



HTML5, CSS3 en JavaScript

Auteurs	Linda B ; content VO-
Team	Wikiwijs Maken Auteurs
Laatst gewijzigd	9 juli 2019
Licentie	CC Naamsvermelding-GelijkDelen 4.0 Internationale licentie
Webadres	https://maken.wikiwijs.nl/80244/



Dit lesmateriaal is gemaakt met Wikiwijs van Kennisnet. Wikiwijs is hét onderwijsplatform waar je leermiddelen zoekt, maakt en deelt.

Inhoudsopgave

A Inleiding	3
Vooraf	3
Leerdoelen	3
Zo werkt het	4
B1 HTML5	5
Een sjabloon voor HTML5	5
★ Aan de slag 1	6
★ Aan de slag 2	6
★ Aan de slag 3	7
Nesten	8
★ Aan de slag 4	8
Controleren van code	9
★ Aan de slag 5	10
Elementen en attributen	10
★ Aan de slag 6	12
Kopteksten	12
★ Aan de slag 7	13
B2 HTML5	14
Alinea's	14
★ Aan de slag 8	15
★ Aan de slag 9	15
Afbeeldingen	17
★ Aan de slag 10	18
Verwijzingen	19
★ Aan de slag 11	20
★ Aan de slag 12	20
Lijsten	20
★ Aan de slag 13	21
★ Aan de slag 14	22
★ Aan de slag 15	23
Tabellen	23
★ Aan de slag 16	24
★ Aan de slag 17	25
★ Aan de slag 18	25
Video en audio	25
★ Aan de slag 19	27
Structuur	27
★ Aan de slag 20	28
Toets 1	28
C CSS3	29
Inleiding	29
Syntax van CSS	29
★ Aan de slag 21	30
Koppelen CSS en HTML	31
★ Aan de slag 22	31
★ Aan de slag 23	32
Extern stylesheet	32
★ Aan de slag 24	33
★ Aan de slag 25	34
★ Aan de slag 26	34
★ Aan de slag 27	35
Eigenschappen lettertypen	35
★ Aan de slag 28	37
Eigenschappen tekst	37
★ Aan de slag 29	39
★ Aan de slag 30	40
★ Aan de slag 31	40

Verwijzingen	41
Het box model	42
★ Aan de slag 32	43
Positionering elementen	45
★ Aan de slag 33	49
★ Aan de slag 34	49
★ Aan de slag 35	50
Een menu in HTML en CSS	51
★ Aan de slag 36	52
Toets 2	53
D JavaScript	54
Client-side scriptingtaal	54
JavaScript-code	54
★ Aan de slag 37	55
Een eenvoudig script	56
★ Aan de slag 38	58
★ Aan de slag 39	58
★ Aan de slag 40	59
★ Aan de slag 41	59
★ Aan de slag 42	60
Commando's	60
★ Aan de slag 43	62
★ Aan de slag 44	63
Popup schermen	63
★ Aan de slag 45	66
★ Aan de slag 46	66
HTML5 en JavaScript	67
★ Aan de slag 47	70
Voortgang	71
Over dit lesmateriaal	72

A Inleiding

Vooraf

HTML is een taal die gebruikt wordt voor het maken van webpagina's. Webpagina's kunnen worden bekeken in een browser. Voorbeelden van browsers zijn Internet Explorer, Safari, Google Chrome, Opera en Mozilla Firefox.

De allereerste versie van HTML, de afkorting staat voor Hypertext Markup Language, werd in 1990 ontwikkeld door de Britse wetenschapper Tim Berners-Lee. In datzelfde jaar zag ook de eerste browser het levenslicht. Het programma kreeg de toepasselijke naam "WorldWideWeb".

In deze module gaan we aan de slag met HTML5. Dit is momenteel de meest recente versie van HTML. Voor het schrijven van de HTML-code gebruiken we onze interne HTML5-tool en het programma Notepad++.

Je kunt Notepad++ gratis downloaden via notepad-plus-plus.org/.

Bij het bouwen van een website heb je altijd met 3 aspecten te maken:

1. De structuur van de webpagina.
2. De opmaak (lay-out) van de webpagina.
3. De interactiviteit van de webpagina.

Voor het beschrijven van de structuur van webpagina's gebruiken we in dit lesmateriaal HTML5. De opmaak van een webpagina leggen we vast in zogenaamde stijlbladen (CSS3). HTML5 maakt gebruik van JavaScript. Dit is een scriptingtaal waarmee interactiviteit op een webpagina gerealiseerd kan worden. Daar ronden we deze module mee af.

In deze module leer je hoe je webpagina's kunt bouwen in HTML5 in combinatie met CSS3 en JavaScript.



Leerdoelen

Na verwerking van deze module kun je:

- Een sjabloon maken voor een HTML5-document;
- HTML-code valideren met de validator van W3C;
- Uitleggen wat het verschil is tussen elementen en attributen;
- Kopteksten en alinea's maken in HTML;
- Een afbeelding opnemen in een webpagina;
- Geordende en ongeordende lijsten maken in HTML;
- Verwijzingen maken in HTML;
- Een tabel maken in HTML;
- Uitleggen welke nieuwe structuurelementen in HTML5 worden gebruikt;
- Gebruik maken van CSS-regels voor de opmaak van webpagina's;
- Een externe stylesheet koppelen aan een HTML-bestand;
- Eigenschappen van lettertypen en tekst in CSS beschrijven;



- Uitleggen waarvoor het box model wordt gebruikt;
- Elementen positioneren met CSS;
- Een menu maken in HTML en CSS;
- Uitleggen waarvoor JavaScript gebruikt wordt;
- Een stukje JavaScript-code schrijven;
- Uitleggen wat het verschil is tussen een statement, een variabele, een functie en een object in JavaScript;
- Uitleggen waarvoor canvas wordt gebruikt.

Zo werkt het

Je bent begonnen met het doorwerken van de interactieve module **HTML5**.

Naast een inleiding bestaat de module uit drie onderdelen: "HTML5", "CSS3" en "JavaScript". In ieder onderdeel vind je, verdeeld over verschillende pagina's, informatie in de vorm van teksten, afbeeldingen en video's.

Daarnaast ga je zelf aan de slag. Onder het kopje "Aan de slag" vind je steeds toepassingsopdrachten. Deze opdrachten maak je alleen of met een klasgenoot.

De antwoorden op deze opdrachten bespreek je met elkaar of in de klas.

Er zijn ook toetsen. Deze herken je aan de blauwe knop met daarop "Adaptieve Toets".

Een toets bestaat uit meerdere vragen. Dat kunnen gesloten vragen zijn, die door de computer worden nagekeken, of open vragen, die moet je zelf nakijken.

Bij een enkele vraag moet je een bestand uploaden.

Van de toetsen wordt, als je ingelogd bent, de voortgang bijgehouden.

Het resultaat vind je onder de knop "Voortgang". Deze voortgang is ook door je docent te bekijken.

Om goed met deze module te kunnen werken, heb je een aantal bestanden nodig.

Deze bestanden kun je hier downloaden: [bestanden](#).

Sla de bestanden op in je homedirectory in de submap html. De submap html bestaat nog niet. Maak in je homedirectory dus eerst een submap html aan. In de map html sla je alle bestanden op die je tijdens het doorwerken van deze module zelf maakt.

Succes met de module HTML5.



B1 HTML5

Een sjabloon voor HTML5

Voor het uitwerken van de opdrachten maken we gebruik van een sjabloon (template) voor HTML5. De sjabloon functioneert als een voorbeelddocument.

Je voegt daaraan code toe voor het uitwerken van de diverse opdrachten.

Dat heeft als voordeel dat je niet telkens alle code opnieuw moet typen. De vereiste basis voor een HTML5-pagina heb je immers beschreven in de sjabloon.

Hieronder staat de broncode van ons voorbeelddocument:

```
<!DOCTYPE html>
<html lang="nl">
<head>
<title>sjabloon</title>
<meta charset="utf-8">
<link rel="stylesheet" href="https://maken.wikiwijs.nl/css/
opmaak.css">
</head>
<body>
</body>
</html>
```

Bekijk het volgende filmpje.

Voor extra uitleg staat de tekst van de video eronder.



<https://maken.wikiwijs.nl/userfiles/d/df69edb05f6c97ae2a6e151c38e014acf1ac7765.mp4>

Tekst van het filmpje

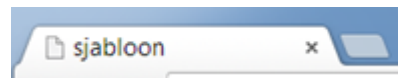
Bovenaan in de broncode van de sjabloon staat de regel `<!DOCTYPE html>`. Het doctype verschaft aan een browser informatie over wat voor type document het is. Daarnaast fungeert het doctype als een hulpmiddel voor de browser om de webpagina correct weer te geven. De regel `<!DOCTYPE html>` geeft aan dat we te maken hebben met een webpagina die gebruik maakt van HTML5.

Na het doctype volgt de regel `<html lang="nl">`. Daaraan kun je zien dat het om een webdocument in de Nederlandse taal gaat. Er had hier ook alleen `<html>` kunnen staan. Dat was ook voldoende geweest.

Om de browser onderscheid te laten maken tussen tekst en commando's wordt er in HTML gebruik gemaakt van zogenaamde "tags". Een tag is een opdracht die door de browser geïnterpreteerd kan worden. In HTML begint een tag altijd met een `<`-teken en wordt afgesloten met het `>`-teken. In HTML komen tags vaak in paren voor. In dat soort gevallen is er sprake van een begintag en een eindtag.

In de code van ons voorbeelddocument zien we de begintag <head> en even verderop de eindtag </head>. De eindtag begint altijd met </. In het headgedeelte van de sjabloon treffen we vervolgens een titel aan, daarna volgt informatie over de gebruikte karakter set en tot slot zien we een verwijzing naar het gebruikte CSS-bestand. In het CSS-bestand wordt de lay-out van de webpagina geregeld. We komen hier later in deze module uitgebreid op terug.

In de Nederlandse taal spreken we van een karakterset of tekenverzameling. Karakters worden door de computer opgeslagen als een getal. In een karakterset leg je vast welke afspraken daarover worden gemaakt. Zo wordt er bijvoorbeeld afgesproken welk karakter met welk getal wordt weergegeven. Een veel gebruikte karakterset is utf-8. Na informatie over de gebruikte karakterset staat in ons voorbeelddocument een verwijzing naar het CSS-bestand.



Na het head-gedeelte volgt het body-gedeelte van het HTML-document. Alles wat in de body van het HTML-document komt te staan, plaatsen we tussen de begintag <body> en de eindtag </body>. Dit is het deel van de pagina dat zichtbaar is als de broncode in een browser wordt verwerkt. Tot slot sluiten we het document met de tag </html>.

★ Aan de slag 1

HTML5 is een taal die nog volop in ontwikkeling is. De diverse browsers gaan dan ook nog verschillend om met het ondersteunen van HTML5. Op de website

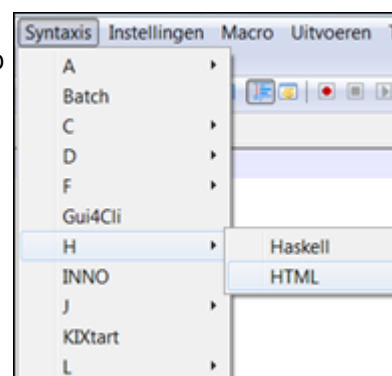
www.html5test.com kun je controleren in welke mate de browser die je gebruikt HTML5 ondersteunt. Er kunnen in totaal 555 punten behaald worden.

Open Internet Explorer en noteer hoeveel punten Internet Explorer van het totaal aantal te behalen punten scoort. Hoeveel punten scoort Google Chrome?



★ Aan de slag 2

- Open Notepad++ en klik in de menubalk van het programma op het tabblad Syntaxis.
Ga naar de letter H en klik op HTML.
Je hebt nu in Notepad++ aangegeven dat je code gaat schrijven voor een HTML-document.
Bij het opslaan van het bestand wordt nu automatisch de extensie .html toegevoegd.
Bekijk de code.



```
<!DOCTYPE html> <html lang="nl">
<head>
<title>sjabloon</title>
<meta charset="utf-8">
<link rel="stylesheet" href="https://maken.wikiwijs.nl/css/opmaak.css">
</head>
<body>
</body>
</html>
```

Selecteer de code. Kopieer en plak vervolgens de code in Notepad++.
Sla het bestand op in je homedirectory in de submap html onder de naam **template.html**.

Bekijk het resultaat in een browser. Dat kan door in Notepad++ onder het menu- item Uitvoeren te kiezen voor Run en daarna een keuze te maken voor de browser waarin je de webpagina wilt bekijken.

- b. Met HTML5-code kun je ook aan de slag in onze eigen HTML-editor.
Open hieronder de HTML-editor. Klik op de knop "Laad HTML" en blader naar je home-directory waar je net in de map html het bestand **template.html** hebt opgeslagen. Open het HTML-document. De code wordt in het HTML venster geladen. Klik vervolgens op de knop "Uitvoeren" en bekijk het resultaat in het onderste gedeelte.
Het is overigens ook mogelijk om in het HTML venster direct zelf code te schrijven en het resultaat daarvan te bekijken in het resultaat venster.



Zoals je hebt gezien is het mogelijk om HTML5-code in verschillende programma's te schrijven en te bewerken.

In deze module werk je met zowel Notepad++ en onze eigen HTML5-editor.
De editor gebruiken we vooral bij de wat kleinere opdrachten. Voor grote opdrachten ga je aan de slag met Notepad++ in combinatie met een browser.

★ Aan de slag 3

- a. Open de HTML5-editor. In het HTML vak zie je een niet complete HTML-code.
Bekijk de code en vul op de plaats van de puntjes de juiste tags in.
Klik nadat je de tags hebt ingevuld op "Uitvoeren" en bekijk het resultaat.



- b. In de tag <html = 'nl'> is de taal (nl = Nederlands) aangegeven.
Bedenk een reden waarom het handig is om de taal aan te geven.
- c. Er is ook een tag waarin de karakterset is aangegeven.
Welke tag is dat?
Zoek op welke functie een karakterset heeft?



Nesten

We kijken nog even naar de broncode van ons voorbeelddocument. Daarin is sprake van wat ook wel "nesten" wordt genoemd. Nesten (insluiten) houdt in dat de tag die het laatst geopend is als eerste weer gesloten moet worden. Dus nog voordat een andere tag wordt gesloten.

Bovenin de broncode wordt het HTML-document geopend met de tag `<html lang="nl">`. Hier had ook alleen de tag `<html>` kunnen staan. Daarna wordt het element `head` geopend met de begintag `<head>`. Dit element moet dus eerst gesloten worden voordat het `html`-element wordt gesloten. Met andere woorden:

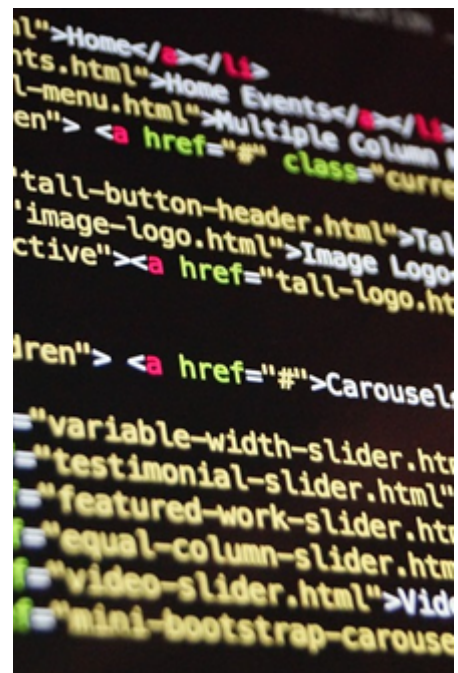
de tag `</html>` mag in de code van onze sjabloon nooit voor de tag `</head>` geplaatst worden. Zie de afbeelding.



```
<!DOCTYPE html>
<html>
  <head>
    <title>nesten</title>
    <meta charset="utf-8">
    <link rel="stylesheet" href="opmaak.css">
  </head>
  <body>
    ...
  </body>
</html>
```

★ Aan de slag 4

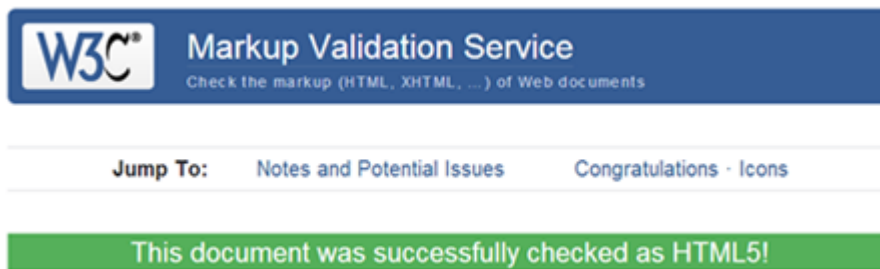
Schrijf op wat wordt verstaan onder het nesten van HTML-code.



Controleren van code

In 1994 werd in de Verenigde Staten het World Wide Web Consortium (W3C) opgericht. Dit is een organisatie die webstandaarden voor het World Wide Web ontwerpt. HTML is zo'n webstandaard. De organisatie staat onder leiding van Tim Berners-Lee.

De W3C-groep heeft een tool ontwikkeld waarmee gecontroleerd kan worden of er geen fouten in de code van een HTML-pagina zitten. Die tool wordt de W3C Markup Validation Service genoemd. Het controleren van de code op fouten wordt valideren genoemd. Valideren kan via het webadres validator.w3.org. We zouden dus kunnen controleren of de code van onze sjabloon voldoet aan de eisen voor een HTML5-document. Als dat het geval is, zal na het controleren van de code een melding verschijnen dat het document een geldig HTML5-document is.



Bekijk het volgende filmpje.
Lees daarna eventueel de tekst van de video.



https://maken.wikiwijs.nl/bestanden/555656/validator_v2.mp4

Open een browser en typ in de adresbalk validator.w3.org in. Klik op het tabblad "Validate by File Upload". Blader nu naar de plek waar je de template hebt opgeslagen en selecteer dit bestand. Klik vervolgens op "Check". De validator controleert nu of we een geldig HTML5-document hebben geschreven.

★ Aan de slag 5

- Klik op de volgende link: validator.w3.org.
Klik op het tabblad "Validate by File Upload".
Zoek in de homedirectory het bestand **template.html** op en upload dit bestand.
Klik vervolgens op "Check". De validator controleert nu of we een geldig HTML5-document hebben geschreven.
- Open in Notepad++ het bestand **template.html**.
Verwijder de tags `<title>` en `</head>`. Sla het bestand daarna op onder de naam `template_fout.html`. Controleer de broncode van het bestand **template_fout.html** in de validator.
Schrijf op hoeveel en wat voor soort foutmelding(en) je krijgt te zien.
- De validator biedt nog twee andere mogelijkheden aan om te controleren of er fouten in je code zitten. Welke?



Elementen en attributen

In het stukje over het sjabloon hebben we uitgelegd wat tags zijn en waaraan je tags kunt herkennen. Een tag is eigenlijk niets anders dan de naam van wat in HTML elementen worden genoemd. Elementen zijn de bouwstenen van HTML-pagina's. In onze sjabloon staan bijvoorbeeld de elementen `body` en `head`.



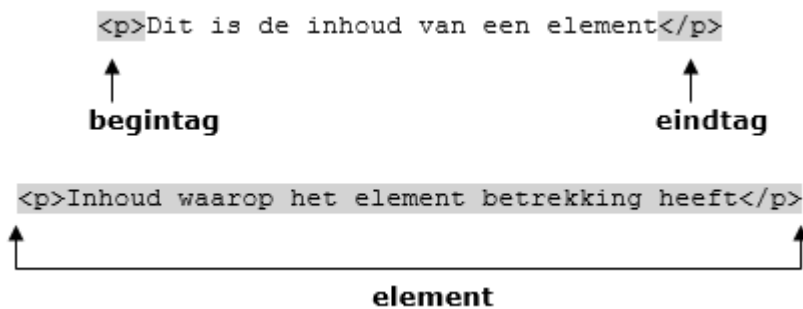
De meeste elementen bestaan uit een begintag en een eindtag inclusief de inhoud waarop het element betrekking heeft. De tags geven het begin en het einde van een element aan.



Soms wordt er geen eindtag gebruikt. In die gevallen hebben we te maken met elementen zonder inhoud. Dat worden ook wel lege elementen genoemd.

Voorbeelden van elementen zonder inhoud zijn `
` en ``. Lege elementen hebben slechts één tag, namelijk een begintag. Voor het opnemen van een afbeelding in een webpagina volstaat dus de regel ``.

In de afbeelding zie je een tag voor tekst met inhoud.



In XHTML, een eerdere versie van HTML, is het verplicht om de tags van elementen zonder inhoud netjes af te sluiten. Dat gebeurt in XHTML door in de begintag voor het >-teken een / te plaatsen. Het element `
` wordt dan genoteerd als `
`.

