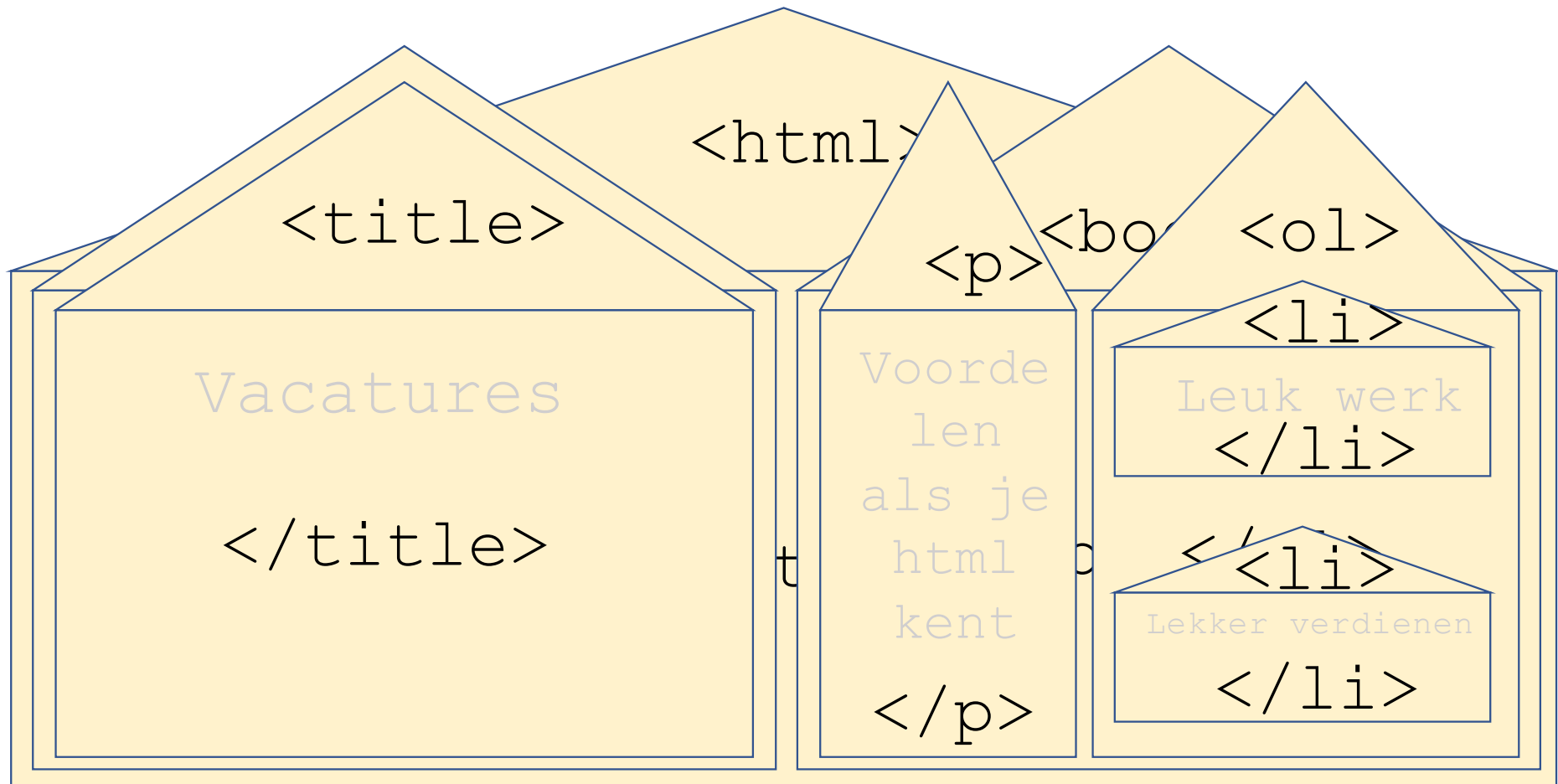
Parent, Child, Sibling



Parent, Child, Sibling



Parent, Child, Sibling



Samenvatting Parent, Child, Sibling

- Een **Parent** is een element in een programmeertaal, dat één of meerdere andere elementen (**Children**) bevat.
- Children van dezelfde Parent noemen we **Sibling**.
- Een element dat een Child is, kan weer Parent zijn van andere elementen. Dit fenomeen noemen we **Nesting**.
- Parent, Child en Sibling zijn Engelse begrippen. In het Nederlands zou je kunnen zeggen: Ouder, Kind en Zus/Broer. Binnen de informatica is gebruikelijk om de Engelse begrippen te gebruiken.



