**Voorbereiden**

|  |
| --- |
| In dit hoofdstuk leert u de voor- en nadelen kennen van het gebruik van voorgedefinieerde bibliotheken en plug-ins. Verder krijgt u zicht op het gebruik van AJAX-elementen en -procedures.  Dit hoofdstuk bevat online oefening met voorgedefinieerde scripts, waarbij u direct resultaat ziet en zelf wijzigingen kunt aanbrengen in de code. Daarnaast wordt u uitgenodigd zelf oefeningen te maken om het geleerde toe te passen.  **Leerdoelen** Na dit hoofdstuk:   * kunt u de voor- en nadelen omschrijven van het gebruik in JavaScript van voorgedefinieerde bibliotheken en plug-ins als JQuery, Spry, Dojo, MooTools en Prototype * kunt u stappen omschrijven voor het gebruik van bibliotheken (als jQuery) en plug-ins * kunt u fundamentele AJAX-procedures en -elementen omschrijven * kunt de schematische opbouw weergeven van gebruikelijke interacties tussen JavaScript, XML en XHTML * kunt u typische AJAX-requests omschrijven * kunt u alternatieven aangeven voor het op XML-gebaseerd AJAX.   *Ingeschatte tijdsduur: 30 minuten.*  Bovenkant formulier  **Status**  Klik Voltooien als u deze activiteit hebt afgerond.  Onderkant formulier |

**Bestuderen**

|  |
| --- |
| **Bibliotheken (libraries)** U hebt reeds kennisgemaakt met het DOM en een aantal manieren om het DOM via JavaScript te manipuleren. U hebt wellicht ook gezien dat dit af en toe best ingewikkeld kan worden. Bijkomend probleem is dat niet alle browsers op precies dezelfde manier reageren of precies dezelfde voorzieningen ondersteunen. Hoewel de verschillen tussen browsers de laatste jaren zijn afgenomen, kan het voor een website een vereiste zijn goed te functioneren in een grote hoeveelheid browsers, ook in verouderde maar nog wel gebruikte browsers.  Om deze problemen op te lossen, zijn er zogeheten *libraries*(bibliotheken of frameworks) geschreven. Een library bestaat in het algemeen uit voorzieningen in de vorm objecten, properties en methoden die de JavaScript-ontwikkelaar kan gebruiken en die hem veel werk kunnen besparen.  Een veelgebruikte library is jQuery (ontstaan in 2006). Andere frameworks zijn Prototype, MooTools, Dojo en Spry.  In relatie met frameworks komt u ook vaak de term API tegen. API is een afkorting voor Application Programming Interface. Het is een algemene term voor de verzameling van properties en methoden die u in uw eigen programma kunt gebruiken.  Zo kunnen we ook spreken over de DOM-API als het gaat over methoden als getElementById() of een property als innerHTML.  **jQuery** jQuery is op dit moment wellicht het meest gebruikte framework voor JavaScript. Om jQuery te kunnen gebruiken, hoeft u niet veel anders te doen dan de library in een script-tag in uw pagina op te nemen. U kunt jQuery downloaden op uw eigen computer, maar dat hoeft niet, want er zijn zogeheten CDN's (Content Delivery Network) die de library online hebben. [Hier](http://www.w3schools.com/jquery/jquery_install.asp) vindt u de voordelen van het gebruik van een CDN.  Bestudeer tevens [deze voorbeelden](http://www.w3schools.com/js/js_lib_jquery.asp).  De belangrijkste functie in jQuery is de functie die aangegeven wordt met het dollarteken. Deze functie heet voluit jQuery(), maar wordt meestal geschreven als $().  *Ingeschatte tijdsduur: 70 minuten.*  Bovenkant formulier  **Status**  Klik Voltooien als u deze activiteit hebt afgerond.    Onderkant formulier |

**Oefenen**

|  |
| --- |
| In de volgende oefenopgave past u de zojuist geleerde kennis toe. Het is erg belangrijk de oefenopgave eerst zelf te maken, voordat u de uitwerking bekijkt.  Maak opgave 1, de opgave vindt u [hier](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20oefenopgaven/Oefenopgaven_hoofdstuk_6.pdf).  Controleer uzelf aan de hand van [deze uitwerkingen](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20oefenopgaven/Uitwerkingen_oefenopgaven_H6.pdf).  *Ingeschatte tijdsduur: 30 minuten.*  Bovenkant formulier  **Status**  Klik Voltooien als u deze activiteit hebt afgerond.  Onderkant formulier |