In HTML5 is het afsluiten van lege elementen niet nodig. In HTML5 mag je dus gewoon `
` noteren.

Een begintag van een element kan worden aangevuld met één of meerdere attributen. Het komt regelmatig voor dat een begintag helemaal geen attributen heeft. Attributen beïnvloeden de tag. Een attribuut heeft altijd een naam en een waarde. De waarde van een attribuut wordt tussen dubbele aanhalingstekens genoteerd. Tussen de naam en de waarde van een attribuut mag je geen spaties gebruiken.

In het onderstaande voorbeeld wordt bij de tag `` het attribuut `src` (source) vermeld. Dit attribuut heeft als waarde een verwijzing naar de plek waar de afbeelding te vinden is.

```

```

Bekijk het volgende filmpje.
Lees daarna eventueel de tekst van het filmpje.



https://maken.wikiwijs.nl/bestanden/555664/IDenClass_v2.mp4

Tekst id en class filmpje

Speciale attributen die je aan een element kunt geven zijn het attribuut `id` en het attribuut `class`. Door een `id` of `class` een naam te geven, kun je in de CSS-code beschrijven hoe de opmaak van een specifiek element eruit moet komen te zien. Kijk maar eens naar de onderstaande code:

```
<p id='naam'>Inhoud waarop het element betrekking heeft</p>
```

In dit voorbeeld heeft het HTML-element `p` een attribuut `id` gekregen met als naam "naam". In het stijlblad kunnen we opmaak van deze alinea verder beschrijven. Ergens in het CSS-bestand zal dan staan:

```
#naam
```



```
{  
...  
}
```

De opmaakcode is alleen van toepassing op het element p met de id "naam" en dus niet op alle andere p-elementen.

We kunnen ook kiezen voor het attribuut class. Dan noteren we:

```
<p class='naam'>Inhoud waarop het element betrekking heeft</p>
```

En in het CSS-bestand zal dan ergens staan:

```
.naam  
{  
...  
}
```

Met het attribuut id, de afkorting van "identification", geef je een element een unieke naam. Deze naam mag op een webpagina slechts één keer voorkomen. Dat geldt niet voor het attribuut class. Een class kun je aan verschillende elementen toewijzen. In bovenstaand voorbeeld hebben we `<p class='naam'> ... </p>` gebruikt. In de broncode van de pagina zou ook nog ergens mogen staan `<h2 class="naam"> ... </h2>` Als we de alinea van een id voorzien, mogen we de naam van deze id niet voor een ander element gebruiken. In de praktijk blijkt dat heel vaak dezelfde namen voor een class of id worden gebruikt. Dat geldt bijvoorbeeld voor onderdelen die gebruikt worden om de structuur van een webpagina te beschrijven. In HTML5 zijn daarom een reeks nieuwe structuurelementen geïntroduceerd.

★ Aan de slag 6

- Geef voor de onderstaande HTML-code aan wat het element, de begintag, de eindtag is en welke naam en waarde het attribuut heeft:

```
<p class="naam">Inhoud waarop het element betrekking heeft</p>
```

- Schrijf in je eigen woorden op wat lege elementen zijn.
- Zoek op met welke tag je naar de volgende regel springt.
- Zoek op internet op welke functie de tag `<hr>` in HTML5 heeft.



Kopteksten

De inhoud van een webpagina bestaat vaak uit teksten. Die

teksten moeten duidelijk en overzichtelijk zijn zodat de

lezer geen moeite heeft om te begrijpen waar een tekst over gaat. Voor het aanbrengen van structuur in een tekst worden kopteksten en alinea's

gebruikt. In HTML5 bestaan er

kopteksten voor 6 verschillende niveaus. De grootste kop is de kop van niveau 1. Voor deze koptekst wordt de tag `<h1>` gebruikt. Voor de kleinste koptekst wordt de tag `<h6>` gebruikt. Het h-element bestaat altijd uit een begin- en een eindtag.

Dit is een koptekst van niveau 1

Dit is een koptekst van niveau 2

Dit is een koptekst van niveau 3

Dit is een koptekst van niveau 4

Dit is een koptekst van niveau 5

Dit is een koptekst van niveau 6

Voor het bekijken van het resultaat van HTML-code hebben we tot nu toe een browser gebruikt. In deze module bestaat ook de mogelijkheid om de uitvoer van HTML-code te tonen met behulp van een speciaal ontwikkelde tool.

Je hebt dan geen browser meer nodig.

Voor sommige opdrachten maken we gebruik van deze omgeving. Aan de linkerkant staat de HTML-code. Aan de rechterkant is na het klikken op verwerken het resultaat van de code te zien.

★ Aan de slag 7

- Als je de HTML5 tool opent zie je aan de linkerkant de code van **template.html**. Schrijf de code voor de body van het HTML-document zodat bij het bekijken van de pagina hetzelfde te zien is als op de afbeelding hiernaast. Geef het document de titel "kopteksten". Bekijk het resultaat door te klikken op uitvoeren



- Aangezien we dezelfde broncode nog een aantal keren gaan gebruiken, gaan we de code opslaan in een apart bestand. Open Notepad++ en neem de code voor de kopteksten over. Sla het bestand op in de map html onder de naam **kopteksten.html**. Bekijk het resultaat in een browser.



B2 HTML5

Alinea's

Bekijk nu het filmpje over het gebruik van alinea's:



<https://maken.wikiwijs.nl/bestanden/555668/Alinea.mp4>

Op de website van Apple staat een tekstfragment met informatie over de Apple ID:

Een Apple ID configureren voor iCloud en iTunes

Een Apple ID is het e-mailadres dat u als inlognaam gebruikt voor vrijwel alle Apple-gerelateerde activiteiten, zoals het gebruik van iCloud voor het bewaren van materiaal, de aankoop van nummers in de iTunes Store en de download van apps vanaf de App Store.

Wanneer u een iPhone, iPad of iPod touch configureert, kunt u dezelfde Apple ID gebruiken voor iCloud-voorzieningen en aankopen in de iTunes Store, App Store en iBooks Store. U kunt ook een Apple ID voor iCloud-voorzieningen gebruiken en een andere Apple ID voor aankopen in Stores (inclusief iTunes in de cloud en iTunes Match). U geniet van alle voordelen van iCloud, ongeacht of u nu dezelfde Apple ID of verschillende Apple ID's voor iCloud en aankopen in Stores gebruikt. Als u de Apple ID wilt wijzigen waarmee u bij iCloud of iTunes op een iPhone, iPad of iPod touch bent ingelogd, logt u uit en gebruikt u de onderstaande stappen om opnieuw in te loggen met de bijgewerkte Apple ID.



Het fragment heeft als koptekst "Apple ID configureren voor iCloud en iTunes" en bestaat uit 2 alinea's. In de eerste alinea wordt uitgelegd wat een Apple ID is. In de tweede alinea wordt uitgelegd waar een Apple ID voor gebruikt kan worden. In HTML geef je dat aan als:

```
<h2>Een Apple ID configureren voor iCloud en iTunes</h2>
```

```
<p>
```

Een Apple ID is het e-mailadres dat u als inlognaam gebruikt voor vrijwel alle Apple-gerelateerde activiteiten, zoals het gebruik van iCloud voor het bewaren van materiaal, de aankoop van nummers in de iTunes Store en de download van apps vanaf de App Store.

```
</p>
```

```
<p>
```

Wanneer u een iPhone, iPad of iPod touch configureert, kunt u dezelfde Apple ID gebruiken voor iCloud-voorzieningen en aankopen in de iTunes Store, App Store en iBooks Store. U kunt ook een Apple ID voor iCloud-voorzieningen gebruiken en een andere Apple ID voor aankopen in Stores (inclusief iTunes in de cloud en iTunes Match). U geniet van alle voordelen van iCloud, ongeacht of u nu dezelfde Apple ID of verschillende Apple ID's voor iCloud en aankopen in Stores gebruikt. Als u de Apple ID wilt wijzigen waarmee u bij iCloud of iTunes op een iPhone, iPad of iPod touch bent ingelogd, logt u uit en gebruikt u de onderstaande stappen om opnieuw in te loggen met de

bijgewerkte Apple ID.
</p>

Soms wil je dat in een alinea naar een volgende regel wordt gegaan. Daarvoor gebruiken we het element break. In HTML5 wordt dat genoteerd als
. Een andere keer wil je een woord of een stukje tekst schuingedrukt of vetgedrukt weergeven. Daarvoor worden de tags ... en ... gebruikt. De naam van het element em is afgeleid van emphasize, het Engelse woord voor benadrukken.

Deze regel is schuingedrukt
Deze regel is schuingedrukt

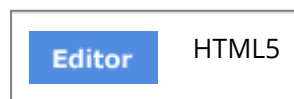
Deze regel is vetgedrukt
Deze regel is vetgedrukt

★ Aan de slag 8

Pas in de HTML-tool de reeds aanwezige code van de template aan. Maak een visitekaartje waarin je NAW-gegevens (Naam Adres Woonplaats) komen te staan plus je mailadres en je telefoonnummer. De titel van de pagina is "Visitekaartje" en voor de naam wordt een kop-tag gebruikt.

Hiernaast zie je hoe de pagina er in een browser uit zou moeten zien.

Bekijk het resultaat door op uitvoeren te klikken



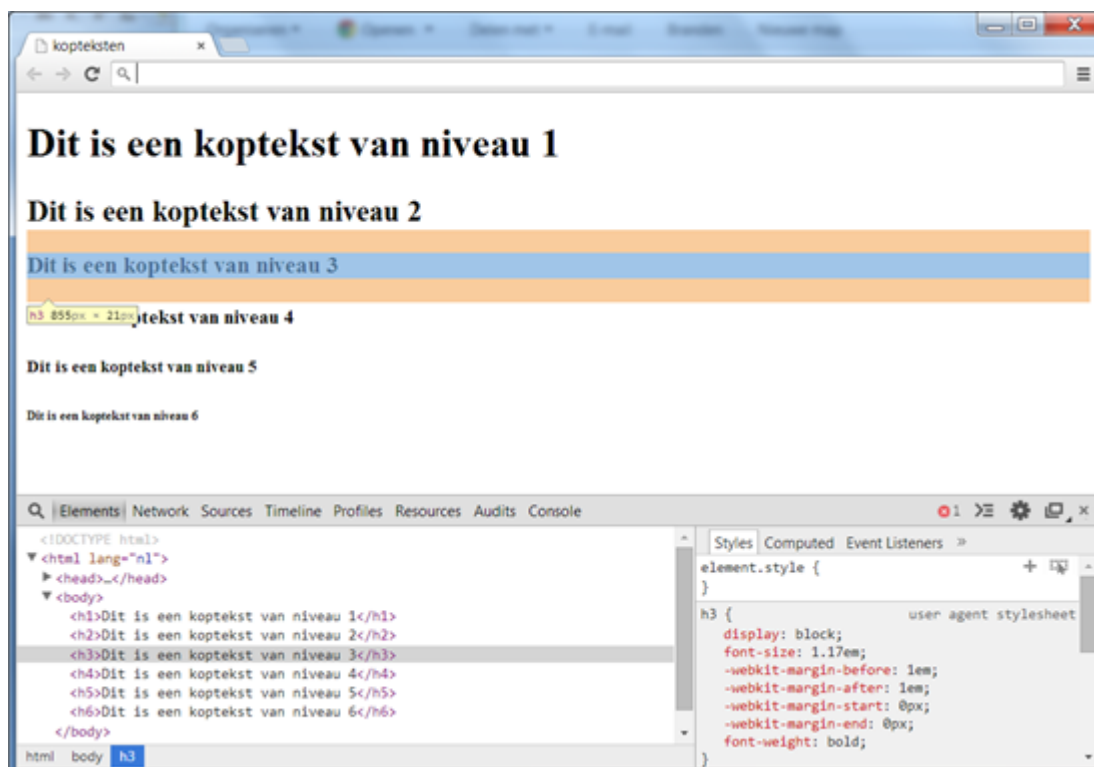
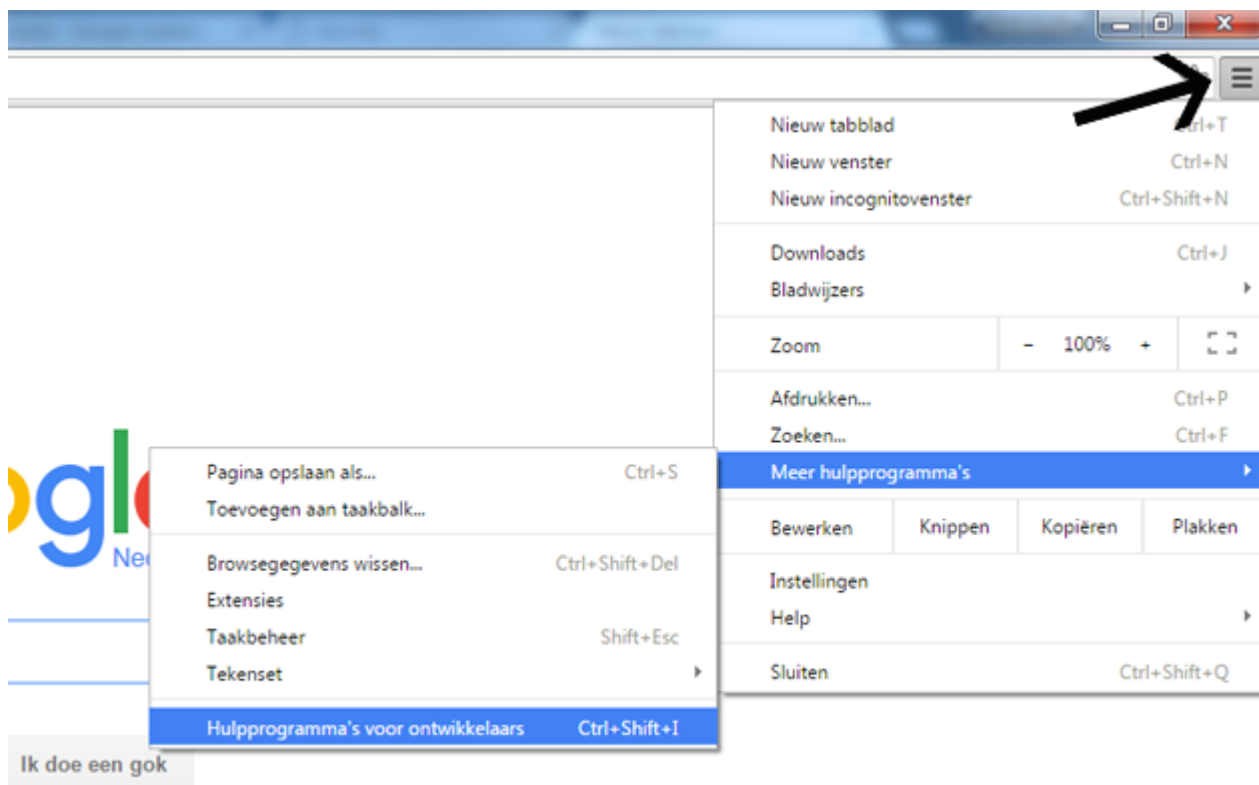
★ Aan de slag 9

In de browser Chrome bestaat een hulpprogramma voor ontwikkelaars dat erg handig kan zijn bij het bekijken van de structuur en opmaak van een webpagina.

Open in de browser Chrome het bestand kopteksten.html. Dit is de webpagina met de verschillende kopteksten. Klik in Chrome op het icoontje met de streepjes en kies voor meer Hulpprogramma's voor ontwikkelaars.

Zorg ervoor dat in het venster onderaan het tabblad Elements is geselecteerd. Klik in de HTML-code op de regel "<h3>Dit is een koptekst van niveau 3</h3>". Rechts is nu de opmaak te zien die bij dit element standaard gebruikt wordt. In de webpagina is het element h3 geselecteerd.

Voor extra uitleg kijk de afbeeldingen hieronder:



Afbeeldingen

In webpagina's kunnen afbeeldingen worden opgenomen. Een afbeelding is in de broncode van een HTML-document niets anders dan een verwijzing naar waar een afbeelding te vinden is. Dat is te zien in de onderstaande regel waarin verwezen wordt naar de afbeelding "Enigma.jpg" in de map afb:

```

```

Aan het element `img` worden twee attributen toegekend. Het attribuut `src` (source) verwijst naar de plek waar de afbeelding te vinden is. Het attribuut `alt` (alternative) toont een alternatieve tekst die verschijnt wanneer de afbeelding om wat voor reden dan ook niet kan worden weergegeven. Het `alt` attribuut wordt niet door elke browser uitgevoerd.

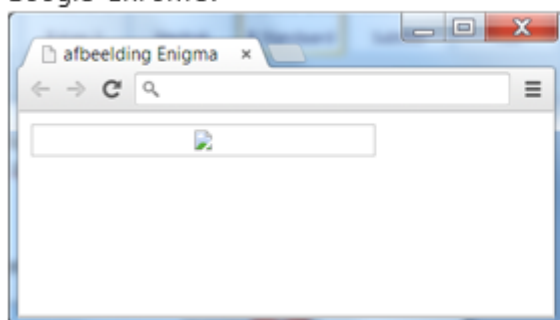
In Google Chrome zien we geen tekst als de afbeelding niet geladen kan worden.

In Internet Explorer weer wel.

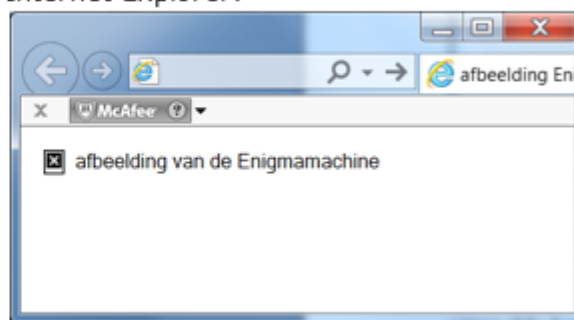
Bekijk de afbeelding.



Google Chrome:



Internet Explorer:



Als een afbeelding niet geladen wordt, kunnen daar diverse verklaringen voor zijn. Mogelijk klopt de verwijzing naar het bestand niet. Daar is bijvoorbeeld sprake van als het bestand "Enigma.jpg" niet in de map afb staat, maar op dezelfde plek als het HTML-document.

Een andere reden kan zijn dat het in de code vermelde bestandsformaat niet overeenkomt met het werkelijke bestandsformaat. Wij gebruiken een afbeelding van het formaat jpg. Als de afbeelding van de Enigma-machine het formaat png of gif zou hebben, zouden we na het laden van de pagina de afbeelding niet te zien krijgen.

In HTML5 zijn 2 nieuwe elementen toegevoegd waarmee een afbeelding in een document kan worden gemarkeerd en van een bijschrift worden voorzien. Deze elementen worden `figure` en `figcaption` genoemd. Als we de afbeelding van de Enigmamachine van een bijschrift zouden willen voorzien, gebruiken we daar de tag `<figcaption>` voor.

```
<figure>

<figcaption>De Enigma</figcaption>
</figure>
```

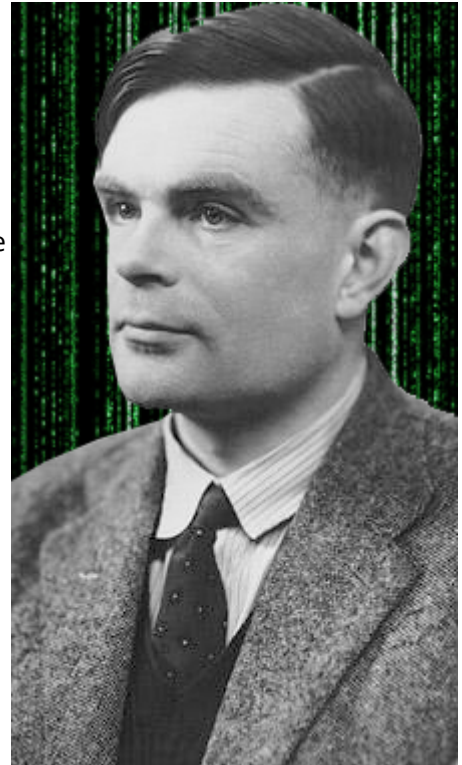
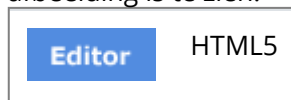
Het is aan te bevelen om de afbeeldingen die voor een website worden gebruikt, op te slaan in een aparte map. Deze map zou dan afb kunnen worden genoemd. Voor opmaakbestanden zou de map `css` gebruikt kunnen worden. JavaScript-bestanden worden vaak opgeslagen in de map `js`.

Het is niet gebruikelijk om de HTML-bestanden in een submap op te slaan.

Deze bestanden staan gewoon in de hoofdmap. Door bepaalde typen bestanden op te slaan in aparte mappen behoud je overzicht en zijn bestanden sneller terug te vinden.