Separation of Concerns

Wat heeft spaghetti met HTML en CSS te maken?



Separation of Concerns

Separation of Concerns in de informatica betekent dat je software of hardware modulair opbouwt.

Anders gezegd: Code die je maakt moet je opdelen in onderdelen die elk een duidelijk af te bakenen functie hebben.

Door Separation of Concerns kun je:

- Code die anderen gemaakt hebben sneller te begrijpen
- Code later gemakkelijker aanpassen

Toepassing van SoC bij HTML+CSS

- HTML beschrijft de inhoud, wat er op de webpagina staat
- CSS beschrijft de opmaak, hoe het er op de webpagina uit moet zien

Scheiden van inhoud en opmaak

In moderne webpagina's worden inhoud en opmaak gescheiden.

- HTML beschrijft de inhoud, wat er op de webpagina staat
- CSS beschrijft de opmaak, hoe het er op de webpagina uit moet zien

Voordelen:

- Webpagina's die anderen gemaakt hebben kun je sneller te begrijpen
- Webpagina's kun je later gemakkelijker aanpassen



CSS selector, property, value

CSS

CSS is een computertaal waarmee je de opmaak van een webpagina beschrijft.

- Selector: wijst aan welk stuk html je wilt aanpassen, bijvoorbeeld alle tekst tussen `<h1>` en `</h1>`
- Property: geeft aan welke eigenschap je wilt aanpassen, bijvoorbeeld `color`
- Value: de waarde die je de eigenschap wilt geven, bijvoorbeeld `red`.

CSS voorbeeld

```
h1 {  
  color: red;  
}
```

CSS selector

Er zijn 3 soorten selectoren:

- Tags
- Classes
- ID's

HTML voorbeeld

```
<h1 class="title">  
rode letters gele achtergrond  
</h1>
```

```
<p ID="publish-time">  
grijs  
</p>
```

CSS voorbeeld

```
h1 {  
    color: red;  
}  
  
.title {  
    background-color: yellow;  
}  
  
#publish-time {  
    color: gray;  
}
```


CSS selector specificiteit

Als er meerdere selectoren naar hetzelfde stuk html verwijzen, dan bepaalt de meest specifieke selector hoe de layout wordt.

- ID's zijn specifieker dan Classes.
- Classes zijn specifieker dan Tags.

HTML voorbeeld:

```
<h1 class="title"
      ID="publish-time">
```

Grijze letters, gele achtergrond

```
</h1>
```

CSS voorbeeld

```
h1 {
    color: red;
}

.title {
    background-color: yellow;
}

#publish-time {
    color: gray;
}
```

HTML met CSS voorbeeld

HTML voorbeeld

```
<h1 class="title">
  Roosterwijziging
</h1>
<p>
  <div id="publish-time">
    Maandag 08:30
  </div>
</p>
<h1>
  Eerste lesuur
</h1>
<p>
  vervallen
</p>
```

CSS voorbeeld

```
h1 {
  color: green;
}
p {
  color: blue;
}
.title {
  color: red;
}
#publish-time {
  color: gray;
}
```



Boxen

Boxen

- HTML / CSS kent twee soorten boxen.
 - Inline elementen staan op een regel.
 - Een block is een onzichtbare rechthoek op het scherm waarbinnen een stukje html wordt weergegeven.
- Alle stukjes HTML wordt in boxen gezet, standaard over de volle breedte van het scherm, van boven naar beneden.
- *Children* zijn kleine boxen binnen de box van hun *Parent*
- Je kunt de afmeting van boxen aanpassen, bijvoorbeeld om meerdere kolommen te maken.

Inline en block elementen

Inline elementen

- Elementen worden op 1 regel geplaatst
- ``, ``, ``, ...



Dit is een korte tekst om te laten weten dat jullie allemaal geweldig zijn!

Block elementen

- Blokken worden onder (of naast) elkaar geplaatst
- `<h1>`, `<p>`, `<div>`, ...



Dit is een korte tekst om te laten weten dat jullie allemaal geweldig zijn!