**Bestuderen**

|  |
| --- |
| Een belangrijke eigenschap die u in de voorbeelden ziet, is *chaining*(het achter elkaar uitvoeren van verschillende methoden op hetzelfde object (zie ook Method Chaining in paragraaf 8.2.2 van Flanagan)).  De objecten waar jQuery mee werkt zijn jQuery-objecten, geen HTML- of DOM-objecten. U kunt dus geen DOM-methode uit de voorgaande leerstof toepassen op een jQuery-object.  Een voorbeeld is de aanroep $("#h01"). Hiermee wordt het element met id h01doorgegeven aan jQuery. Dat levert als resultaat een jQuery-object waarop u alle jQuery-methoden kunt loslaten. Het jQuery-object is als het ware een verpakking om het HTML-object. Deze verpakking heeft voorzieningen om het HTML-object dat erin verstopt zit te veranderen.  Een ander voorbeeld is de aanroep $(document)of $(window). Hiermee wordt het document of window als jQuery-object afgeleverd, zodat u vervolgens de jQuery-methoden op dit object kunt toepassen.  **Prototype** Een andere framework is Prototype. Ook hiermee wordt DOM-programmeren eenvoudiger. Prototype heeft een weer iets andere API dan jQuery. Bekijk [hier](http://www.w3schools.com/js/js_lib_prototype.asp) de voorbeelden.  **Versies van jQuery** Sinds 2006 bestaan er verschillende versies van jQuery. Er is een onderscheid tussen de versies 1.x en 2.x. De versies uit de 1.x-reeks werken ook met oudere browsers. De versies uit de 2.x-reeks zijn bedoeld voor moderne omgevingen, bijvoorbeeld een intranet met alleen recente browsers of voor mobiele apparaten. Het voordeel van de 2.x-reeks is dat de code compacter en sneller is. Zowel de 1.x- als de 2.x-reeks blijven ondersteund. Lees meer op [Wikipedia - JQuery](http://nl.wikipedia.org/wiki/JQuery" \t "_blank).  *Ingeschatte tijdsduur: 60 minuten.*  Bovenkant formulier  **Status**  Onderkant formulier |

**Oefenen**

|  |
| --- |
| In de volgende oefenopgave past u de zojuist geleerde kennis toe. Het is erg belangrijk de oefenopgave eerst zelf te maken, voordat u de uitwerking bekijkt.  Maak opgave 2, de opgave vindt u [hier](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20oefenopgaven/Oefenopgaven_hoofdstuk_6.pdf).  Controleer uzelf aan de hand van [deze uitwerkingen](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20oefenopgaven/Uitwerkingen_oefenopgaven_H6.pdf).  *Ingeschatte tijdsduur: 20 minuten.*  Bovenkant formulier  **Status**  Klik Voltooien als u deze activiteit hebt afgerond.  Onderkant formulier |