★ Aan de slag 10

- a. Open een browser en zoek op internet naar een foto van Alan Turing.
Sla de afbeelding op in de map html in je home-directory. Open onze HTML-editor. In de editor is het mogelijk om bestanden te uploaden. Daar kan in een HTML-document naar verwezen worden.
In het venster rechts boven met de titel Bestanden staan de knoppen selecteer en uploaden. Klik op selecteer, ga naar de plek waar de afbeelding staat. Klik op de naam van de afbeelding en kies tot slot uploaden. Als alles naar wens verloopt, krijg je een melding dat het bestand succesvol is geupload. De naam van het bestand staat daarna in het venster.
Klik nu op de knop "Laad HTML" en open het bestand template in de map html. Vul de code van het HTML-bestand zo aan dat na het verwerken van de code de afbeelding is te zien.



- b. Voorzie de foto van Alan Turing van het onderschrift "Alan Turing (1912-1954)". Pas daarvoor de HTML-code aan. Klik daarna op verwerk en bekijk het resultaat.
- c. Maak nu in de map html in je homedirectory een submap afb aan. Verplaats na deze opdracht de foto van Alan Turing naar de map afb. In deze map moeten voortaan alle afbeeldingen worden opgeslagen. In de editor is het namelijk niet mogelijk te verwijzen naar bepaalde mappen waar afbeeldingen zijn opgeslagen. Dat verklaart waarom we de afbeelding in eerste instantie op dezelfde plek als het HTML-bestand hebben opgeslagen.
Werk je met een browser en Notepad++ dan zijn verwijzingen naar specifieke mappen wel mogelijk. Vergeet dan niet dat voorafgaand aan de naam van de afbeelding de naam van de map waar deze afbeelding in staat genoemd moet worden. In de HTML-code komt dan bijvoorbeeld te staan:
`img src="https://maken.wikiwijs.nl/afb/Alan_Turing.jpg"`

Verwijzingen

Behalve afbeeldingen zien we in webpagina's ook vaak verwijzingen staan.

Een verwijzing wordt ook wel een hyperlink genoemd. Er kan zowel van tekst als van een afbeelding een hyperlink worden gemaakt. Voor het maken van een hyperlink wordt het element `a` in combinatie met het attribuut `href` gebruikt. In het `href`-attribuut geef je aan waar naar toe moet worden verwezen.

Onderstaand stukje code, dat je kunt zien in de popup, zorgt er bijvoorbeeld voor dat de tekst "klik hier" een verwijzing wordt naar de pagina `https://maken.wikiwijs.nl/afbeelding.html`. Dat is de naam van de webpagina waarop de afbeelding van de Enigma-machine is te zien.

```
<body>
<h3>Een verwijzing </h3>
<p>
<a href="https://maken.wikiwijs.nl/afbeelding.html">Klik hier</a> om naar de pagina met de afbeelding
van de Enigma-machine te gaan.
</p>
</body>
```



★ Aan de slag 11

Open in Notepad++ het bestand onderschrift.html.
Dit bestand is onderdeel van het zip-bestand dat je al eerder hebt moeten downloaden en hebt uitgepakt in je home-directory in de map html.
Maak van de foto van Alan Turing een link die verwijst naar en.wikipedia.org/wiki/Alan_Turing.
Sla het bestand op als verwijzing.html en bekijk het resultaat in een browser.



★ Aan de slag 12

Open Notepad++.
Maak 2 aparte webpagina's en noem deze pagina's [pagina_1.html](#) en [pagina_2.html](#).
Plaats op [pagina_1.html](#) een link die verwijst naar [pagina_2.html](#) en vanaf [pagina_2.html](#) een link die verwijst naar [pagina_1.html](#).
Bekijk en test het resultaat in een browser.



Lijsten

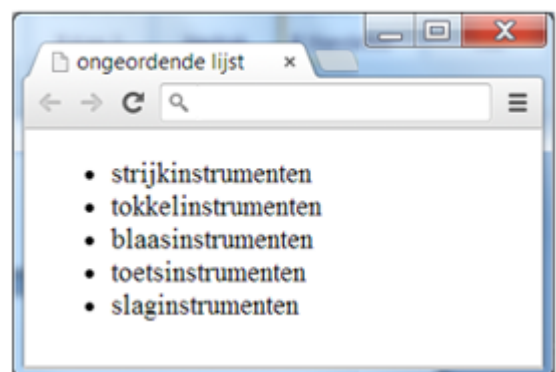
Lijsten worden gebruikt om opsommingen te maken. Er bestaan 2 soorten lijsten in HTML:

1. De ongeordende lijst (unordered list).
2. De geordende lijst (ordered list).

We kijken eerst naar een ongeordende lijst. De lijst bestaat uit muziekinstrumenten die zijn ingedeeld op basis van de manier waarop ze bespeeld worden:

- strijkinstrumenten
- tokkelinstrumenten
- blaasinstrumenten
- toetsinstrumenten
- slaginstrumenten

Het resultaat in een browser

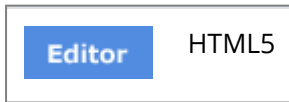


De HTML-code voor deze ongeordende lijst wordt:

```
<ul>
<li>strijkinstrumenten</li>
<li>tokkelinstrumenten</li>
<li>blaasinstrumenten</li>
<li>toetsinstrumenten</li>
<li>slaginstrumenten </li>
</ul>
```

★ Aan de slag 13

Neem de HTML-code voor de ongeordende lijst over in de HTML-tool. Geef het document de titel "ongeordende lijst". Plaats de code in de body van het HTML-document en kijk naar het resultaat.



Voor de ongeordende lijst wordt het element `ul` gebruikt (` ... `).

Elk element van een lijst, dat geldt zowel voor de ongeordende als de geordende lijsten, staat altijd tussen de tags `` en ``. De naam `li` is afgeleid van list item.

We zouden van de ongeordende lijst ook een geordende lijst kunnen maken. Daarvoor moeten we in de broncode van het HTML-document de begintag `` en de eindtag `` vervangen door `` en ``.

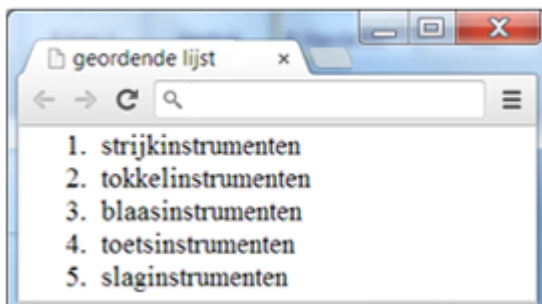
De namen `ul` en `ol` zijn, maar dat had je waarschijnlijk al in de gaten, afgeleid van `unordered list` en `ordered list`.

Hieronder zie je de HTML-code voor de geordende lijst van muziekinstrumenten.

Het resultaat in een browser zie je op de afbeelding hieronder.

```
<ol>
<li>strijkinstrumenten</li>
<li>tokkelinstrumenten</li>
<li>blaasinstrumenten</li>
<li>toetsinstrumenten</li>
<li>slaginstrumenten </li>
</ol>
```

Het resultaat in een browser



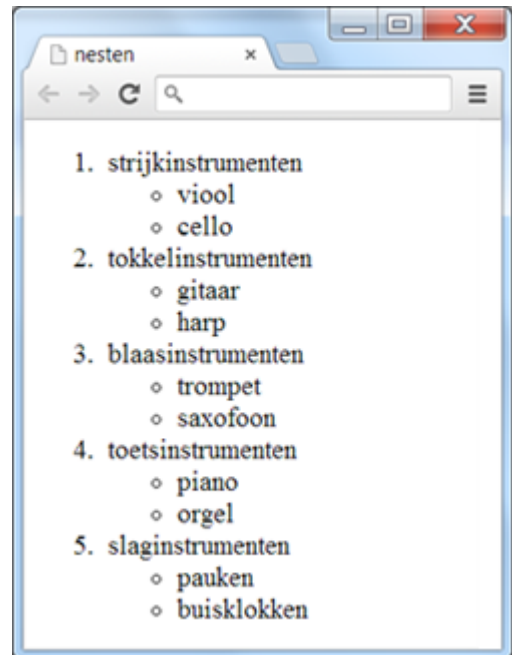
★ Aan de slag 14

Neem de HTML-code voor de geordende lijst over in de HTML-tool. Dit kan je doen door de code van de vorige pagina te kopiëren en te plakken in de Editor. Geef het document de titel "geordende lijst". Plaats de code in de body van het HTML-document en kijk naar het resultaat.



Het is ook mogelijk om lijsten te nesten. Hiernaast zie je nogmaals de door ons gehanteerde indeling van muziekinstrumenten, maar nu aangevuld met enkele voorbeelden:

```
<ol>
<li>strijkinstrumenten
<ul>
<li>viool</li>
<li>cello</li>
</ul>
</li>
<li>tokkelinstrumenten
<ul>
<li>gitaar</li>
<li>harp</li>
</ul>
</li>
<li>blaasinstrumenten
<ul>
<li>trompet</li>
<li>saxofoon</li>
</ul>
</li>
<li>toetsinstrumenten
<ul>
<li>piano</li>
<li>orgel</li>
</ul>
</li>
<li>slaginstrumenten
<ul>
<li>pauken</li>
<li>buisklokken</li>
</ul>
</li>
</ol>
```



In dit voorbeeld zijn ongeordende lijsten genest in een geordende lijst. Er zijn daarop tal van andere variaties te bedenken. Misschien is het je opgevallen dat de rondjes van de ongeordende lijsten nu ineens niet meer dicht, maar open zijn.

De opsommingstekens voor een ongeordende opsomming in een andere opsomming zijn standaard open rondjes. We noemen dit het tweede niveau van een opsomming. Voor het eerste niveau worden standaard de gesloten rondjes gebruikt.

Het is mogelijk om in de HTML-editor ook bestaande HTML-documenten te laden. Voor sommige opdrachten gaan we hier gebruik van maken.

★ Aan de slag 15

Open in de HTML-editor het bestand **template.html** door op de knop "Laad HTML" te klikken en in de verkenner die nu geopend wordt dit bestand te selecteren in de map html in je homedirectory. Schrijf in de body van het document de HTML-code voor een lijstje van jouw 10 favoriete liedjes. Klik daarna in de editor op de knop "Uitvoeren" en bekijk het resultaat.

Editor

HTML5

Tabellen

De data die we hebben gebruikt voor het maken van lijsten zouden we ook in een tabel kunnen zetten. Dat hebben we hieronder gedaan:



strijkinstrumenten	tokkelinstrumenten	blaasinstrumenten	toetsinstrumenten	slaginstrumenten
viool	gitaar	trompet	piano	pauken
cello	harp	saxofoon	orgel	buisklokken

Een tabel bestaat uit kolommen, rijen en cellen. Vaak staan boven in de tabel de kopjes van de kolommen. Voor het maken van een tabel worden diverse tags gebruikt. We zetten de belangrijkste tags even voor je op een rijtje:

<table>	Deze tag wordt gebruikt voor het definiëren van een tabel.
<th>	Met de tag <th> worden de kopjes van de kolommen aangeven. De afkorting staat voor table heading.
<tr>	Met de tag <tr> worden tabellen onderverdeeld in rijen. De afkorting staat voor table row.
<td>	Met de tag <td> worden rijen onderverdeeld in cellen. De afkorting staat voor table data

De tabel is voorzien van een rand. In HTML5 regel je dat bij voorkeur in het CSS-bestand en niet in de HTML-code. Dat doe je in CSS omdat een rand onderdeel is van de opmaak van een tabel. Een tabel

heeft standaard geen rand.
Wil je dat de tabel een rand krijgt, moet je dat dus altijd specifiek aangeven.

Boven de tabel kunnen we een titel plaatsen. Daarvoor gebruiken we het element caption. De tag <caption> moet direct na de tag <table> ingevoegd worden.
We geven de tabel de titel "Indeling muziekinstrumenten".
De code hiervoor wordt dan:

```
<caption>Indeling muziekinstrumenten</caption>
```

Standaard wordt de titel gecentreerd.

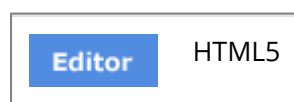
★ Aan de slag 16

Open in de HTML-editor het bestand [template.html](#). Schrijf in de body van het document de HTML-code voor de afgebeelde tabel van de diverse muziekinstrumenten.



strijkinstrumenten	tokkelinstrumenten	blaasinstrumenten	toetsinstrumenten	slaginstrumenten
viool	gitaar	trompet	piano	pauken
cello	harp	saxofoon	orgel	buis klokken

Klik daarna in de editor op de knop "Uitvoeren" en bekijk het resultaat. Voor de rand van de tabel mag je nu nog het attribuut border="1" gebruiken aangezien je nu nog niet weet hoe je dat in CSS moet aangeven. Later regel je dat dus in CSS.



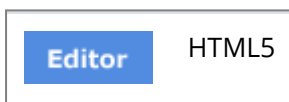
★ Aan de slag 17

Ga na waarvoor in de HTML-code voor tabellen de attributen `colspan` en `rowspan` worden gebruikt.

Je kunt hiervoor gebruikmaken van [deze webpagina](#).

★ Aan de slag 18

Open in de HTML-editor het bestand **template.html**. Probeer om de tabel die hier rechts staat na te bouwen in de editor. Bekijk na het schrijven van de code het resultaat in de editor.



Indeling muziekinstrumenten	
snaarinstrumenten	viool
	cello
tokkelinstrumenten	gitaar
	harp
blaasinstrumenten	trompet
	saxofoon
toetsinstrumenten	piano
	orgel
slaginstrumenten	pauken
	buisklokken



Video en audio

Moderne webpagina's maken vaak gebruik van beeld en/of geluid. In HTML5 zijn speciaal daarvoor 2 nieuwe elementen bedacht. Voor beeld is dat het element `video`. Voor geluid is dat het element `audio`.

Video

De HTML-code voor het gebruik van een film of video op een webpagina is niet heel erg ingewikkeld.

```
<video src="https://maken.wikiwijs.nl/film.mp4"></video>
```

Het element `video` heeft een begintag en een eindtag. De inhoud van het element wordt getoond in het geval de browser het element `video` niet kent. Het is daarom handig om tussen de begin- en de eindtag een mededeling te plaatsen. Het element `video` werkt overigens in alle moderne browsers.

```
<video src="https://maken.wikiwijs.nl/film.mp4"> ondersteunt het element video niet</video>
```

Aan het element `video` kunnen diverse attributen worden toegevoegd:

- `src`: verwijst naar het bestand.
- `width` en `height`: geven de breedte en hoogte van het afspelenvenster aan.
- `controls`: toont de bedieningsknoppen.
- `autoplay`: laadt en speelt de video direct af.
- `preload`: laadt de video direct met de pagina. De video wordt echter niet direct afgespeeld.

In het voorbeeld heeft de film en/of video de extensie `mp4`. Andere extensies die voor beeld gebruikt kunnen worden, zijn `mov`, `flv` of `avi`. Een extensie zegt normaal iets over het bestandstype dat gebruikt wordt. Bij een film ligt dat echter iets gecompliceerder. Een film is meestal opgebouwd uit meerdere tracks. De extensie van een filmbestand zegt dus niet iets over één bestand maar eigenlijk over een



verzameling van tracks.

De extensie van een filmbestand kan vergeleken worden met wat ook wel een container wordt genoemd. Het fungeert als een omhulsel voor één of meerdere tracks. Het probleem dat zich nu voordoet, is dat deze tracks van verschillende typen kunnen zijn.

Er bestaan vaste afspraken voor het coderen van tracks. Als een film wordt afgespeeld, moeten de tracks ook weer gedecodeerd worden. De gemaakte afspraken voor het coderen en decoderen worden codecs genoemd. Er zijn heel veel verschillende soorten codecs in omloop, maar drie codecs worden het meest gebruikt. Dat zijn H.264, Theora en VP8. H.264 voor de extensie mp4. Theora voor ogv-bestanden en VP8 voor bestanden met de extensie webm.

Als je er zeker van wilt zijn dat je film de meeste kans heeft om te worden afgespeeld, moet het filmbestand met alle drie de vermelde codes worden gedecodeerd. Daarvoor is een oplossing bedacht. Kijk maar eens naar de onderstaande HTML-code:

```
<video width="640" height="480" controls preload="none">
<source src="https://maken.wikiwijs.nl/film.mp4" type="video/mp4">
<source src="https://maken.wikiwijs.nl/film.ogv" type="video/ogg">
<source src="https://maken.wikiwijs.nl/film.webm" type="video/webm">
De gebruikte browser ondersteunt het element video niet.
</video>
```

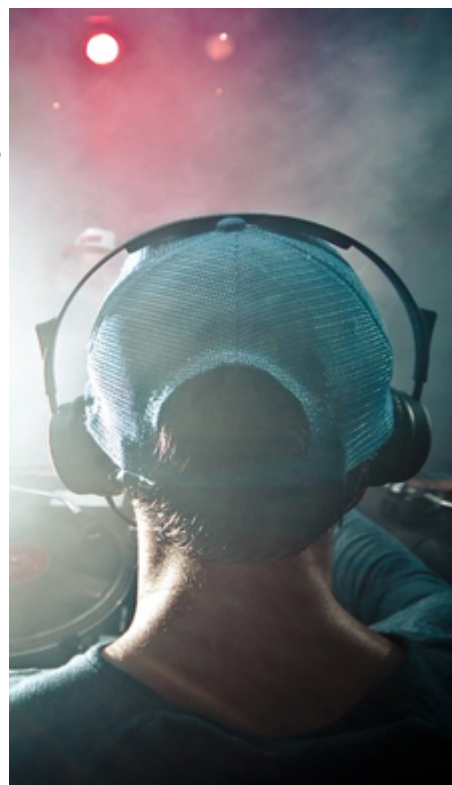
Aan het element video worden meerdere elementen source toegekend. Elk element source bevat de attributen src en type. Het komt er dus op neer dat het element video meerdere bronnen bevat.

Bij het verwerken van het element video gaat de browser een lijstje af. Kan de browser de eerste bron niet afspelen, kijkt de browser vervolgens naar de tweede en de derde bron. Mislukt het decoderen alsnog dan wordt de tekst "De gebruikte browser ondersteunt het element video niet" afgebeeld.

Audio

We hebben gezien dat het afspelen van een filmfragment nog niet zo eenvoudig is. Het afspelen van een geluidsfragment is gelukkig iets eenvoudiger. Toch lijkt het gebruik van het element audio wel op het gebruik van het element video. Voor audio worden vooral de bestandstypen mp3 en ogg gebruikt. De belangrijkste attributen voor audio zijn src en controls.

```
<audio controls preload="none">
<source src="https://maken.wikiwijs.nl/geluid.mp3" type="audio/
mp3">
<source src="https://maken.wikiwijs.nl/geluid.ogg" type="audio/
ogg">
De gebruikte browser ondersteunt het element audio niet.
</audio>
```



★ Aan de slag 19

Open in Notepad++ het bestand [template.html](#).
Voeg aan de body van dit document de volgende regel toe:

```
<video src="http://www.youtube.com/watch?v=gtRLmL70TH0"
controls autoplay></video>
```

Verander de titel van het HTML-document in video.
Sla het bestand op onder de naam [video.html](#).
We hebben nu een verwijzing naar een videofragment op YouTube over Alan Turing in de HTML-code opgenomen.
Bekijk het resultaat in Internet Explorer en daarna ook nog in een andere browser.
Welke conclusie kun je trekken?



Structuur

Bekijk de animatie:

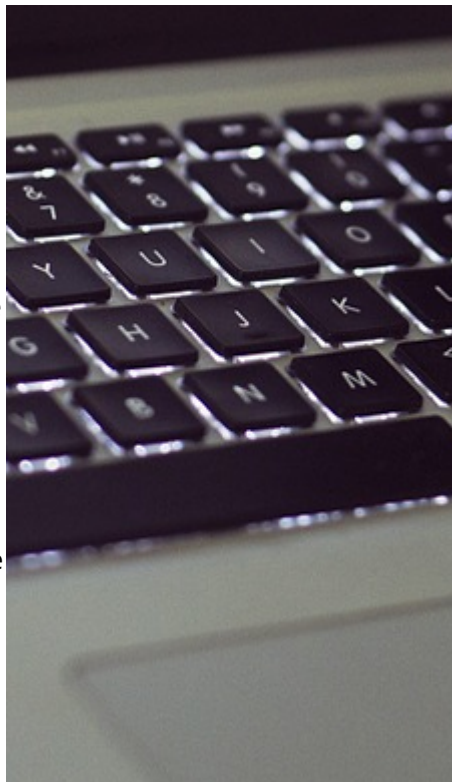
- [Structuurelementen](#)

De plaats waar de structuurelementen geplaatst worden staat niet vast. Web developers kunnen de elementen plaatsen waar zij zelf willen om de website die ze bouwen zo mooi of functioneel mogelijk te maken.