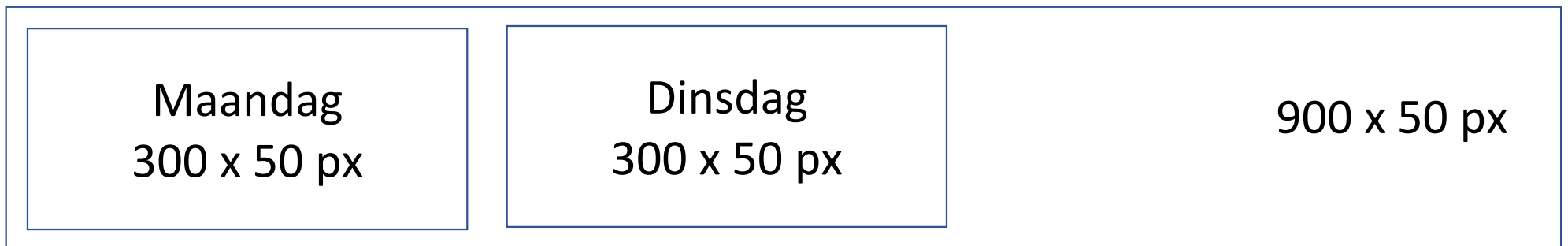
Op block-elementen kun je alle CSS-properties die bij het box-model horen toepassen.

Voorbeeld van gebruik van boxen

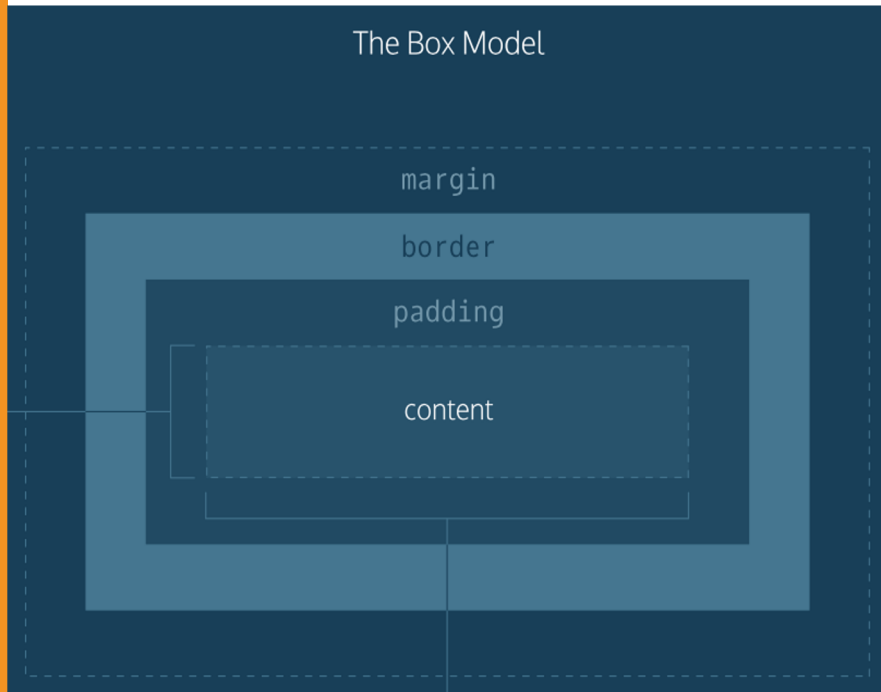
```
<body>
  <p class="box">
    Maandag
  </p>
  <p class="box">
    Dinsdag
  </p>
</body>
```

```
.box {
  width: 300px;
  height: 50px;
}
```

body heeft ook een box, het is de parent van *p*



Layout van boxen aanpassen



Een box bestaat uit 4 gebieden

- Content = Inhoud, plaats voor tekst in de box
- Padding = afstand tussen tekst en rand van de box
- Border = rand van de box
- Margin = leeg gebied, afstand met andere boxen

- Het uiterlijk van alle box onderdelen kun je veranderen

CSS-properties voor afmeting van boxen

Enkele voorbeelden:

- `width: 100px;`
Breedte 100 pixels
- `width 50%;`
Breedte 50% van de parent box
- `height: 200px;`
Hoogte 200 pixels
- `box-sizing: content-box;`
De opgegeven breedte en hoogte gaan over content, exclusief border+padding (dit is de standaard instelling).
- `box-sizing: border-box;`
De opgegeven breedte en hoogte gaan over content+border+padding.



Layout met float

Blokken op de juiste plek zetten met behulp van float

Inline en block elementen

Inline elementen

- Elementen worden op 1 regel geplaatst
- ``, ``, ``, ...



Dit is een korte tekst om te laten weten dat jullie allemaal geweldig zijn!

Block elementen

- Blokken worden onder (of naast) elkaar geplaatst
- `<h1>`, `<p>`, `<div>`, ...