**Bestuderen**

|  |
| --- |
| Naast de jQuery-standaardbibliotheek zijn er aanvullende libraries ontstaan, zoals jQuery UI (User Interface) om geavanceerde gebruikersinterfaces te maken en jQuery Mobile voor mobiele apparaten die door hun kleinere scherm en andersoortige bediening speciale voorzieningen behoeven.  Behalve deze extra bibliotheken zijn er ook zogeheten [plugins](http://learn.jquery.com/plugins/" \t "_blank), kleine aanvullingen op de standaardbibliotheek die het leven van de programmeur kunnen veraangenamen.  **Selector** Om elementen op een pagina te selecteren (om deze vervolgens te bewerken), maakt de jQuery-functie gebruik van *selectors.*Lees de informatie op [w3schools.com - jquery\_syntax](http://www.w3schools.com/jquery/jquery_syntax.asp) en [w3schools.com - jquery\_selectors](http://www.w3schools.com/jquery/jquery_selectors.asp).  Er zijn verschillende soorten selectors:   * Selectors gebaseerd op de naam van de tag, bijvoorbeeld $("p"). * Selectors gebaseerd op de id van een element, bijvoorbeeld $("#demo"). * Een css-selector.   Een mooie demonstratie vindt u [hier](http://www.w3schools.com/jquery/trysel.asp).  **Event-afhandeling** De syntax voor event-handling wordt met jQuery vrij eenvoudig. De voorbeelden op w3schools maken allemaal gebruik van de onload-event, dat in jQuery het document-ready-event heet. Dit garandeert dat het script pas wordt uitgevoerd nadat het document geladen is. Zie onder aan [deze pagina](http://www.w3schools.com/jquery/jquery_syntax.asp).  Merk op dat er verschillende vormen zijn om de code te noteren. De vorm met $(document).ready() is het meest duidelijk.  [Hier](http://www.w3schools.com/jquery/jquery_events.asp) ziet u andere voorbeelden van event-afhandeling.  **Effecten** Met jQuery kunt u een groot aantal effecten toepassen, zoals hide(), show(),fade(), slide() en animate(). Bekijk de voorbeelden op [deze pagina](http://www.w3schools.com/jquery/jquery_hide_show.asp), lees alle volgende pagina's door op Next Chapter te klikken t/m de pagina Chaining.  Merk op dat de scripts met jQuery nogal wat accolades, haakjes en puntkomma's bevatten:  $(document).ready(function(){   $("#hide").click(function(){     $("p").hide();   }); });  Om te begrijpen wat hier staat, is het misschien nuttig de regels wat meer ruimte te geven, de openingshaakjes op een aparte regel te zetten en het overeenkomstige sluithaakje eveneens en recht onder het openingshaakje:  $(document).ready (   function()   {     $("#hide").click     (       function()       {         $("p").hide()       }     );   // einde van click   } );  // einde van ready  **HTM en jQuery** Ook het wijzigen van HTML-elementen gaat met jQuery vrij eenvoudig. Met de methoden text(), html() en val() kunt u de inhoud [opvragen](http://www.w3schools.com/jquery/jquery_dom_get.asp) of [wijzigen](http://www.w3schools.com/jquery/jquery_dom_set.asp).  Bekijk de pagina's voor het [toevoegen](http://www.w3schools.com/jquery/jquery_dom_add.asp) en [verwijderen](http://www.w3schools.com/jquery/jquery_dom_remove.asp) van elementen.  **CSS en jQuery** Lees onderstaande pagina's over het manipuleren van CSS:   * [jQuery – Get and Set CSS Classes.](http://www.w3schools.com/jquery/jquery_css_classes.asp) * [jQuery – css() Method.](http://www.w3schools.com/jquery/jquery_css.asp)   *Ingeschatte tijdsduur: 210 minuten.*  Bovenkant formulier  **Status**  Klik Voltooien als u deze activiteit hebt afgerond.  Onderkant formulier |

**Bestuderen**

|  |
| --- |
| Bestudeer uit Flanagan:   * de inleiding van hoofdstuk 19 (The jQuery Library) * paragraaf 19.1 (jQuery Basics) * paragraaf 19.2 (jQuery Getters and Setters).   *Ingeschatte tijdsduur: 180 minuten.*  Bovenkant formulier  **Status**  Klik Voltooien als u deze activiteit hebt afgerond.  Onderkant formulier |

**Bestuderen**

|  |
| --- |
| AJAX is in 2005 ontstaan. AJAX betekent Asynchronous JavaScript and XML en gaat over een techniek om op de achtergrond gegevens van de server op te halen en deze in een webpagina te plaatsen.  Lees [hier](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20SA/AJAX.pdf) meer over AJAX.  *Ingeschatte tijdsduur: 180 minuten.*  Bovenkant formulier  **Status**  Klik Voltooien als u deze activiteit hebt afgerond.  Onderkant formulier |

**Bestuderen**

|  |
| --- |
| Bestudeer uit Flanagan over Scripted HTML:   * de inleiding van hoofdstuk 18 (Scripted HTTP) * de opmerkingen over *Comet*kunt u overslaan * paragraaf 18.1 (Using XMLHttpRequest).   *Ingeschatte tijdsduur: 240 minuten.*  Bovenkant formulier  **Status**  Klik Voltooien als u deze activiteit hebt afgerond.  Onderkant formulier |

**Bestuderen**

|  |
| --- |
| Meer informatie en een overzicht van de methoden met voorbeelden kunt u vinden op   * [jQuery API](http://api.jquery.com/) * [jQuery Selectors.](http://www.w3schools.com/jquery/jquery_ref_selectors.asp)   *Ingeschatte tijdsduur: 60 minuten.*  Bovenkant formulier  **Status**  Klik Voltooien als u deze activiteit hebt afgerond.  Onderkant formulier |