Stel we maken gebruik van het element header. In de header van de webpagina nemen we een afbeelding van een logo op. In het body-gedeelte van de webpagina staat dan:

```
<header>

</header>
```



In het stijlblad kunnen we opmaak van de header verder beschrijven. Ergens in het CSS-bestand zal dan staan:

```
header
{
}
}
```

Tussen de accolades geven we in het CSS-bestand aan hoe de lay-out van de header van de webpagina eruit moet komen te zien. Bijvoorbeeld welke achtergrondkleur we gebruiken.

```
header
```



```
{  
background-color: blue;  
}
```

Voor het aanbrengen van structuur in HTML-documenten werd in het verleden vaak gebruik gemaakt van het div-element. Div is de afkorting van division. Alles wat in de broncode wordt geplaatst tussen de begintag <div> en de eindtag </div> wordt als een blok bij elkaar gehouden.

Het is niet verboden om ook in HTML5 gebruik te maken van het div-element. In sommige gevallen kan dat niet anders. Probeer als het kan de nieuwe structuurelementen te gebruiken.

★ Aan de slag 20

- Open in Notepad++ het bestand **template.html** Schrijf de HTML-code voor een pagina met een header, een nav, een section en een footer. Sla het bestand op in je home-directory in de map html onder de naam **structuur.html**. Bekijk het resultaat in een browser.
- Open in Notepad++ het bestand **structuur.html** Voeg binnen het element section 2 articles toe. Elk article bestaat uit een koptekst van niveau 2 en een alinea. Gebruik voor de kopteksten in HTML de tekst "Dit is een koptekst". Voor de beide alinea's kan de tekst "Dit is een alinea" gebruikt worden. Sla het bestand op onder de naam **section.html** en bekijk het resultaat in een browser



Toets 1

In de volgende toets wordt getoetst wat je inmiddels zou moeten weten over het onderwerp HTML5.

Maak de toets:

Adaptieve Toets

HTML5



C CSS3

Inleiding

In de inleiding van deze module heb je kunnen lezen dat je voor de bouw van een website altijd te maken hebt met 3 aspecten. Namelijk de structuur, de opmaak en de interactiviteit. Voor de structuur van een webpagina wordt HTML gebruikt.

Dit aspect van een website is inmiddels behandeld. In dit onderdeel gaan we in op hoe de opmaak (lay-out) van een website gerealiseerd kan worden.

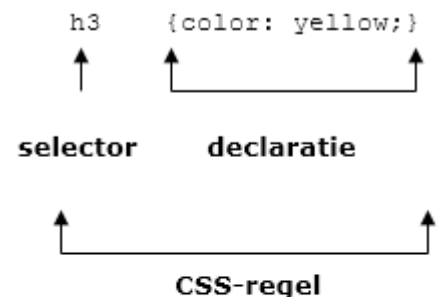
Voor de vormgeving van een webpagina worden stijlbladen gebruikt. In het Engels worden deze stijlbladen Cascading Style Sheets of afgekort CSS genoemd. De laatste standaard voor CSS wordt CSS3 genoemd.



Het werken met CSS biedt een aantal voordelen. Zo leg je één keer de basisopmaak van een website vast. Een wijziging, bijvoorbeeld de keuze voor een andere tekstkleur, breng je aan in dit basisbestand. Het resultaat is direct te zien op alle webpagina's die gekoppeld zijn aan dit basisbestand. Een ander groot voordeel is dat je met CSS elementen nauwkeurig kunt positioneren.

Syntax van CSS

Onder de syntax verstaan we de manier waarop CSS genoteerd wordt. In CSS wordt gebruik gemaakt van CSS-regels. Zo'n regel bestaat telkens uit een selector en een declaratie. Hiernaast zie je een voorbeeld van een CSS-regel:



In dit voorbeeld is h3 de selector en is color:yellow; de declaratie. De selector bepaalt op welk element de regel van toepassing is. Hier is dat dus een koptekst van niveau 3.

In de declaratie regel je vervolgens de opmaak van het betreffende element.

Als er een koptekst van niveau 3 wordt gebruikt, is de kleur van het lettertype geel. Dat is wat we hier in de CSS-regel hebben vastgelegd. De declaratie bestaat ook weer uit 2 onderdelen: een eigenschap (property) en een waarde (value).



De opbouw van een CSS-regel is telkens hetzelfde:

- Allereerst wordt de selector genoemd.
- Na het openen van de accolades volgt de eigenschap van de selector.
- Na elke eigenschap staat een dubbele punt.
- Achter de dubbele punt staat de waarde die de eigenschap krijgt.
- Daarna volgt een puntkomma.
- De accolades worden gesloten.

In een CSS-regel kunnen meerdere eigenschappen en bijbehorende waarden worden opgenomen. Er had dus ook kunnen staan:

```
h3 {color: yellow; font-size: 10px;}
```

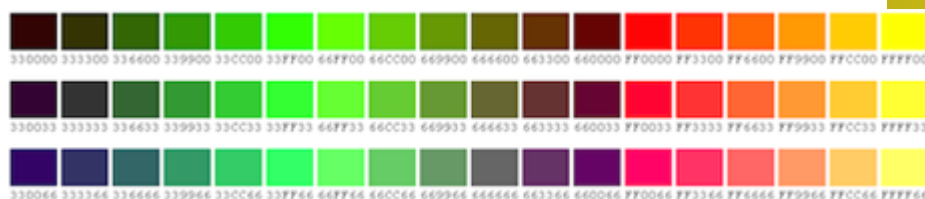
In bovenstaand voorbeeld is er sprake van 2 declaraties. Voor de leesbaarheid van de code wordt elke declaratie op een nieuwe regel begonnen. De notatie van de CCS-regel wordt dan:

```
h3
{
color: yellow;
font-size: 10px;
}
```

★ Aan de slag 21

- Leg uit wat we bedoelen met de syntax van CSS.
- Leg uit wat het verschil is tussen een selector, een eigenschap en een waarde in CSS.
- In een CSS-bestand willen we regelen dat voor de body van de webpagina het lettertype Verdana wordt gebruikt. De kleur van het lettertype is zwart.

Dat noteren we door de hexadematicale waarde voor de kleur zwart te gebruiken. Deze kleurwaarde kun je op internet opzoeken door te kijken naar kleurentabellen. De grootte van het gebruikte lettertype stellen we in op 100%. Hoe ziet de code voor het body-gedeelte van het CSS-bestand eruit?



Koppelen CSS en HTML

De koppeling van de opmaak aan de structuur van een website kan op 3 manieren:

1. Door gebruik te maken van een interne stylesheet
2. Door gebruik te maken van wat ook wel inline style genoemd wordt.
3. Door gebruik te maken van een externe stylesheet

Intern stylesheet

Het interne stylesheet maakt gebruik van het element `style`. Dit element heeft een begin- en een eindtag. Bij een intern stylesheet wordt de CSS-code tussen deze 2 tags in het head-gedeelte van het HTML-document geplaatst.

We willen voor de webpagina het lettertype (font) Arial gebruiken. Het lettertype krijgt de kleurwaarde `#ffc0cb`. De code voor het interne stylesheet wordt dan:

```
<head>
<style>
body {font-family: Arial; color: #ffc0cb;}
</style>
</head>
```

Inline style

Bij een inline style wordt style als attribuut gekoppeld aan het element waar de opmaakcode op van toepassing is.

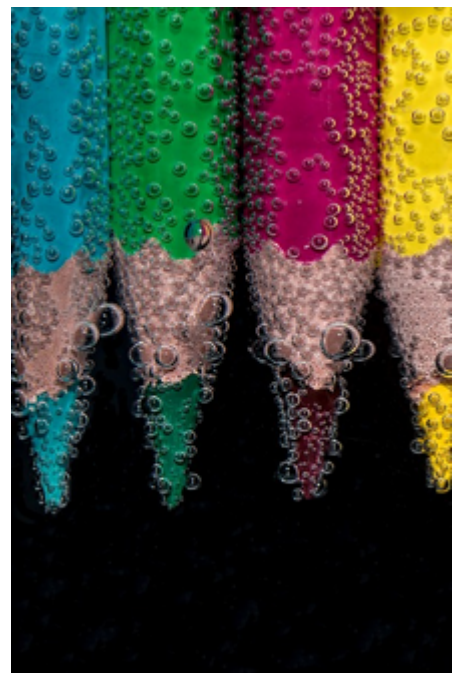
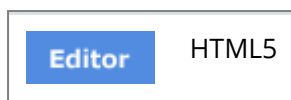
Hieronder zie je hoe we voor het element `p` het lettertype Arial gebruiken en de tekst de kleur `#ffc0cb` geven:

```
<p style="font-family: Arial; color: #ffc0cb;"> ... </p>
```



★ Aan de slag 22

In de HTML-tool zie je HTML-code met daarin een interne stylesheet. Bekijk hiervan het resultaat door op uitvoeren te klikken. Verander daarna het font in Verdana met als kleurcode `#cceecc`. Bekijk het resultaat opnieuw.



★ Aan de slag 23

Nu tref je in de HTML5 tool de HTML-code waarin een inline style is verwerkt.

Bekijk het resultaat.

Verander daarna het font in Verdana met als kleurcode #cceecc.

Bekijk de uitvoer van de code daarna opnieuw.

Editor

HTML5

Extern stylesheet

In deze module gaan we vooral in op het gebruik van externe stijlbladen.

Het gebruik van externe stijlbladen is ideaal als een website uit meerdere pagina's bestaat die allemaal dezelfde opmaak hebben. De lay-out van de webpagina's wordt in een apart CSS-bestand beschreven. De opmaakcode staat niet in de code van het HTML-document. Dit in tegenstelling tot het interne stylesheet en de inline style waarbij de CSS-code onderdeel is van de HTML-code.

Hoe koppel je nu een extern CSS-bestand aan een webpagina? Dat doe je door in het head-element van de pagina de onderstaande regel op te nemen:

```
<link rel="stylesheet" href="https://maken.wikiwijs.nl/opmaak.css">
```

We hebben het CSS-bestand de naam `https://maken.wikiwijs.nl/opmaak.css` gegeven. Met de tag `<link>` geven we aan dat de webpagina gekoppeld moet worden aan het vermelde CSS-bestand. In ons voorbeeld is dat dus `https://maken.wikiwijs.nl/opmaak.css`. Met `rel`, dit is de afkorting van relation, wordt aangegeven dat we te maken hebben met een stylesheet.

Je hebt al eerder kunnen lezen dat het handig is om voor de opmaakbestanden een aparte map met de naam `css` aan te maken. De koppeling van de webpagina aan het externe CSS-bestand wordt dan:

```
<link rel="stylesheet" href="https://maken.wikiwijs.nl/css/opmaak.css">
```

Deze regel zie je in het head-gedeelte van de HTML-code van ons voorbeelddocument terug:

```
<!DOCTYPE html>
<html lang="nl">
<head>
<title>sjabloon</title>
```



```

<meta charset="utf-8">
<link rel="stylesheet" href="https://maken.wikiwijs.nl/css/opmaak.css">
</head>
<body>
</body>
</html>

```

In het bestand <https://maken.wikiwijs.nl/opmaak.css> beschrijven we hoe de vormgeving van de webpagina `template.html` eruit moet gaan zien.

Een website bestaat meestal uit meerdere HTML-pagina's die aan elkaar zijn gelinkt. De pagina die bij het opvragen van het adres van de website standaard wordt geopend, noemen we `index.html` of als er gebruik gemaakt wordt van scriptingtalen bijvoorbeeld `index.php`.

In de tekst hieronder staat de koppeling schematisch uitgelegd.

template.html

```

<head>
  <title>sjabloon</title>
  <meta charset="utf-8">
  <link rel="stylesheet"
    href="css/opmaak.css">
</head>
<body>

</body>
</html>

```

opmaak.css

```

body
{
  font-family: arial;
  color: black;
}

```

★ Aan de slag 24

Open in de HTML-editor het bestand `template.html`. Voeg aan het body-gedeelte van het HTML-document een alinea toe met de tekst "Dit is een alinea". Verwijder de link die verwijst naar het gebruik van een apart CSS-bestand

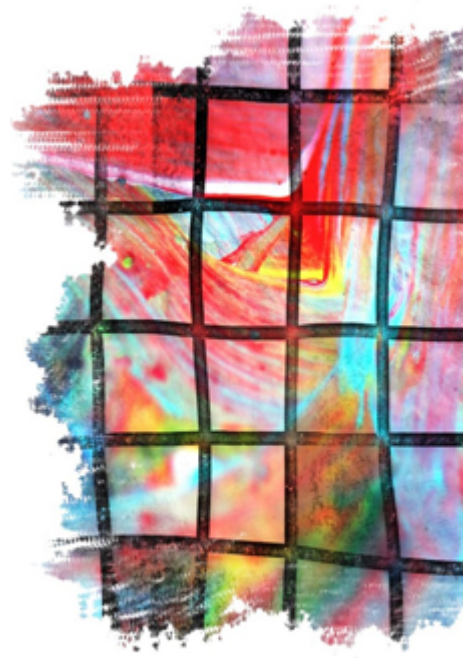
(`<link rel="stylesheet" href="https://maken.wikiwijs.nl/css/opmaak.css">`).

Regel daarna in het head-gedeelte van het document met een interne stylesheet dat voor de body het lettertype Courier met de tekstkleur #7b68ee wordt gebruikt.

Bekijk het resultaat door op uitvoeren te drukken.

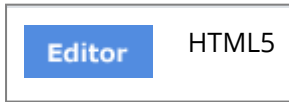
Editor

HTML5



★ Aan de slag 25

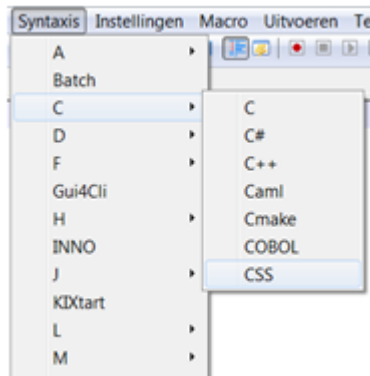
Open in de HTML-editor het bestand [htmlcss_1.html](#). Dit bestand is onderdeel van het zip-bestand dat je al eerder hebt moeten downloaden en hebt uitgepakt in je home-directory in de map html. Pas de HTML-code zo aan dat er in plaats van een interne stylesheet gebruik gemaakt wordt van de inline style voor het opmaken van de alinea. Bekijk het resultaat in de editor.



★ Aan de slag 26

- a. Maak in de map html waar je tot nu toe al je HTML-bestanden hebt opgeslagen een aparte map css aan.

- b. Open Notepad++ en klik in de menubalk van het programma op het tabblad Syntaxis. Ga naar de letter C en klik op CSS. Je hebt nu in Notepad++ aangegeven dat je code gaat



schrijven voor een CSS-document. Bij het opslaan van het bestand wordt automatisch de extensie .css toegevoegd.

- c. Sla het bestand op als [opmaak.css](#) in de map css die je al eerder hebt aangemaakt. We hebben nu een externe stylesheet met de naam [opmaak.css](#) gemaakt.

★ Aan de slag 27

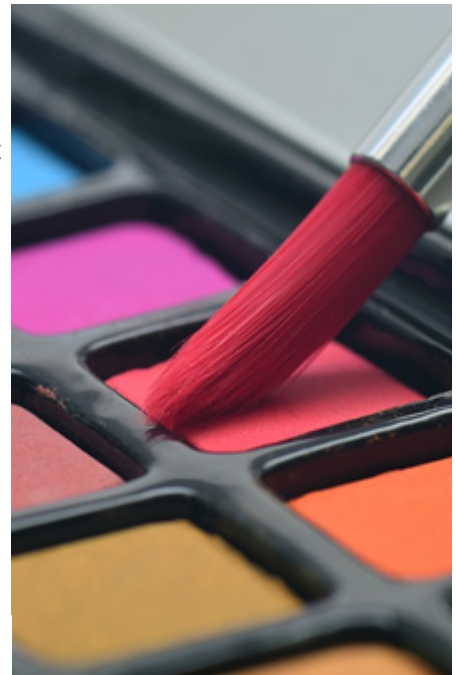
Open in de editor achtereenvolgens de bestanden [template.html](#) en [opmaak.css](#).

Deze bestanden staan in je home-directory in de map html en de map css. Regel voor de body van het html bestand dat de achtergrondkleur van de pagina de waarde #66CDAA krijgt. Voor het instellen van de kleur van een achtergrond gebruik je in CSS de eigenschap background-color.

Klik in de editor op de knop "Uitvoeren" en controleer of de achtergrondkleur van de webpagina is veranderd.

Editor

HTML5



Eigenschappen lettertypen

In de declaratie van een CSS-regel staan eigenschappen met de daarbij behorende waarden. Bijvoorbeeld: `color: yellow;` In dit onderdeel bespreken we de meest gangbare eigenschappen/waarden die gebruikt kunnen worden om tekst op te maken. Een andere naam voor lettertype is font.



Eigenschap	Waarde(n)
font-family	Verschillende lettertypes
font-style	normal italic oblique
font-weight	normal bold bolder lighter

font-size	px pt % em
-----------	---------------------

Font-family

De keuze voor het lettertype van de webpagina regel je met de eigenschap font-family. In het CSS-bestand zou de regel font-family: Verdana, "Times New Roman", Arial; kunnen staan. Het lettertype dat als eerste gebruikt wordt, is Verdana.

Zou het font Verdana niet op de computer van de bezoeker van de website staan, dan wordt automatisch het tweede genoemde lettertype gebruikt. Dat is hier Times New Roman. De naam van dit font staat in het CSS-bestand tussen aanhalingstekens. Dat is verplicht voor lettertypen waarvan de naam uit meerdere woorden bestaat.

Font-style

Met font-style geef je aan of tekst normaal, cursief of schuin wordt weergegeven. Hiervoor worden achtereenvolgens de waarden normal, italic en oblique gebruikt.

```
font-style: normal;
font-style: italic;
font-style: oblique;
```

Font-weight

Font-weight gebruik je om het lettergewicht te regelen. Dat wil zeggen dat je met deze eigenschap aan kunt geven of een letter vet, minder vet of normaal wordt getoond. Het Engelse woord voor vet is bold. We hebben in de tabel niet alle waarden genoemd die gebruikt kunnen worden voor de eigenschap font-weight. Het gewicht van een letter kan worden uitgedrukt in een getal. Het getal 400 komt overeen met de waarde normal, het getal 700 met de waarde bold.

```
font-weight: normal;
font-weight: bold;
font-weight: 700;
font-weight: bolder;
font-weight: lighter;
```

Font-size

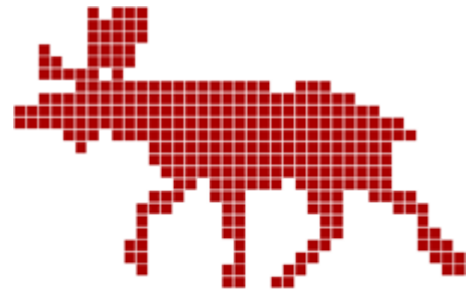
Met font-size wordt de grootte van het lettertype geregeld.

De grootte van een lettertype kan worden aangegeven in pixels (px), punten (pt), procenten (%) of in em. De W3C-groep beveelt aan om voor de grootte van het lettertype em te gebruiken.

```
font-size: 16px;
font-size: 16pt;
font-size: 100%;
font-size: 1em;
```

★ Aan de slag 28

- Zoek op waarom je de grootte van een font beter niet in pixels kunt opgeven.
- Leg uit waarom het verstandig is om op het einde van een lijst met lettertypes een algemeen lettertype op te geven.
- Zoek uit waar de term "em" voor het aangeven van de grootte van een lettertype vandaan komt.
- We willen in CSS de kleur van het lettertype van het element h4 roze maken.
Werk drie manieren uit waarop we dat in CSS kunnen regelen.



Bespreek je antwoorden met een klasgenoot

Eigenschappen tekst

Eigenschappen tekst Color

De eigenschap color wordt gebruikt om de kleur van het gebruikte lettertype aan te geven. De notatie van de kleur kan door middel van een RGB-waarde, een hexadecimale waarde of door gewoonweg een standaardkleur te noemen.

color: rgb(0,0,255);
color: #0000ff;
color: blue;

Text-decoration

Het woordje decoration betekent letterlijk versiering. Met deze eigenschap regel je dat een woord wordt onderstreept (underline) of bijvoorbeeld doorgestreept (line-through). De waarde overline zorgt ervoor dat er een lijn boven de tekst wordt geplaatst.

text-decoration: none;
text-decoration: overline;
text-decoration: line-through;
text-decoration: underline;

Text-transform

Soms wil je een tekst alleen in hoofdletters (uppercase) of alleen in kleine letters (lowercase) weergeven. Daarvoor gebruik je de eigenschap text-transform. De waarde capitalize zorgt ervoor dat elk woord in de tekst begint met een hoofdletter.

text-transform: uppercase;
text-transform: lowercase;
text-transform: capitalize;



Text-align

Voor het uitlijnen van een tekst wordt de eigenschap text-align gebruikt. Een tekst kan worden gecentreerd (center), rechts worden uitgelijnd (right), links (left) worden uitgelijnd of worden uitgevuld (justify). Uitvullen wil zeggen dat de tekst zowel links als rechts wordt uitgelijnd.