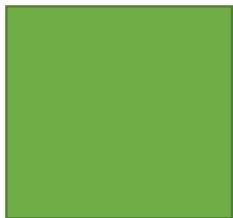


Dit is een korte tekst om te laten weten dat jullie allemaal geweldig zijn!

Op block-elementen kun je alle CSS-properties die bij het box-model horen toepassen.

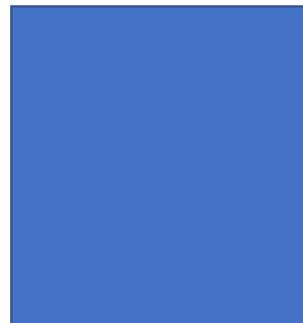
Float gebruiken bij plaatjes

Plaatje zonder float



Dit is een korte tekst om te laten weten dat jullie allemaal geweldig zijn!

Plaatje met float



Dit is de tweede paragraaf, die is iets langer.

Dit is de eerste paragraaf.

Dit is de volgende paragraaf na clear.

Als je een plaatje float, dan wordt het een block-element in plaats van een inline-element. Je kunt dat in CSS doen met:

```
float: left;      of      float: right;
```



Als je wilt dat het floaten stopt, zodat de volgende paragraaf (of ander block) onder het plaatje wordt geplaatst, dan kan dat door aan die paragraaf in css het volgende toe te voegen:

```
clear: both;
```

