

## Les 2: The WWW – CSS(3)

### Denkvragen

1. Zoals beschreven in de slides biedt CSS3 een groot aantal selectors aan. Zoek in de CSS3 selectors specificatie (<http://www.w3.org/TR/css3-selectors/>) op waarvoor de attribute selector dient, en geef minstens twee voorbeelden van goede toepassingen voor deze selector. De class selector kan ook gezien worden als een attribuut selector. Hoe zit de syntax dan in elkaar? Zoek dit op in de specificatie.
2. Waarom is het gebruik van een CSS reset kit nodig bij het maken van een website? Wat zou er gebeuren indien je deze niet zou gebruiken? Zoek naar informatie over de default styles die toegepast worden door bv. firefox of chrome en zoek tenminste 2 verschillen hierin.
3. Gebruik de Chrome Web Inspector om uit te zoeken hoe de CSS stijl van een link in de inhoudstabel van een artikel (bv. het artikel <http://nl.wikipedia.org/wiki/Wikipedia>) wordt berekend. Tracht dit te koppelen aan de regels die je kent voor specificity voor selectoren en de regels voor cascading in het algemeen.
4. Zoek op in de officiële specificatie wat het effect van de “!important” declaratie in CSS stylesheets is. Hoe kan deze declaratie gebruikt worden door slechtzienden om hun gebruikerservaring te verbeteren?
5. Zoek op wat het verschil is tussen een specified, een computed en een actual value in CSS. Welke van deze waarden veranderen wanneer je met de webbrowser inzoomt op een webpagina?
6. Gebruik de CSS validator op <http://jigsaw.w3.org/css-validator/> om de CSS definitie van een aantal populaire websites te valideren. Zijn er bepaalde warnings/errors die je vaak tegenkomt?

## Programmeeropdrachten

1. Maak een HTML document waarin een geneste lijst met muziekgenres weergegeven wordt. Het buitenste niveau van de lijst bevat drie elementen: rock, elektronica en klassiek. De rock-lijst bevat als elementen hard rock, metal en psychedelisch rock. De elektronica-lijst bevat als elementen ambient, chiptune en drum and bass. De lijst met klassieke muziek bevat tenslotte als elementen barok en romantiek. Ken aan elke sublijst minstens drie artiesten toe.  
De buitenste lijst maakt gebruik van Romeinse cijfers als opsommingstekens, de tweede lijst gebruikt hoofdletters, en voor de binnenste opsomming gebruik je getallen. De achtergrond van de lijst met rockmuziek is rood, die van de lijst met elektronica is zwart.  
Behalve de lijst van muziekgenres bevat deze pagina ook een afbeelding die rechts van de tabel wordt weergegeven.  
Maak van dit document drie versies: één waarin je inline style sheets gebruikt, één waarin de CSS apart in het HTML document staat, en één waarbij je met een extern CSS bestand werkt. Wat zijn de voor- en nadelen van deze verschillende methodes?
2. Maak een HTML document met een ongeordende lijst van films. Geef elke film als opsommingsteken een afbeelding van de filmposter.
3. Maak een persoonlijk CV. Alle layout-kenmerken dienen via een externe style sheet gedefinieerd te worden, zodat ze eenvoudig aanpasbaar zijn. Gebruik zo veel mogelijk verschillende element-soorten, om de mogelijkheden van CSS te demonstreren. Overdrijf ook niet: de layout van de pagina dient sober en functioneel te zijn. Vermijd het gebruik van <div>, tenzij er echt geen toepasselijk HTML5 element beschikbaar is. Uiteraard moeten zowel je HTML als je CSS gevalideerd kunnen worden.
4. Maak een webpagina met drie <div>'s die naast elkaar staan. De <div>'s hebben een breedte van respectievelijk 50px, 400px en 80px. Vul de <div>'s op met tekst.  
Vorm nu de layout van de pagina om naar een liquid layout, waarbij de breedte van de verschillende <div>'s wordt aangepast aan de grootte van de browser. Kijk eventueel op <http://www.dynamicdrive.com/style/layouts/category/C13/P0/> voor voorbeelden.
5. Zoek uit hoe de stijl van de website op <http://appscity.be/nl/programma> automatisch wordt aangepast aan de grootte van de browser. Welk CSS bestand wordt gebruikt?  
Bij bepaalde schermgrootte's zit de layout niet helemaal goed. Gebruik de Chrome Browser Inspector om deze probleempjes op te lossen in de CSS.

## Projectopgave

Met de kennis van HTML en CSS heb je alles wat nodig is om een volledige statische website te maken. Begin echter niet zomaar met het programmeerwerk! Belangrijk is dat je eerst voor jezelf een idee hebt van hoe je wil dat je website eruit ziet, en dat je dit achteraf in code omzet. Zo zorg je ervoor dat de implementatie in functie staat van het design, in plaats van omgekeerd. Je zal ook merken dat sommige keuzes in layout additioneel werk vereisen in de HTML source; denk daarbij aan het toevoegen van <div> elementen. Ga hier wel spaarzaam mee om en gebruik liever de aangewezen HTML5 semantische elementen.

Een aantal delen van je website worden (uiteindelijk) ingevuld op basis van informatie uit de databank. Denk daarbij aan het overzicht van categorieën, de informatie over specifieke evenementen, de kalender, enz. Uiteraard heb je de koppeling met de databank momenteel nog niet operationeel, maar je kan probleemloos hiervoor statische pagina's voorzien met placeholder content. Hierop kan je de styling al meteen toepassen. In de laatste fase zullen we dan deze statische content vervangen door de inhoud die opgevraagd wordt uit de databank – indien je HTML en CSS structuur goed in elkaar zit is het enkel kwestie van de inhoud van de elementen nog te wijzigen!