De teksten in kranten en tijdschriften zijn vaak uitgevuld. Bijvoorbeeld wanneer teksten in meerdere kolommen staan. Uitvullen kun je beter alleen gebruiken bij voldoende woorden op een regel, anders kun je grote stukken wit tussen de woorden krijgen.

text-align: center;
text-align: right;
text-align: left;
text-align: justify;

text-indent

Met text-indent kan de eerste regel van een alinea worden ingesprongen. Standaard staat de waarde op 0 ingesteld.

text-indent: 3em;

In de tekst hieronder staat alles nogmaals in een schema uitgelegd.

Eigenschap	Waarde(n)
color	rgb(0,0,255) #0000ff blue
text-decoration	none overline line-through underline
text-transform	uppercase lowercase capitalize
text-align	center right left justify
text-indent	px em

★ Aan de slag 29

In het onderdeel over HTML5 hebben we uitgelegd wat het verschil is tussen de attributen id en class. Voor deze opdracht maken we gebruik van het bestand [id_class.html](#). Dit bestand tref je aan in het zip-bestand dat je al eerder hebt moeten downloaden en uitpakken in je home-directory in de map html.

In de tekst hieronder staat broncode van dit bestand.

```
<h3 class="naam">
Koptekst eerste alinea
</h3>
<p>
Tekst eerste alinea
</p>
<h3>
Koptekst tweede alinea
</h3>
<p class="naam">
Tekst tweede alinea
</p>
<h3>
Koptekst derde alinea
</h3>
<p id="naam">
Tekst derde alinea
</p>
```



Open in een browser het bestand [id_class.html](#) en bekijk het resultaat. We gaan de lay-out van deze webpagina nu regelen in de externe stylesheet [opmaak.css](#)

Open in Notepad++ het bestand [opmaak.css](#) dat staat in de map css. Regel in het CSS-bestand dat de koptekst h3 het font Arial en de kleur blue krijgt. Een alinea krijgt als font Courier en de kleur van het font wordt red. Sla het bestand op onder dezelfde naam. Dus [opmaak.css](#). Open daarna het bestand [id_class.html](#) en bekijk het resultaat. Of ververs de webpagina als het bestand nog geopend is. Hebben de kopteksten `<h3 class="naam">` en `<h3>` een andere opmaak gekregen?

★ Aan de slag 30

- Open in Notepad++ het bestand **opmaak.css** dat staat in de map css. Voor de alinea met de id naam moet het lettertype Arial met als kleur yellow gebruikt worden. Sla het bestand op onder dezelfde naam. Open daarna in een browser het bestand **id_class.html** Wat valt je op?
- Open in Notepad++ het bestand **opmaak.css** dat staat in de map css. Regel voor alle elementen met het attribuut class dat de tekst cursief wordt weergegeven. Voor het lettertype gebruiken we Verdana. Sla het bestand nadat deze wijzigingen zijn aangebracht op onder dezelfde naam. Open in een browser het bestand **id_class.html** en bekijk het resultaat.

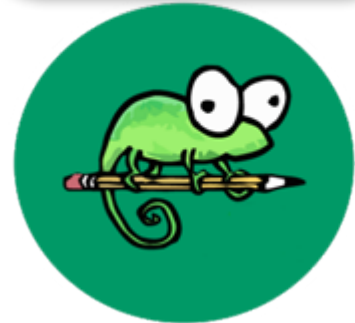


★ Aan de slag 31

Open in Notepad++ het bestand **opmaak.css** dat staat in de map css.

Voeg in het CCS-bestand de volgende code toe:

```
h3.naam
{
font-family: arial;
color: pink;
font-weight: bold;
}
```



Sla het bestand op onder dezelfde naam. Open het bestand **id_class.html** in een browser. Wat kun je aan de hand van wat je ziet, concluderen?

Verwijzingen

De opmaak van verwijzingen, ook wel links genoemd, regel je in een CSS-bestand. Een verwijzing in een webpagina is vaak te herkennen aan de kleur van de tekst en aan het feit dat de tekst van de verwijzing onderstreept is. Het is mogelijk om de standaardinstellingen van een verwijzing naar eigen voorkeur aan te passen.

Zo wordt de eigenschap en waarde `text-decoration: none;` ook wel gebruikt om de lijn onder een link te laten verdwijnen omdat dat mooier uitziat.

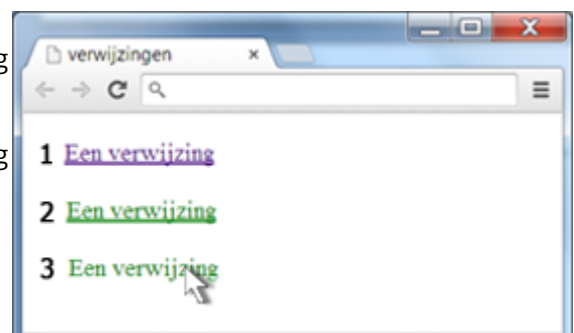
Een verwijzing kan in 4 verschillende stadia verkeren:

1.	a: link	Verwijzing die nog niet bezocht is
2.	a: visited	Bezochte verwijzing
3.	a: hover	Verwijzing waar overheen gegaan wordt met de muis
4.	a: active	Verwijzing op het moment waarop er op geklikt wordt

De opmaak van de status waarin een verwijzing verkeert, leg je vast in CSS. Lees de tekst.

1	2	3
<code>a:link { color: blue; text-decoration: underline; }</code>	<code>a:visited { color: green; }</code>	<code>a:hover { text-decoration: none; }</code>

1. De bezoeker van de webpagina heeft de verwijzing nog niet bezocht. De kleur van de verwijzing is blauw en de verwijzing is onderstreept.
2. De bezoeker van de webpagina heeft de verwijzing nu wel bekeken. De kleur van de verwijzing is nu groen.
3. De bezoeker van de webpagina beweegt met de muis over de verwijzing. De verwijzing is op dat moment niet meer onderstreept.



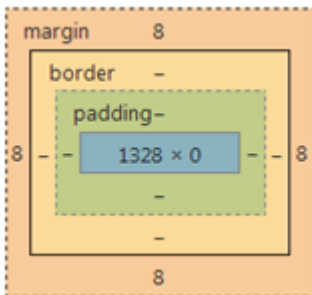
Het box model

In HTML kunnen alle elementen worden voorgesteld als een box (doos). Dat geldt voor tekst, een afbeelding, een kop etc. Op een webpagina treffen we meerdere van dit soort dozen aan. De manier waarop deze dozen zich tot elkaar verhouden, wordt beschreven in wat ook wel het box model wordt genoemd.

In de ontwikkelomgeving van Google Chrome, deze omgeving is al eerder besproken, zie je onder de CSS-code een afbeelding van het box model en de daarbij voor het geselecteerde element bijbehorende eigenschappen en waarden. Hieronder staan de instellingen die van toepassing zijn op de body van het bestand template.html.

Een box of doos heeft verschillende eigenschappen:

- margin (de ruimte rondom de rand)
- border (de rand)
- padding (de ruimte rondom de inhoud)
- content (de inhoud van het element)



★ Aan de slag 32

- a. Open in Notepad++ het bestand **template.html** Voeg aan de body van het HTML-document de code uit de onderstaande tekst toe.

```
<section>
<article>
<h2 class= "rand">
Dit is een koptekst
</h2>
<p class= "rand">
Dit is een alinea
</p>
</article>
<article>
<h2 class= "rand">
Dit is een koptekst
</h2>
<p class= "rand">
Dit is een alinea
</p>
</article>
</section>
```



Sla het bestand op in de map html onder de naam **boxmodel.html**

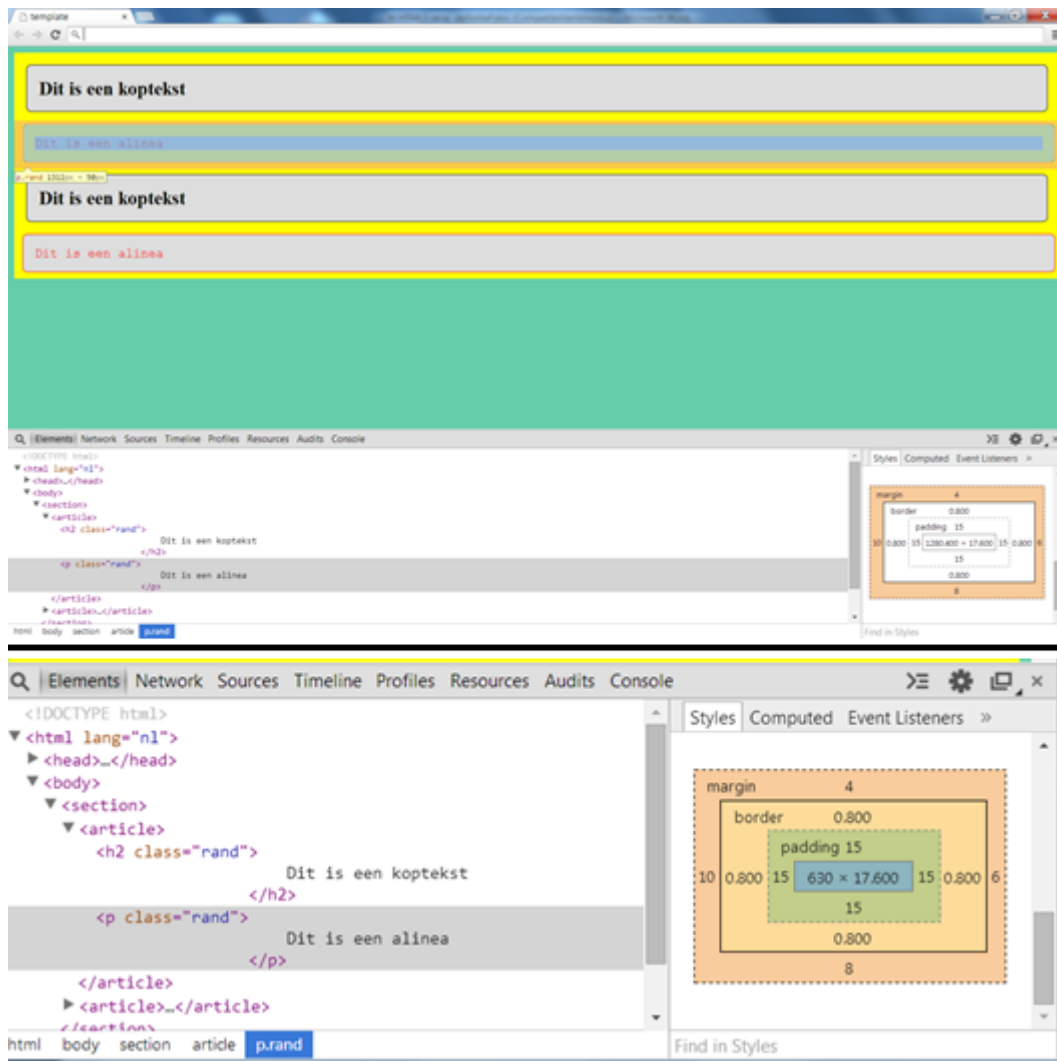
- b. Open in Notepad++ het bestand **opmaak.css**
Voeg de code uit de onderstaande tekst toe voor de opmaak van het element p met het attribuut class= "rand" en voor de opmaak van het element h2 met het attribuut class= "rand" .

```
p.rand
(
background: #dddddd;
margin: 15px 15px 15px 15px;
padding: 15px 15px 15px 15px;
border: 1px solid;
border-radius: 7px;
}
```

```
h2.rand
(
background: #dddddd;
margin: 15px 15px 15px 15px;
padding: 15px 15px 15px 15px;
border: 1px solid;
border-radius: 7px;
}
```

Sla daarna het bestand op onder dezelfde naam
Open het bestand **boxmodel.html** in een browser.
Wat is de functie van de regel border-radius: 7px;?

- c. Open in Chrome het bestand **boxmodel.html**
Activeer de ontwikkelomgeving en verander de waarden van de margin van het eerste element p in de code van margin: 15px 15px 15px 15px; in margin: 4px 6px 8px 10px;.
Bekijk wat er verandert in het boxmodel van de ontwikkelomgeving.
De veranderingen die je aanbrengt in de ontwikkelomgeving worden niet opgeslagen.
Na het sluiten van de browser heeft het element p met het attribuut class= "rand" voor de margin nog steeds de waarden margin:15px 15px 15px 15px;.
Voor meer uitleg bekijk de afbeelding hieronder.



- d. Open in Notepad++ het bestand **opmaak.css** Verander voor het element p met het attribuut `class="rand"` de waarde van de padding van `padding:15px 15px 15px 15px;` in `padding: 15px;`. Sla het bestand op onder dezelfde naam.
Open nu in Chrome het bestand **boxmodel.html** en bekijk de instellingen van het element p in de ontwikkelomgeving. Verander in het CSS-gedeelte de regel `padding: 15px 15px 15px 15px;` in `padding: 15px;`.
Welke conclusie kun je nu trekken?

Positionering elementen

Nu we weten hoe in HTML het box model werkt, laten we zien hoe bepaald wordt waar een element (box) op een webpagina komt te staan. In CSS kunnen aan elementen met de eigenschap "position" waarden worden toegekend waarmee de plaats van een element (box) op een webpagina kan worden vastgelegd. We noemen dat het positioneren van elementen. De eigenschap position kan 4 waarden hebben:

1. **static**
Standaard worden elementen gepositioneerd met deze waarde. Een element (box) komt op de plaats waar het volgens de normale opbouw van de pagina zou moeten staan.
In CSS: position: static;
2. **relative**
Relatieve positionering houdt in dat een element (box) verplaatst wordt ten opzichte van de positie die het normaal zou hebben. De verplaatsing is niet van invloed op omliggende elementen (boxen). Het is dus mogelijk dat onderdelen elkaar gaan overlappen en/of afdekken. Om de positie van een element aan te geven worden de eigenschappen top, right, bottom en left gebruikt.
In CSS: position: relative;
3. **absolute**
Bij een absolute positionering van een element (box) wordt het element op een aangegeven plaats binnen het omliggende element geplaatst. De plaats van het element wordt bepaald door een verticaal en een horizontaal punt. Voor het aangeven van de verticale plaats worden de eigenschappen top en/of bottom gebruikt. Voor het bepalen van de horizontale plaats gebruiken we left en/of right. Met top left en top right bepalen we de linker- en rechterbovenhoek van het element. Bottom left en bottom right bepalen de linker- en rechteronderhoek van het element. De coördinaten die worden vastgesteld, bepalen de plaats van het element. Daarbij wordt géén rekening gehouden met wat er op de aangegeven plek staat. Er kan dus ook hier sprake zijn van overlappingen.
In CSS: position: absolute;
4. **fixed**
Fixed wil zeggen dat het element (box) gefixeerd wordt binnen het zichtbare venster. Het blijft altijd op de aangegeven plek staan. Ook al scroll je in een browser met een schuifbalk naar beneden. De plaats van het element kan op dezelfde manier worden aangegeven als bij de absolute positionering van een element.
In CSS: position: fixed;



Bekijk het volgende filmpje daarna kan je eventueel de tekst nog lezen.



<https://maken.wikiwijs.nl/bestanden/555737/Positie.mp4>

Positioneren van elementen

Aan de hand van een paar voorbeelden zullen we proberen te verduidelijken wat het effect is van deze 4 verschillende waarden. We gebruiken daarvoor een webpagina met de afbeelding en beschrijving van

een muziekinstrument.

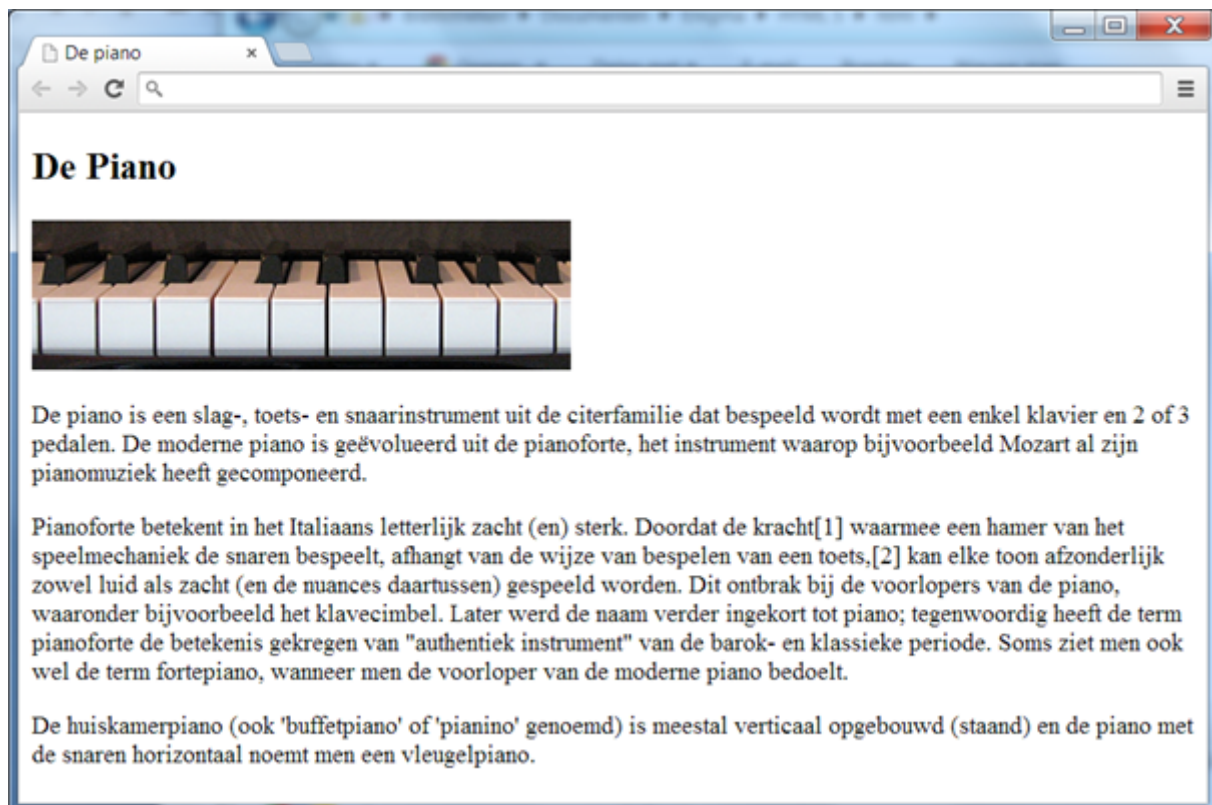
Namelijk een piano. Voor de beschrijving van de piano hebben we de tekst van een wikipediapagina gebruikt.

Piano

In de broncode van onze webpagina staat in de body van het HTML-document:

- een element H2.
- een afbeelding van de toetsen van een piano. We hebben van de afbeelding een class met de naam piano gemaakt.
- drie keer een alinea's met teksten over de piano.

In het HTML-document wordt verwezen naar een apart CSS-bestand. We hebben dat bestand voor het gemak even piano.css genoemd.