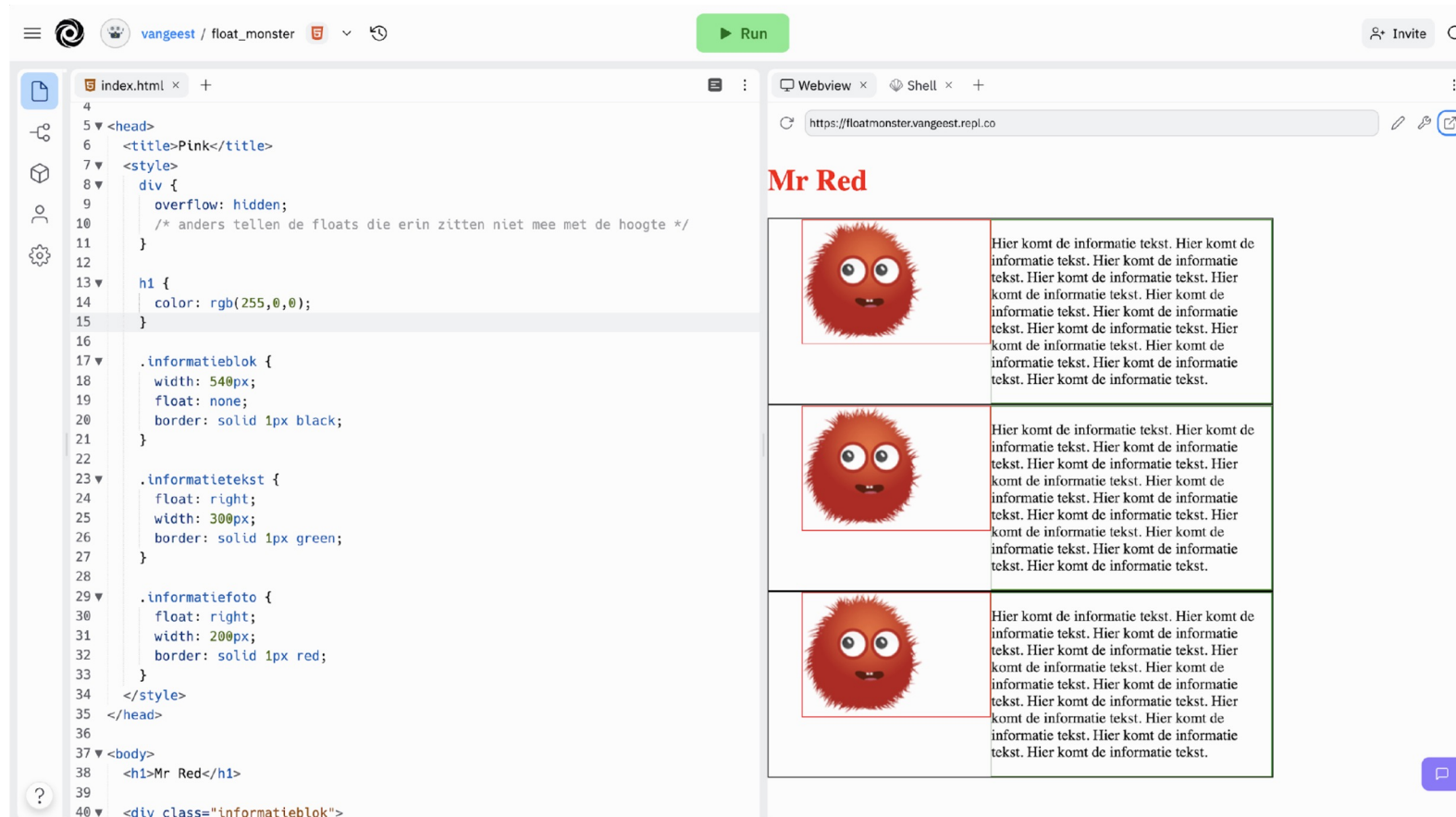
23-09-2022

Layout maken met float

Mr Red

	<p>Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst.</p>
	<p>Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst.</p>

Layout maken met float



The screenshot shows a Replit editor with a code editor on the left and a web browser on the right. The code editor displays the following HTML and CSS:

```
4  
5 <head>  
6 <title>Pink</title>  
7 <style>  
8   div {  
9     overflow: hidden;  
10    /* anders tellen de floats die erin zitten niet mee met de hoogte */  
11  }  
12  
13  h1 {  
14    color: rgb(255,0,0);  
15  }  
16  
17  .informatieblok {  
18    width: 540px;  
19    float: none;  
20    border: solid 1px black;  
21  }  
22  
23  .informatietekst {  
24    float: right;  
25    width: 300px;  
26    border: solid 1px green;  
27  }  
28  
29  .informatiefoto {  
30    float: right;  
31    width: 200px;  
32    border: solid 1px red;  
33  }  
34 </style>  
35 </head>  
36  
37 <body>  
38 <h1>Mr Red</h1>  
39  
40 <div class="informatieblok">
```

The browser window shows the rendered output: a red heading "Mr Red" followed by a table with three rows. Each row contains a red, spiky monster image on the left and a block of placeholder text on the right. The text in each cell reads: "Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst. Hier komt de informatie tekst."

23-09-2022

<https://replit.com/@vangeest/floatmonster>

Layout maken met float (genest)

Voorbeeld layout met float

Op deze pagina zie je een voorbeeld hoe je blokken een plaats kunt geven met de css-property float.

Zorg dat je het box-model van css begrijpt voordat je met dit voorbeeld aan de slag gaat.

De div-tag gebruik je om stukjes html te groeperen. Aan de div-tags kennen we classes toe. We gebruiken classes die stapelen en classes die floaten. Stapelen betekent stukjes html onder elkaar zetten. Floaten betekent stukjes html naast elkaar zetten, zolang het past.

Door de div-tags te nesten kun je (bijna) alle layouts maken die je wilt. Nesten van div-tags betekent dat je div-tags in div-tags stopt. Je maakt dan parents, children en siblings van de div-tags.

De class container bevat css elementen die in alle andere classes worden gebruikt. Een class die stapelt of float wordt altijd gecombineerd met de class container. Door het gebruik van deze extra container-class voorkomen we dat we dezelfde css-elementen moeten herhalen in alle andere classes voor stapelen en floaten.

Als je zelf een layout gaat maken, dan is het handig om tijdelijk een lijn (border) van een bepaalde kleur om elk blok te zetten. Dan kun je goed zien of de blokken de afmeting hebben die je wilt.

Bekijk de broncode van onderstaande voorbeeld en pas deze aan naar je eigen wensen om te leren hoe het precies in elkaar zit.

Bovenste rij eerste kolom. De eerste box met tekst zie je hier. Vind je hem mooi?	Bovenste rij tweede kolom. De tweede box met tekst zie je hier. Mooi toch?	Bovenste rij derde kolom. De 3e box met tekst zie je hier. Geweldig!
Tweede rij linker kolom met een vaste breedte van 200 pixels. Vind je hem mooi? En hier nog meer tekst bla	Tweede rij rechter kolom vult de rest van de ruimte. Dus deze rechter kolom is de volle breedte van de parent container min 200 pixels van de linker kolom. De eerste gestapelde box met tekst zie je hier. Top!	
	Er zitten drie gestapelde containers in de rechter kolom. Alle drie zitten in een parent div die float. De tweede gestapelde box met tekst zie je hier. Super!	
	De derde gestapelde box met tekst zie je hier. Geweldig!	

Dit is een paragraaf onderaan, deze zit niet in een div.

Layout maken met float (genest)