Hieronder zie je de broncode van piano.html:

Broncode Piano

```
<!DOCTYPE html>
<html lang="nl">
<head>
<title>De piano</title>
<meta charset="utf-8">
<link rel="stylesheet" href="https://maken.wikiwijs.nl/css/piano.css">
</head>
```

```

<body>
<h2>
De Piano
</h2>

<p>
De piano is een slag-, toets- en snaarinstrument uit de ..... pianomuziek heeft gecomponeerd
</p>
<p>
Pianoforte betekent ..... piano bedoelt.
</p>
<p>
De huiskamerpiano ..... een vleugelpiano.
</p>
</body>
</html>

```

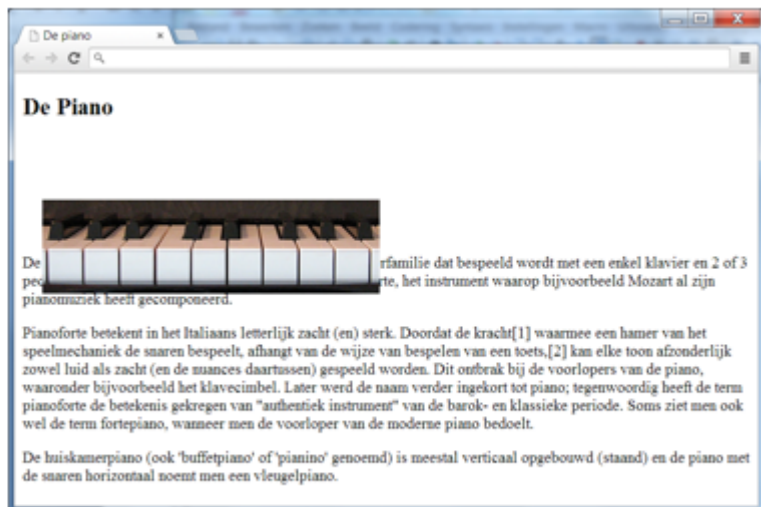
We gaan het element `img class="piano"` een relatieve positie geven. In `piano.css` noteren we:

Relatieve positie

```

.piano
{
    position: relative;
    top: 60px;
    left: 20px;
}

```

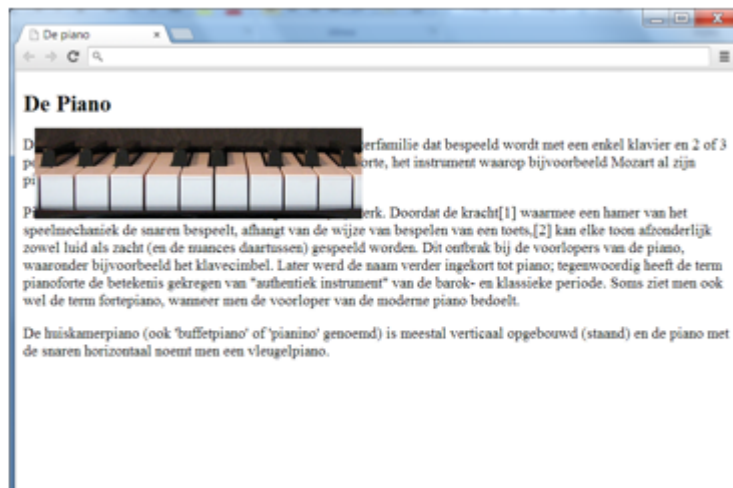


De afbeelding van de pianotoetsen is nu verplaatst en staat deels over de tekst van de eerste alinea. De afbeelding is 60px naar beneden en 20px naar links verschoven. Bedenk dat de verplaatsing naar beneden ten opzichte van het element `h2` is. In de broncode van het HTML-document wordt eerst het `h2`-element en daarna de afbeelding opgenomen. Zouden we de afbeelding als eerste in de body opnemen en daarna het `h2`-element krijgen we de afbeelding op een andere plek te zien. De afbeelding van de pianotoetsen staat nu voor een groot deel over het element `h2` heen.

We kunnen de afbeelding ook een absolute positie geven.
We passen daarvoor de CSS-code aan.
Links staat de aangepaste code, rechts zien we het resultaat:

Absolute positie

```
.piano
{
    position: absolute;
    top: 60px;
    left: 20px;
}
```

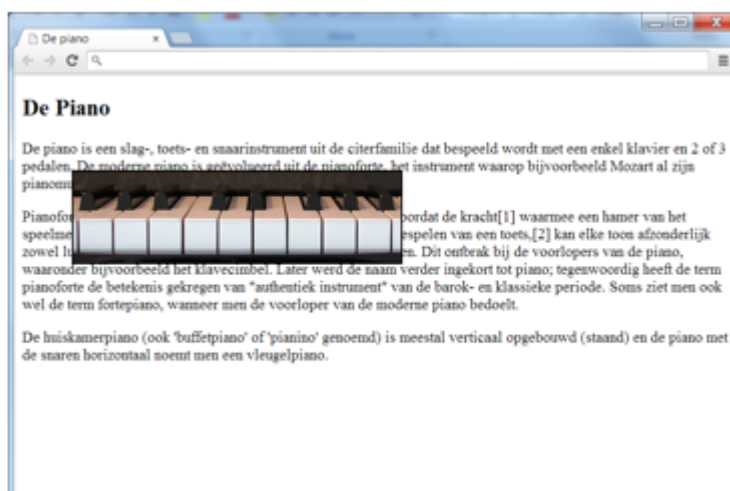


De pianotoetsen staan nu over delen van de eerste en tweede alinea. Het omliggende element waar bij het bepalen van de plaats van de afbeelding naar gekeken wordt, is in dit voorbeeld het element body. De afbeelding krijgt nu als linkerbovenhoek de waarde 20 pixels naar links en 60 pixels naar beneden mee.

We passen nogmaals de CSS-code van de class piano aan. We gaan nu voor het element img class="piano" een border en marge instellen. De afbeelding zakt door de nieuwe instellingen nog verder naar beneden en verschuift nog meer naar rechts.

Border marge

```
.piano
{
    position: absolute;
    top: 60px;
    left: 20px;
    border: 2px solid;
    margin: 40px;
}
```

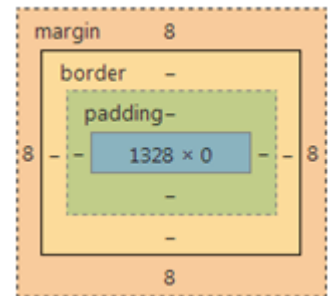


De eigenschappen en waarden van het box model bepalen dus mede op welke plek de inhoud van

een element wordt afgebeeld.

Float

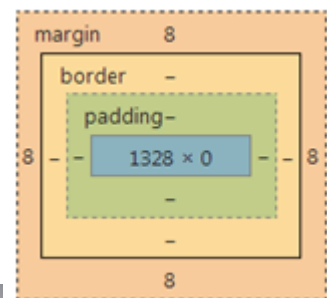
Naast het gebruik van de eigenschap `position` bestaat er ook een eigenschap `float`. Je kunt `float` aan ieder element toekennen. Je maakt er dan een zogenaamd zwevend element van. De waarden zijn `none`, `left` of `right`. Als je `float` toekent aan een element, dan wordt de top gelijk gezet met de top van de box waarin het element staat. Verder wordt het element zo ver mogelijk naar links of naar rechts geplaatst als mogelijk is, afhankelijk van wat je aangeeft.



Er zijn aandachtspunten waar je rekening mee moet houden bij het gebruik van elementen die de `float`-eigenschap hebben. Je moet een breedte (`width`) meegeven en het is mogelijk dat elementen elkaar gaan overlappen. Om dit tegen te gaan kun je gebruik maken van de eigenschap `clear` met de waarden `none`, `left`, `right` of `both`.

★ Aan de slag 33

- Leg uit wat het verschil is tussen `position: absolute` en `position: fixed`
- Waarvoor worden bij de eigenschap `clear` de waarden `left`, `right`, `both` en `none` gebruikt?



★ Aan de slag 34

- Open in Notepad++ het bestand **opmaak.css**. Voeg voor het element section de eigenschap `float` en de waarde `right` toe. Sla het bestand op onder dezelfde naam. Open daarna in een browser het bestand **structuur.html**. Was is er met het element section gebeurd?
- Open in Notepad++ het bestand **opmaak.css**. Pas de CSS-code voor het element section zo aan dat dit element weer net zo breed wordt afgebeeld als de andere structuurelementen. Je moet nog wel gebruik blijven maken van de regel `float: right`; Sla het bestand op onder dezelfde naam.



★ Aan de slag 35

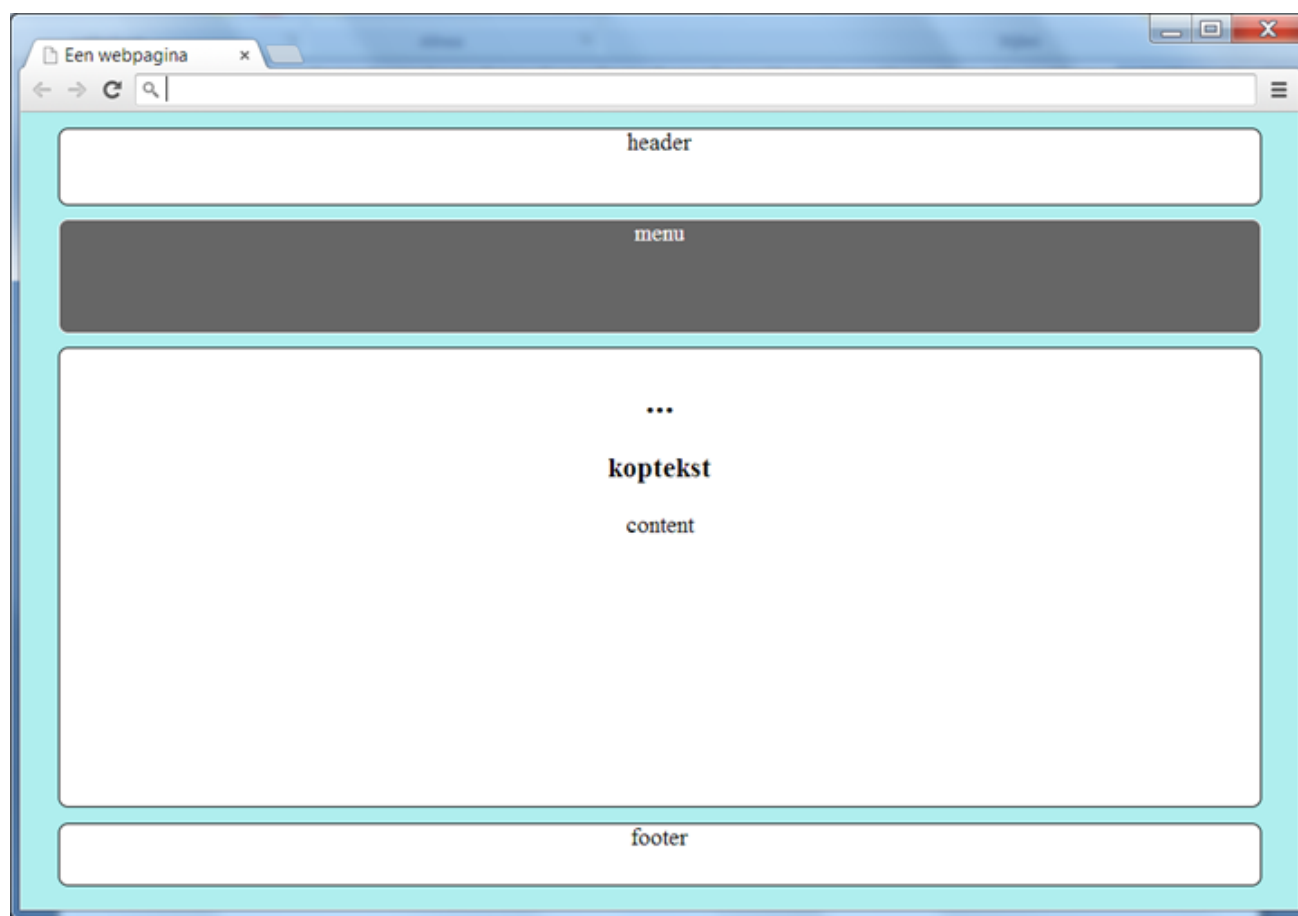
In deze opdracht gaan jullie code schrijven voor een HTML- en een CSS-bestand. Het HTML-bestand moet worden opgeslagen onder de naam **htmlcss_2.html**. Het CSS-bestand krijgt de naam **structuur.css**. Gebruik voor het schrijven van de code van beide bestanden Notepad++.

De opdracht is om de afgebeelde webpagina uit de popup na te bouwen.

Maak hiervoor in het HTML-document gebruik van de structuur-elementen header, nav, section (inclusief één article met koptekst h3 en één alinea) en footer. Regel de lay-out van de webpagina in het bijbehorende CSS-bestand.

Hieronder volgen nog enkele aanwijzingen:

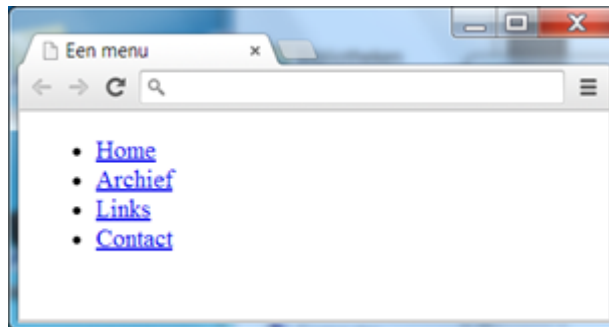
- Voor het positioneren van de structuurelementen gebruik je position: absolute;
- Het element body heeft de achtergrondkleur #afeeee.
- De elementen header, section en footer hebben de achtergrondkleur #ffffff.
- Het element nav heeft de achtergrondkleur #666666.
- Maak voor de ronde hoeken van de elementen gebruik van de eigenschap border-radius.



Een menu in HTML en CSS

Een menu is een relevant onderdeel van een website. Het moet opvallen en overzichtelijk zijn zodat mensen snel kunnen vinden wat ze nodig hebben. Het is de belangrijkste manier om door een website te navigeren.

Je hebt menu's in veel verschillende soorten en stijlen. Vaak wordt er voor een menu gebruik gemaakt van JavaScript in combinatie met een aparte stylesheet voor het menu. Wij beperken ons hier tot een menu met een enkele laag. We hebben daar alleen HTML en CSS voor nodig.



In de HTML-code voegen we een ongeordende lijst toe.

```
<ul id="menu">
<li><a href="https://maken.wikiwijs.nl/#">Home</a></li>
<li><a href="https://maken.wikiwijs.nl/#">Archief</a></li>
<li><a href="https://maken.wikiwijs.nl/#">Links</a></li>
<li><a href="https://maken.wikiwijs.nl/#">Contact</a></li>
</ul>
```

Op de plek van het `https://maken.wikiwijs.nl/#`-teken plaats je de naam van het bestand waarnaar verwezen moet worden.

Voor home is dat bijvoorbeeld `index.html`.

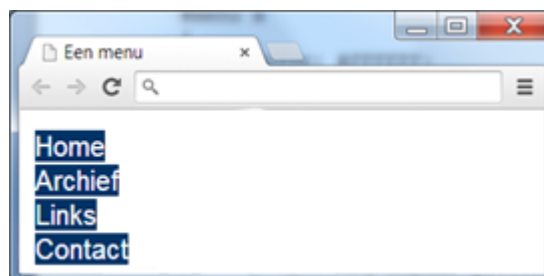
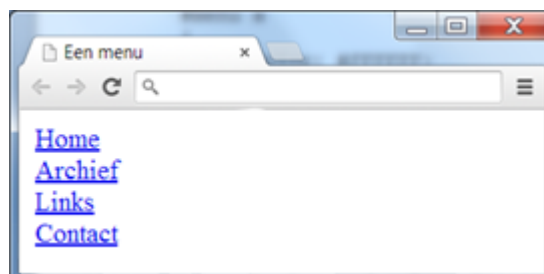
Omdat je over het algemeen maar één menu hebt, maken we gebruik van een id in plaats van een class. We gaan nu CSS-regels toevoegen.

Om te beginnen gaan we markeringstekens die een ongeordende lijst standaard heeft, weghalen. De padding en margin stellen we in op 0.

```
https://maken.wikiwijs.nl/#menu
{
list-style-type: none; /*Haal markeringstekens weg.*/
margin: 0; /*Geen margin.*/
padding: 0; /*Geen padding.*/
}
```

Vervolgens passen we de verwijzingen aan.

```
https://maken.wikiwijs.nl/#menu a
{
color: https://maken.wikiwijs.nl/#FFFFFF; /*Kleur verwijzing
wordt wit.*/
text-decoration: none; /*Tekst niet onderstreept.*/
background-color: https://maken.wikiwijs.nl/#003366;
/*Achtergrondkleur blauw.*/
font-family: Arial; /*Ander lettertype.*/
}
```

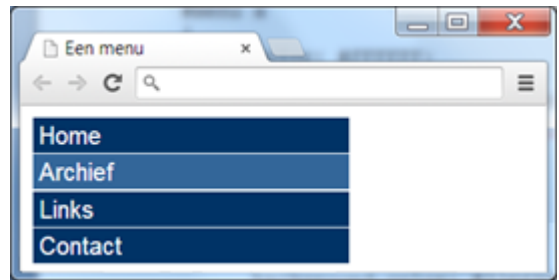


Wat opvalt is dat er bijna niets is veranderd. Er is al helemaal niets terecht gekomen van de breedte. Verder wordt de doorzichtige 'rand' die we er nu expliciet in zetten ook niet weergegeven. Dit heeft allemaal te maken met het feit dat de `a`-tag een inline element is en breedte daar niet op van toepassing is. Door de declaratie `display: block`; toe te voegen, wordt de verwijzing behandeld als block element.

Hieronder staat de uiteindelijke CSS-code van het menu.
De afbeelding rechts is het eindresultaat

<https://maken.wikiwijs.nl/#menu>

```
{
list-style-type: none; /*Haal markeringstekens weg.*/
margin: 0; /*Geen margin.*/
padding: 0; /*Geen padding.*/
}
https://maken.wikiwijs.nl/#menu a
{
color: https://maken.wikiwijs.nl/#FFFFFF; /*Kleur verwijzing wordt wit.*/
text-decoration: none; /*Tekst niet onderstreept.*/
background-color: https://maken.wikiwijs.nl/#003366; /*Achtergrondkleur blauw.*/
font-family: Arial; /*Ander lettertype.*/
width: 200px; /*Breedte naar 200px.*/
padding: 2px 4px; /*Wat padding.*/
margin-bottom: 1px; /*De doorzichtige 'rand'.*/
display: block; /*Als block behandelen.*/
}
https://maken.wikiwijs.nl/#menu a:hover
{
background-color: https://maken.wikiwijs.nl/#336699; /*Kleur als muis er overheen gaat.*/
}
```



★ Aan de slag 36

- a. Open in Notepad++ het bestand `htmlcss_2.html` In deze webpagina heb je de structuurelementen `header`, `nav`, `section`, `article` en `footer` gebruikt. We gaan tussen de tags `<nav>` en `</nav>` HTML-code voor een menu opnemen. Gebruik daarvoor de code uit de onderstaande tekst.

```
<ul>
<li><a href="https://maken.wikiwijs.nl/#">Home</a></li>
<li><a href="https://maken.wikiwijs.nl/#">Archief</a></li>
<ul>
<li><a href="https://maken.wikiwijs.nl/#">Archief 1</a></li>
<li><a href="https://maken.wikiwijs.nl/#">Archief 2</a></li>
</ul>
<li><a href="https://maken.wikiwijs.nl/#">Links</a></li>
<li><a href="https://maken.wikiwijs.nl/#">Contacts</a></li>
</ul>
```

Sla het bestand op als `htmlcss_3.html` en bekijk het resultaat in een browser. Er verschijnt nu een menu, maar de opmaak daarvan is nog niet geregeld.

- b. Open Notepad++ en neem code uit de onderstaande tekst over. Sla daarna het bestand op als `screen_menu balk.css` in de map `css`.

<pre>nav { margin: 0; padding: 0; } nav a {</pre>	<pre>nav ul { list-style-type: none; margin: 0; padding: 0; } nav li</pre>
---	--



<pre> color: https://maken.wikiwijs.nl/#FFFFFF; text-decoration: none; background-color: https://maken.wikiwijs.nl/#003366; font-family: Arial, sans-serif; width: 100px; padding: 2px 4px; margin-bottom: 1px; display: block; list-style-type: none; } nav a:hover { background-color: https://maken.wikiwijs.nl/#336699; color:https://maken.wikiwijs.nl/#FFFFFF; } </pre>	<pre> { float: left; background- color:https://maken.wikiwijs.nl/#003366; min-width:100px; } nav ul li ul { display: none; } nav ul li ul li { float: none; } nav ul li:hover ul { display: block; } </pre>
---	---

- c. Open in Notepad++ het bestand `htmlcss_3.html`
 Koppel het CSS-bestand `screen_menubalk.css` aan het bestand `htmlcss_3.html`
 Sla het bestand op als `htmlcss_4.html`
 We hebben de opmaak van het menu in een apart CSS-bestand geregeld.

Toets 2

In de volgende toets wordt getoetst wat je inmiddels zou moeten weten over CSS3.
 Maak de toets.

Adaptieve Toets

CSS3



D JavaScript

Client-side scriptingtaal

In HTML5 speelt JavaScript een belangrijke rol. JavaScript is een client-side scriptingtaal. Dat houdt in dat het script in de browser van je computer wordt uitgevoerd. Er bestaan ook scriptingtalen die op een server worden uitgevoerd.

Dit zijn server-side scriptingtalen. Een voorbeeld daarvan is de taal PHP.

De scriptingtaal PHP wordt in Enigma in een aparte module besproken. We gaan op deze taal hier verder niet in.

Met JavaScript kan worden geprogrammeerd. Door het uitvoeren van die programmacode kan een webpagina interactief worden gemaakt. Zo leent JavaScript zich uitstekend voor het programmeren van diverse soorten menu's die je voor een website zou kunnen gebruiken. Bijvoorbeeld voor een drop-down menu dat is geprogrammeerd met JavaScript en waarvan de lay-out wordt beschreven in een apart CSS-bestand.



We ronden deze module af met JavaScript. Het is niet onze intentie om in detail te bespreken hoe je moet leren programmeren in JavaScript. Wat we wel nastreven, is om aan de hand van enkele praktische voorbeelden te laten zien wat de mogelijkheden zijn van JavaScript.

JavaScript wordt beschouwd als de programmeertaal van het World Wide Web. In veel moderne HTML-pagina's is deze programmeertaal terug te vinden.

JavaScript-code

JavaScript-code staat altijd tussen de tags `<script>` en `</script>`. Deze tags markeren het punt waarop JavaScript start en eindigt. De scripts in een HTML-pagina kunnen op diverse plekken staan:

1. In het headgedeelte van een HTML-pagina.
2. In het bodygedeelte van een HTML-pagina.
3. In zowel het head-als bodygedeelte van een HTML-pagina.

De JavaScript-code kan ook worden opgeslagen in een extern bestand. JavaScript-bestanden eindigen altijd op de extensie `.js` en worden vaak bewaard in een aparte map. Die map zou je bijvoorbeeld `js` kunnen noemen. Bij het bespreken van stylesheets hebben we uitgelegd hoe een extern CCS-bestand moet worden gekoppeld aan een HTML-document. Kies je ervoor om een extern JavaScript-bestand te gebruiken dan moet er net als bij externe CSS-bestanden een koppeling met het HTML-document worden gerealiseerd. Dat doe je door de



onderstaande regel op te nemen in het head-gedeelte van het HTML-document:

```
<script src="https://maken.wikiwijs.nl/js/mijnscrip.js"></script>
```

In de afbeelding staat een voorbeeld van de koppeling.

index.html

```
<!DOCTYPE html>
<html lang="nl">
  <head>
    <title>siabloon</title>
    <meta charset="utf-8">
    <script
      src="js/mijnscrip.js">
    </script>
  </head>
  <body>

  </body>
</html>
```

mijnscrip.js

In dit document staat de
JavaScript-code.

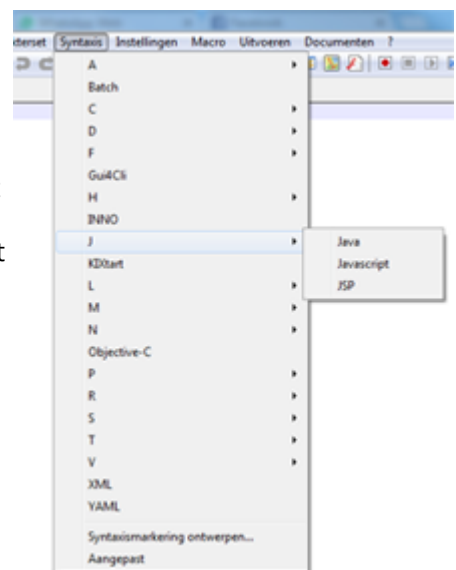
Voor het behoud van het overzicht van onze bestanden hebben we ervoor gekozen mijnscrip.js te bewaren in de map js. Dat is te zien in de regel voor de koppeling van het HTML-document aan de JavaScript-code. De naam van het bestand wordt voorafgegaan door de naam van de map waarin het bestand is opgeslagen.

★ Aan de slag 37

- Maak in de map html waar je tot nu toe al je HTML-bestanden hebt opgeslagen een aparte map js aan.
- Open Notepad++ en klik in de menubalk van het programma op het tabblad Syntaxis. Ga naar de letter J en klik op Javascript. Je hebt nu in Notepad++ aangegeven dat je code gaat schrijven voor een JavaScript-document. Bij het opslaan van het bestand wordt automatisch de extensie .js toegevoegd. Sla het bestand op als mijnscrip.js in de zojuist aangemaakte map js.
We hebben nu een extern JavaScript-bestand met de naam mijnscrip gemaakt. Het bestand bevat nu nog geen code die kan worden uitgevoerd.

Sla het bestand op als **mijnscrip.js** in de zojuist aangemaakte map js.

We hebben nu een extern JavaScript-bestand met de naam mijnscrip gemaakt. Het bestand bevat nu nog geen code die kan worden uitgevoerd.



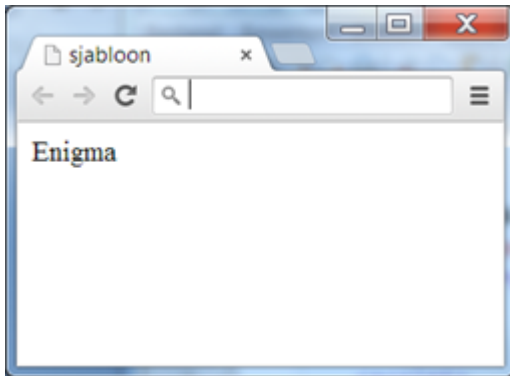
Een eenvoudig script

In de introductie van deze module hebben we beschreven hoe je een sjabloon voor een HTML5-document kan maken. Aan de HTML-code van ons voorbeelddocument gaan we nu JavaScript-code toevoegen. We plaatsen de JavaScript-code uit de tekst hieronder in de body van het HTML-document.

```
<!DOCTYPE html>
<html lang="nl">
<head>
<title>sjabloon</title>
<meta charset="utf-8">
<link rel="stylesheet" href="https://maken.wikiwijs.nl/css/opmaak.css">
</head>
<body>
<script>
/*
Auteur: Enigma-online.nl
Datum: mei 2014-05-30
*/
var tekst="Enigma"; // variabele tekst met inhoud Enigma
document.write(tekst);
</script>
</body>
</html>
```



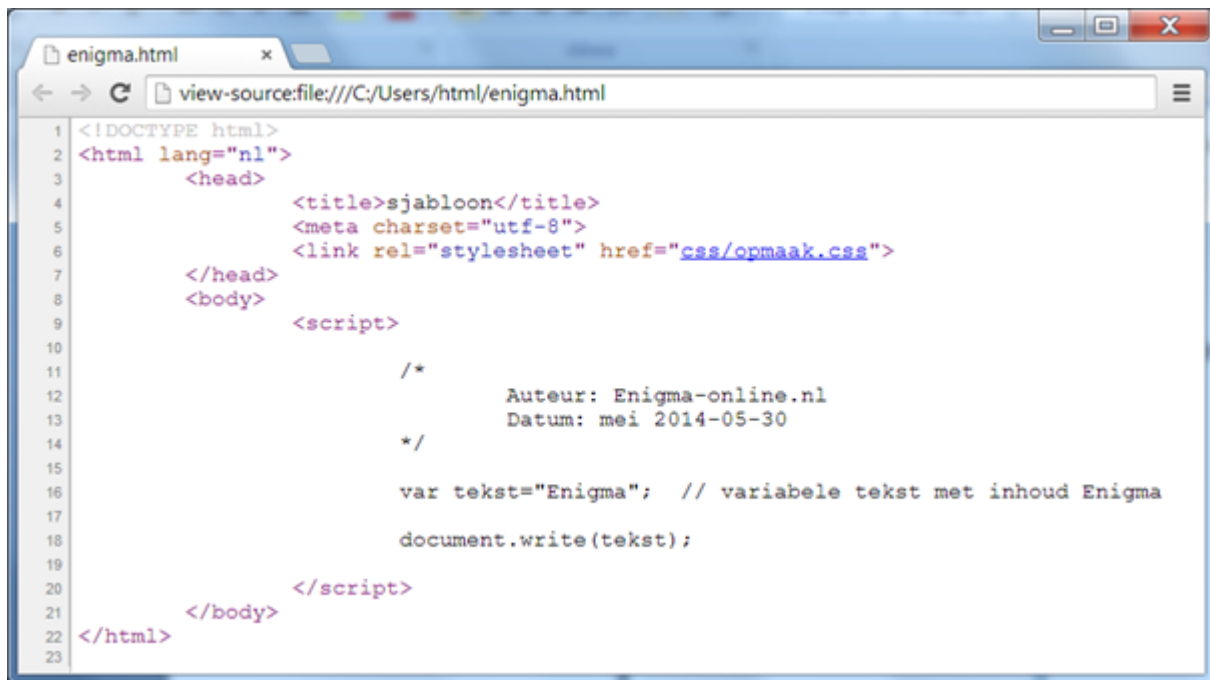
Het HTML-document slaan we op onder de naam enigma.html. Na het openen van dit bestand in een browser zien we als resultaat:



Het script begint met een stukje commentaar. Commentaar over meerdere regels staat tussen `/*` en `*/`. Verderop in de code is er aan één enkele regel commentaar toegevoegd. Commentaar voor een regel begint met `//`. Na het eerste stukje commentaar volgt een stukje code waarin een variabele met de naam `tekst` wordt aangemaakt. De waarde die de variabele toegekend krijgt, is `Enigma`. De regel `document.write(tekst);` zorgt er voor dat bij het uitvoeren van de code de inhoud van de variabele `tekst` op het scherm wordt getoond.

In een browser kan de broncode van een HTML-pagina worden bekeken. Op een Windows-machine klik je daarvoor rechts in een browser waarin een HTML-pagina is geopend. In het snelmenu dat dan verschijnt kun je kiezen voor de optie `Paginabron` weergeven. Omdat JavaScript een client-side scriptingtaal is, wordt de code van het script bij het bekijken van de broncode volledig afgebeeld. Bij server-side scriptingtalen is dat niet het geval.

In de afbeelding hieronder staat een afbeelding van onze code als brondcode.



```
1 <!DOCTYPE html>
2 <html lang="nl">
3   <head>
4     <title>sjabloon</title>
5     <meta charset="utf-8">
6     <link rel="stylesheet" href="css/opmaak.css">
7   </head>
8   <body>
9     <script>
10
11       /*
12          Auteur: Enigma-online.nl
13          Datum: mei 2014-05-30
14        */
15
16       var tekst="Enigma"; // variabele tekst met inhoud Enigma
17
18       document.write(tekst);
19
20     </script>
21   </body>
22 </html>
23
```

Voor het gemak zetten we nog even een paar belangrijke regels voor het schrijven van code op een rijtje:

- Achter elke regel komt een ; (puntkomma), behalve bij for/while-lussen en if- statements
- Commentaar voor een regel begint met //
- Commentaar over meerdere regels staat tussen /* en */
- Een haakje openen (heeft altijd een haakje sluiten)
- Aanhalingstekens komen in paren " " en ' '

Veel fouten kun je voorkomen door je aan deze regels te houden.

★ Aan de slag 38

Onze editor kan ook gebruikt worden voor het uitvoeren van JavaScript-code.

Open in de editor het bestand **template.html**. Schrijf voor de body van het HTML-document een stukje JavaScript-code. Declareer 2 variabelen. Deze variabelen krijgen de naam `getal1` en `getal2`. Ken aan beide variabelen achtereenvolgens de waarden 25 en 50 toe. Declareer daarna een variabele `getal3`. Deze variabele krijgt als waarde de optelsom van de waarden van `getal1` en `getal2`. Toon daarna de waarde van `getal3` op het scherm. Klik na het schrijven van de code op de knop "Uitvoeren" en bekijk het resultaat.

Editor

HTML5



★ Aan de slag 39

Open in de editor het bestand **variabelen_1.html**.

Dit bestand is onderdeel van het zip-bestand dat je al eerder hebt moeten downloaden en hebt uitgepakt in je home-directory in de map `html`. Verwijder achter `var getal2 = 50;` de puntkomma. Verwerk de code en bekijk het resultaat. Wat zie je?

Editor

HTML5



★ Aan de slag 40

Open in een browser het bestand [variabelen_1.html](#)
Bekijk de broncode van het bestand. Is de JavaScript-code zichtbaar?

Welke conclusie kun je trekken op grond van het antwoord dat je hebt gegeven?



★ Aan de slag 41

Open in Notepad++ de bestanden [variabelen_1.html](#) en [mijnscrip.js](#).

Voor [variabelen_1.html](#) heb je een stukje JavaScript-code geschreven in de body van het HTML-document. Neem deze code over in het bestand [mijnscrip.js](#).

Je kopieert daarvoor de code die staat tussen de tags `<script>` en `</script>`.

Sla het bestand op onder dezelfde naam. Sluit daarna het bestand

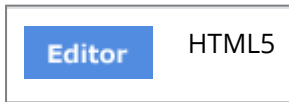
[variabelen_1.html](#). Open nu in Notepad++ het bestand [template.html](#). Koppel het bestand [mijnscrip.js](#) aan het bestand [template.html](#) en sla het bestand [template.html](#) op als [variabelen_2.html](#).

Bekijk het resultaat in een browser.



★ Aan de slag 42

Open in de editor de bestanden `variabelen_2.html` en `mijnscrip.js`. Verwerk daarna de code en bekijk het resultaat. Wat is anders in vergelijking met de uitvoer van de vorige opdracht waar je exact hetzelfde hebt gedaan, maar dan in Notepad++ en een browser?



Commando's

Statements

In JavaScript worden opdrachten statements genoemd. In HTML zijn de statements die met JavaScript geschreven zijn te vergelijken met commando's die door een browser worden uitgevoerd. De regel `document.write("Hello World");` vertelt de browser dat op het scherm de tekst HelloWord moet worden afgebeeld. Statements worden altijd afgesloten met een puntkomma.

Variabelen

In statements kunnen variabelen voorkomen. Een variabele heeft een naam en een inhoud. In een variabele kunnen in JavaScript diverse typen gegevens worden opgeslagen. De inhoud (waarde) van een variabele kan bijvoorbeeld bestaan uit een getal (number) maar ook uit een stukje tekst (string). Kijk maar eens naar de volgende voorbeelden:
`var naam = "Alan Turing";`
`var geboortedatum = 1912;`



In de eerste regel declareren we een variabele naam. Het declareren van een variabele in JavaScript houdt in dat je een variabele aanmaakt. Aan de variabele wordt een naam gegeven. Voor het declareren van een variabele wordt in JavaScript het woordje `var` gebruikt.

De code `var naam` kan dus worden uitgelegd als we maken een variabele met de naam "naam". Vervolgens kennen we aan deze de variabele naam de waarde Alan Turing toe. De waarde bestaat in dit voorbeeld uit een stukje tekst en wordt daarom tussen dubbele aanhalingstekens geplaatst. Voor de toekenning (assignment) wordt het `=` teken gebruikt.

In de tweede regel declareren we een variabele met de naam `geboortedatum`. Aan deze variabele kennen we het jaartal 1912 toe. Omdat we hier met een jaartal te maken hebben, staat de waarde van deze variabele niet tussen dubbele aanhalingstekens.

In JavaScript hoeft je niet aan te geven welke inhoud de variabele heeft. Ook hoeft je niet gelijk de inhoud mee te geven. Je mag dus zeggen:

```
var b,c; // Declareer de variabelen b en c
```

Let op de regel //Declareer de variabelen b en c. De JavaScript compiler ziet dit als tekst en doet er niets mee.

Voor het omschrijven van de variabelen in JavaScript zijn diverse notaties toegestaan:

```
var a,b,c;  
var tekst = "enigma-online.nl", d = "test";  
var g=6, f="zes";
```

Zelfs het veranderen van inhoud van getal naar tekst is zonder problemen mogelijk.

```
var g = 6;  
g = "zes";
```

Functies

Een functie is een stukje code dat ontworpen is om een bepaalde taak uit te voeren. Die taak kan telkens opnieuw worden uitgevoerd door de functie aan te roepen.

We kijken naar de functie toonNaam():

```
function toonNaam()  
{  
  var naam = "Alan Turing";  
  document.write(naam);  
}
```

De naam van de functie wordt voorafgegaan door het woordje function.

In JavaScript wordt dit een keyword genoemd. Het woordje var is ook een voorbeeld van een keyword. Keywords hebben een speciale betekenis. Ze mogen niet gebruikt worden als naam voor bijvoorbeeld een variabele of een functie.

Je mag dus niet zeggen var var = "Alan Turing"; Achter de naam van de functie staan twee haakjes.

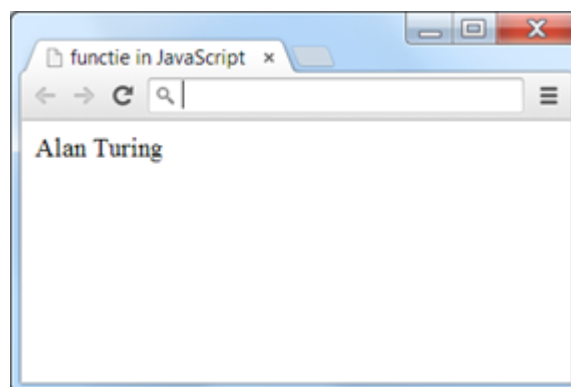
Daaraan herken je dat je met een functie te maken hebt.

De code die door het aanroepen van de functie moet worden uitgevoerd, staat altijd tussen accolades.

De functie toonNaam() declareert een variabele naam en kent daar de waarde "Alan Turing" aan toe.

Daarna wordt de waarde van deze variabele op het scherm getoond. We zouden de code voor de functie toonNaam() kunnen plaatsen in het head-gedeelte van een HTML-document om vervolgens in de body van het HTML-document de functie aan te roepen. In de onderstaande popup is de functie uitgewerkt.

```
<!DOCTYPE html>  
<html lang="nl">  
<head>  
<title>functie in JavaScript</title>  
<meta charset="utf-8">  
<script>  
function toonNaam()  
{  
  var naam = "Alan Turing";  
  document.write(naam);  
}  
</script>  
</head>  
<body>  
<script>  
toonnaam();  
</script>  
</body>  
</html>
```



In een functie kunnen waarden worden vermeld. Deze waarden worden parameters of argumenten

genoemd. In een functiedefinitie noemen we de waarde die tussen de haakjes van de functie kan worden opgenomen de parameter van die functie. Bijvoorbeeld:

```
function <functienaam>(parameter)
{
... code die moet worden uitgevoerd ...
}
```

Als er in een functiedefinitie meerdere parameters voorkomen worden die gescheiden door komma's.

```
function <functienaam>(parameter1, parameter2)
{
... code die moet worden uitgevoerd ...
}
```

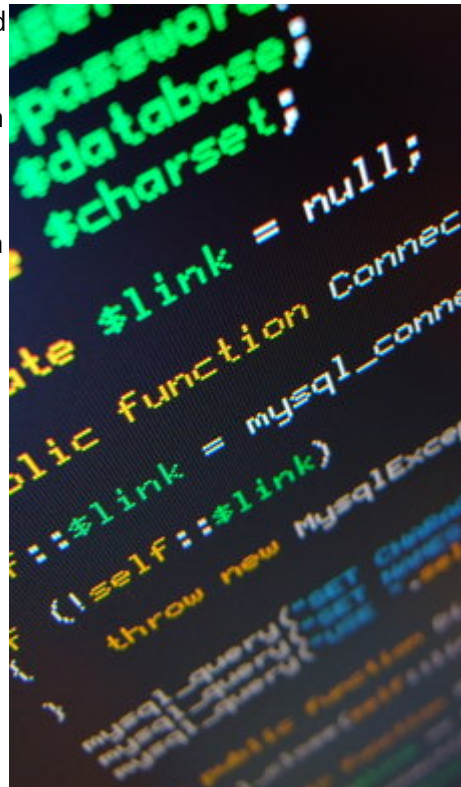
Een methode kan nul, één of meerdere argumenten hebben. Argumenten zijn de waarden die aan een functie worden meegegeven als de functie wordt aangeroepen. Zo heeft de functie som 2 argumenten. Namelijk a en b. Binnen de functie kunnen deze argumenten gebruikt worden als lokale variabelen.

```
function som(a, b)
{
var c;
return a+b;
}
```

★ Aan de slag 43

Open in Notepad++ het bestand [template.html](#). Koppel het bestand [template.html](#) aan een nieuw te schrijven bestand [mijnfuncties.js](#). Sla daarna het bestand [template.html](#) op als [toonnaam.html](#). Open in Notepad++ een nieuw bestand en sla dat op als [mijnfuncties.js](#) in de map js.

Neem de code voor de functie toonNaam() over. De code van de functie toonNaam() wordt besproken bij het onderdeel functies. Sla daarna het bestand op onder dezelfde naam. Roep de functie aan in de broncode van [toonnaam.html](#). Sla het bestand op en bekijk het resultaat in een browser.



★ Aan de slag 44

We gaan een functie tellen schrijven. In deze functie maken we gebruik van een for-statement.

Een for-statement bestaat altijd uit 3 onderdelen. We beginnen met het declareren van een variabele en kennen aan deze variabele een waarde toe, daarna volgt de voorwaarde waaraan moet worden voldaan en tot slot volgt een zogenaamde stapopdracht. Voor de te programmeren functie tellen is het for-statement:

```
for(teller = 1;teller <= 10;teller = teller + 1)
```

```
{  
Hier staat de code die moet worden uitgevoerd als aan de  
voorwaarde wordt voldaan  
}
```

We declareren een variabele teller. Deze variabele teller krijgt de waarde 1.

Als de teller kleiner of gelijk is aan 10 wordt de code tussen de accolades uitgevoerd.

Daarna wordt de teller met 1 verhoogd.

Open in Notepad++ het bestand **template.html**. Koppel het aan het bestand **mijnfuncties.js**. Roep in de body van het HTML-document de functie tellen() aan.

Sla daarna het bestand **template.html** op als **tellen.html** Open in Notepad++ het bestand **mijnfuncties.js**.

In het bestand **mijnfuncties.js** komt de programmacode voor de functie tellen() te staan. Schrijf de code voor de functie tellen. Gebruik de regel

"document.write("
");" voor het regelen van een harde return.

Sla het bestand op onder dezelfde naam. Open vervolgens het bestand **tellen.html** en bekijk het resultaat.

```
for(teller = 1;teller <= 10;teller = teller + 1)
```

```
{  
...  
document.write("<br>");  
}
```

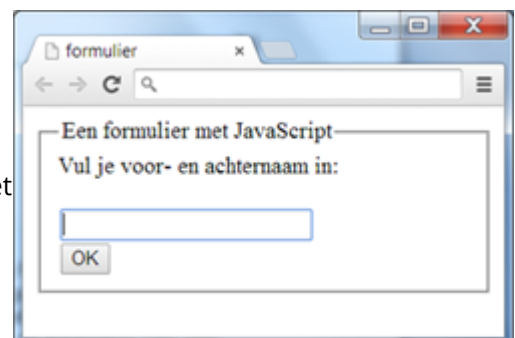


Popup schermen

In de inleiding van deze module hebben we vermeld dat JavaScript een programmeertaal is waarmee webpagina's interactief gemaakt kunnen worden. Die interactie is goed zichtbaar als er op een website gebruik gemaakt wordt van popup's. Een popup is een apart schermpje dat verschijnt bij het bekijken van bepaalde webpagina's. Om door te kunnen gaan moet de bezoeker altijd op een OK-knop klikken.

In JavaScript worden 3 verschillende popup-schermen gebruikt:

- Alert Box
- Confirm Box
- Prompt Box



Voor het gebruik van deze popup's worden achtereenvolgens de functies `window.alert("...")`, `window.confirm("...")` en `window.prompt("...", "standaardtekst")` gebruikt. Bij al deze functies mag de prefix `window` worden weggelaten.

In de tekst hieronder zie je een voorbeeld van een uitwerking voor een Alert Box:

```
<!DOCTYPE html>
<html lang="nl">
<head>
<title>alert</title>
<meta charset="utf-8">
</head>
<body>
<script>
alert ("Met Enigma heeft
informatica geen geheimen
meer");
</script>
</body>
</html>
```

In JavaScript kan een Alert Box ook in een eenvoudig formulier worden verwerkt. Met een formulier kunnen gegevens van de bezoeker van een website verzameld worden. Formulieren maak je in HTML. Dat is, zoals we al eerder hebben aangegeven, een taal waarmee de structuur van een webpagina kan worden beschreven. Net als CSS is HTML géén programmeertaal.

Voor de afhandeling van de gegevens die worden verkregen via een formulier is een programma of script nodig. Vaak is dat PHP, maar ook met JavaScript kunnen de gegevens van een formulier verwerkt worden. We laten hier eerst zien hoe een formulier in HTML-code gemaakt kan worden. Daarna bespreken we hoe JavaScript gebruikt kan worden voor het verwerken van de invoer van een formulier. In de module over PHP wordt uitgebreid stilgestaan bij het verwerken van gegevens uit een formulier met PHP.

In HTML wordt voor een formulier het element `form` gebruikt. Voor het indelen van een formulier kan het element `fieldset` gebruikt worden. Het formulier ziet er dan uit als een rechthoek. Met het element `legend`, dat onderdeel is van het element `fieldset`, kun je een naam geven aan de `fieldset`.

Hieronder staat een voorbeeld van een uitgewerkt formulier.

```
<!DOCTYPE html>
<html lang="nl">
<head>
<title>formulier</title>
<meta charset="utf-8">
<link rel="stylesheet" href="https://maken.wikiwijs.nl/css/opmaak.css">
<script>
function toonNaam()
{
if(document.form1.naam.value == "")
{
alert("Je bent vergeten iets in te vullen");
}
else
{
alert(document.form1.naam.value);
}
}
</script>
</head>
<body>
<form name="form1">
```

```

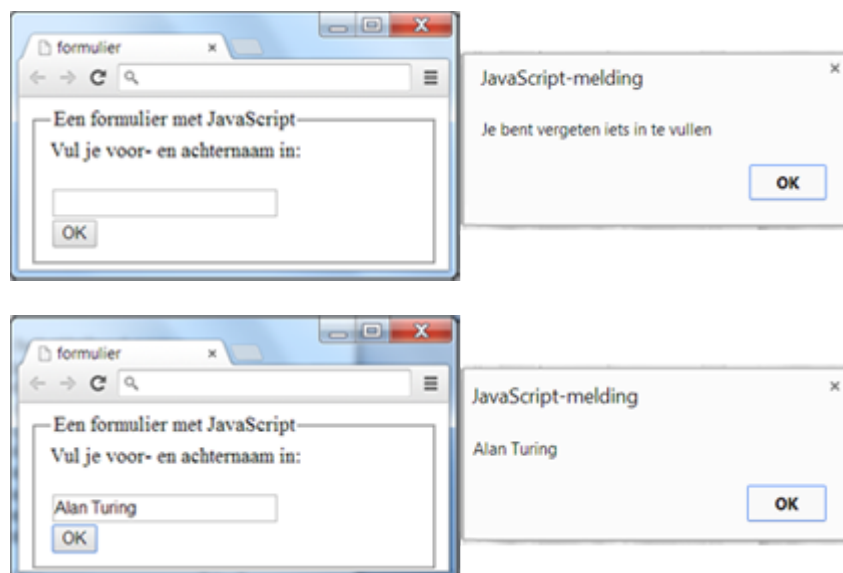
<fieldset>
<legend>Een formulier met JavaScript</legend>
Vul je voor- en achternaam in:<br><br>
<input type = "text" size= "25" name = "naam"><br>
<input type = "button" value = "OK" onClick = "toonNaam()" >
</fieldset>
</form>
</body>
</html>

```

In de broncode van het formulier is te zien dat het element form het attribuut name heeft. Dit attribuut kan gebruikt worden om het formulier een eigen gekozen naam te geven. Standaard is de waarde van dit attribuut "form1". Het belangrijkste element van een formulier is het input-element. Daarmee kan informatie van de bezoeker van een website worden verkregen.

Hoe het input-element eruit ziet, hangt af van het type attribuut dat aan het input-element is gekoppeld. In ons voorbeeld hebben we gebruik gemaakt van een tekstvak. Door te klikken op de knop met het label "OK" wordt de functie toonNaam() uitgevoerd. Bij het uitvoeren van deze functie wordt gecontroleerd of er iets in het tekstvak is ingevuld. Is dat niet het geval dan zal verschijnen er een venster met de melding "Je bent vergeten iets in te vullen".

Is er wel tekst ingevuld, dan verschijnt de ingevulde tekst in een nieuw venster. In de afbeeldingen hieronder staan voorbeelden van beide situaties.



★ Aan de slag 45

Open in Notepad++ het bestand `template.html`.
Koppel het bestand `template.html` aan het bestand `mijnfuncties.js`.
Sla daarna het bestand `template.html` op als `formulier.html`.
Schrijf in de body van het HTML-document de code voor een
formulier waarin het mogelijk is om je geboortedatum in te vullen.

Door te klikken op de OK-knop van het formulier wordt de functie
toonGeboortedatum aangeroepen.
De JavaScript-code voor deze functie schrijf je in `mijnfuncties.js`.
Test of het formulier werkt.

★ Aan de slag 46

Open in Notepad++ het
bestand `template.html`
Schrijf in de body van het
HTML-document een stukje
JavaScript=code voor een
prompt box. In het popup-
scherm staat de tekst "Vul je
naam in".
Na het invullen van je eigen
naam verschijn na het klikken
op de OK-knop op het scherm
de tekst "Mijn naam is"
gevolgd door je eigen naam.
Sla het bestand op als
`popup.html` en test de code in
een browser.



HTML5 en JavaScript

JavaScript wordt beschouwd als de programmeertaal van het World Wide Web.

We hebben al eerder vermeld dat in HTML5-pagina's vaak gebruik gemaakt wordt van deze scriptingtaal. In deze paragraaf laten we als afronding van de module een paar voorbeelden zien van wat er met JavaScript zoal mogelijk is.

Canvas

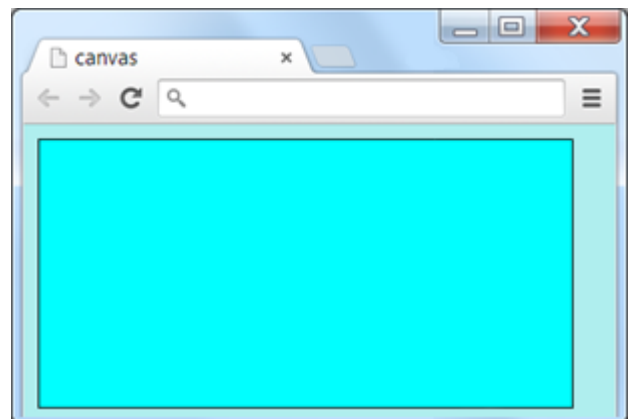
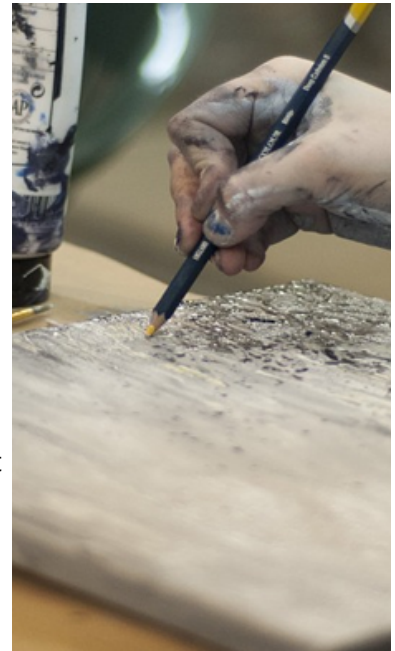
In HTML5 wordt het element canvas gebruikt om te tekenen. De naam canvas refereert naar het doek dat schilders gebruiken om olieverfschilderijen op te maken. Dat doek is gemaakt van canvas. Dat is stof die gemaakt is van vlas, hennep, jute of katoen.

In HTML5 is de canvas het gebied waarop getekend kan worden. Om echt te kunnen tekenen, zijn scripts nodig. Daarvoor wordt onder andere JavaScript gebruikt. Standaard heeft het element canvas geen rand en inhoud. De attributen width en height geven de afmetingen van het canvas aan. In HTML noteer je bijvoorbeeld `<canvas width="300" height="150"></canvas>`.

Aan de begintag voegen we het attribuut id toe:

```
<canvas id="mijnCanvas" width="300" height="150"></canvas>
```

Het vermelden van het attribuut id is nodig omdat we in de JavaScript-code gaan verwijzen naar deze id. We voegen aan de code van het HTML-voorbeelddocument dat we telkens in deze module hebben gebruikt een canvas toe. De opmaak van de webpagina regelen we voor het gemak even in een aparte stylesheet canvas.css



HTML-code:

```
<!DOCTYPE html>
<html lang="nl">
<head>
<title>canvas</title>
<meta charset="utf-8">
<link rel="stylesheet" href="https://maken.wikiwijs.nl/css/canvas.css">
</head>
<body>
<canvas id="mijnCanvas" width="300" height="150"></canvas>
</body>
</html>
```

CSS-code:

```
body
{
color: black;
background: #afeeee;
}
canvas
{
border: 1px black solid;
background: #00ffff;
}
```

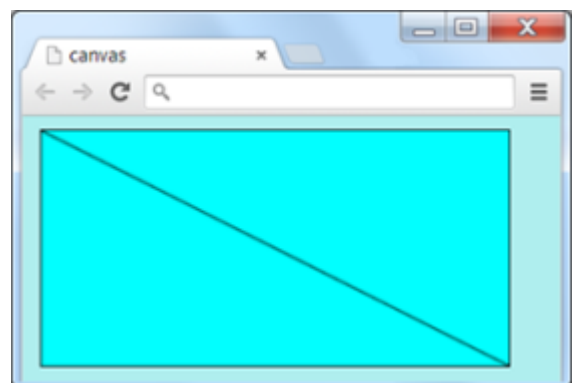
De rechthoek is het gebied waarop we kunnen gaan tekenen. De afmetingen van de tekening worden aangegeven in pixels. Dat geldt ook voor het aangeven van de positie van de tekening op het canvas. Daarvoor wordt een horizontale en verticale nummering gebruikt. Een plaatsbepaling bestaat dus uit twee getallen. Het punt in de linkerbovenhoek van de canvas is het punt (0,0). In de popup heeft het punt in de rechteronderhoek de coördinaten (300, 150).

We gaan nu een lijn op het canvas tekenen die begint op positie (0,0) en eindigt op positie (300, 150). Daarvoor gebruiken we JavaScript:

```
var c = document.getElementById("mijnCanvas");
var ctx = c.getContext("2d");
ctx.moveTo(0,0);
ctx.lineTo(300,150);
ctx.stroke();
```

De code staat in de body van het gebruikte HTML-document.

```
<body>
<canvas id="mijnCanvas" width="300"
height="150"></canvas>
<script>
var c = document.getElementById("mijnCanvas");
var ctx = c.getContext("2d");
ctx.moveTo(0,0);
ctx.lineTo(300,150);
ctx.stroke();
</script>
</body>
```



In de eerste regel JavaScript-code verwijzen we naar onze canvas met id mijnCanvas. Vervolgens roepen we de functie getContext() aan. Deze functie krijgt de parameter "2d" mee.

Zonder het aanroepen van deze functie is het niet mogelijk om te tekenen. De functie moveTo() geeft een punt op het canvas aan vanwaar we willen gaan tekenen. De functie.lineTo() geeft het punt aan waar de lijn naar toe moet lopen. Voor het zichtbaar maken van de lijn gebruiken we de functie stroke()

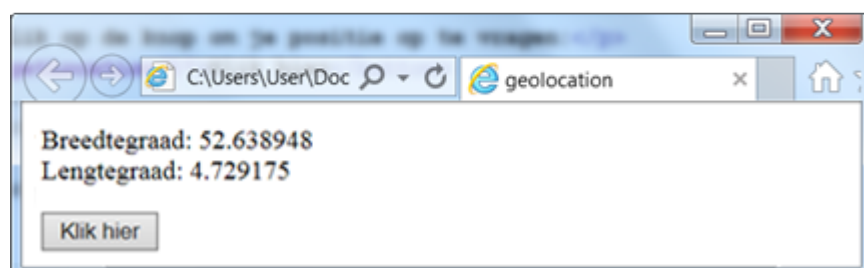
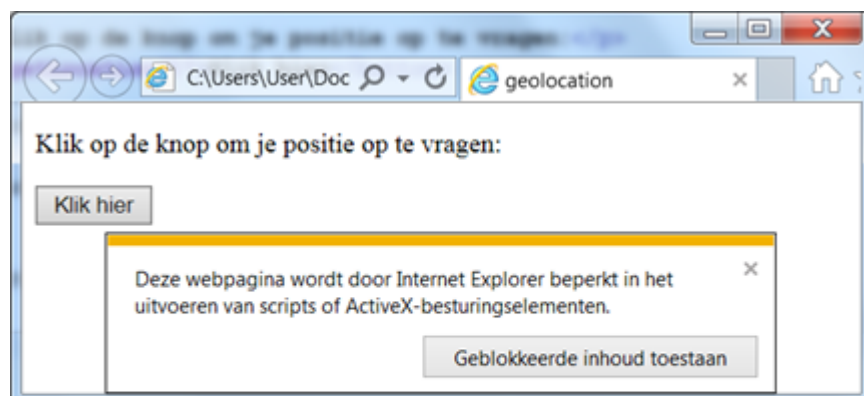
Geolocatie

Geolocation wordt in HTML5 gebruikt om te lokaliseren waar iemand zich bevindt. Voor het aanduiden van plaatsen op aarde en het berekenen van onderlinge afstanden worden lengte- en breedtegraden gebruikt. Lengtegraden zijn de denkbeeldige verticale cirkelbogen (meridianen) die vanaf de Noordpool via de evenaar naar de Zuidpool lopen. De breedtegraad van een plek op Aarde is de hoek die de verbindinglijn tussen die plek en het middelpunt van de Aarde met het vlak van de evenaar maakt. De Engelse term voor lengtegraad is longitude. Een breedtegraad wordt latitude genoemd.

Aangezien het opvragen van geografische coördinaten privacygevoelige informatie kan zijn, moet er van tevoren toestemming worden gegeven om dit te mogen doen.

Niet alle browsers ondersteunen geolocation. Bekijk de screenshots hieronder. Hierin zie je wat er

gebeurt als in Internet Explorer gebruik gemaakt wordt van geolocation. De broncode van de HTML-pagina staat eronder.

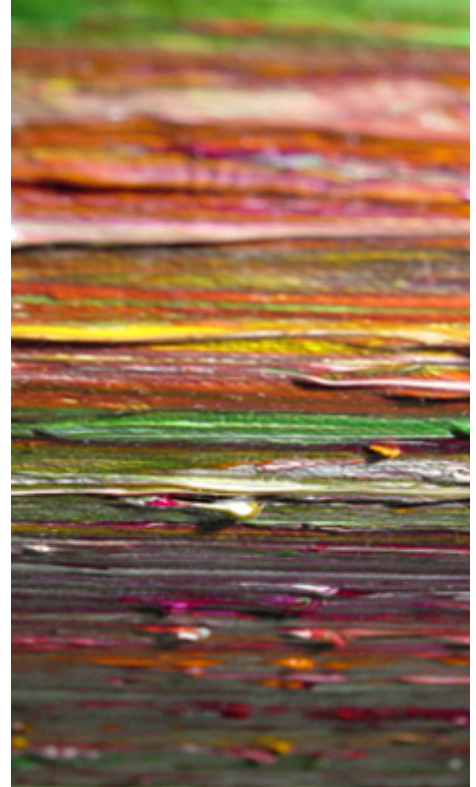


```
<!DOCTYPE html>
<html lang="nl">
<head>
<title>geolocation</title>
<meta charset="utf-8">
</head>
<body>
<p id="positie">Klik op de knop om je positie op te vragen:</p>
<button onclick="getLocation()">Klik hier</button>
<script>
var x = document.getElementById("positie");
function getLocation()
{
navigator.geolocation.getCurrentPosition(showPosition);
}
function showPosition(position)
{
x.innerHTML="Breedtegraad: " +
position.coords.latitude +
"<br>Lengtegraad: " + position.coords.longitude;
}
</script>
```

```
</body>  
</html>
```

★ Aan de slag 47

Open in Notepad++ het bestand [template.html](#)
Schrijf in de body van het HTML-document de code voor een canvas waarin een vierkant wordt getoond.
Sla het bestand op onder de naam [canvas.html](#) en bekijk het resultaat in een browser.



Voortgang

Bekijk hier je voortgang

Voortgang

Kernprogramma

Docent kan klassen aanmaken en leerlingen volgen

Een docent kan op de profielpagina klassen aanmaken. Als een docent dat gedaan heeft, kunnen de leerlignen zich aan de klas koppelen. Als de leerlingen dat gedaan hebben, kan de docent de voortgang van de leerlingen volgen.

>>



Over dit lesmateriaal

Colofon

Auteurs	Linda B ; content VO-
Team	Wikiwijs Maken Auteurs
Laatst gewijzigd	9 juli 2019 om 08:05
Licentie	De Internationale Creative Commons 4.0 licentie waarbij de gebruiker het werk mag kopiëren, verspreiden en doorgeven en afgeleide werken mag maken onder de voorwaarden: Naamsvermelding en Gelijk Delen, zie http://creativecommons.org/licenses/by-sa/4.0/ . Meer informatie over de CC Naamsvermelding-GelijkDelen 4.0 Internationale licentie licentie.

Aanvullende informatie over dit lesmateriaal

Van dit lesmateriaal is de volgende aanvullende informatie beschikbaar:

Leerniveaus	HAVO 4, VWO 6, HAVO 5, VWO 4, VWO 5
Leerinhoud en doelen	Informatica
Eindgebruiker	leerling/student
Moeilijkheidsgraad	gemiddeld
Trefwoorden	arrangeerbaar, arrangeerbare, leerlijn, rearrangeerbare