**1**

**Voorbereiden**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| In dit hoofdstuk leert u over de historie van JavaScript en maakt u kennis met de elementaire principes en mogelijkheden van deze taal. Indien u al eerder geprogrammeerd hebt in een andere taal, zult u waarschijnlijk ontdekken dat er veel overeenkomsten zijn met de taalelementen van JavaScript. U maakt kennis met variabelen en het gebruik van verschillende datatypen en operatoren. U leert vanuit JavaScript op eenvoudige wijze te communiceren met de gebruiker van de webpagina.  Dit hoofdstuk bevat online oefening met voorgedefiniëerde scripts, waarbij u direct resultaat ziet en zelf kleine wijzigingen kunt aanbrengen in de code.  Dit hoofdstuk vormt de grondslag voor het verder ontdekken van de mogelijkheden van JavaScript in de volgende hoofdstukken.  **Leerdoelen** Na dit hoofdstuk:   * kunt u beknopt de geschiedenis van JavaScript beschrijven * kunt u enkele essentiële kenmerken van JavaScript noemen * kent u het verschil tussen client-side en server-side toepassingen * kunt u commentaar toepassen in JavaScript-code * kunt u werken met variabelen en gegevens * kunt u mededelingen voor de gebruiker op het scherm zetten * kunt u rekenkundige operatoren gebruiken * kunt u waarden onderling vergelijken met == en met === * kent u het verschil tussen concatenatie en optellen * kunt u inline scripting toepassen * weet u wat typecasting is * kunt u waarden die de gebruiker invoert in variabelen opslaan * kent u de gereserveerde woorden van JavaScript.   Tijdens de bestudering is het handig, om niet te zeggen noodzakelijk, dat u de voorbeelden van de website van w3schools die in de tekst zijn aangegeven, bestudeert en in deze voorbeelden wijzigingen aanbrengt om het effect daarvan te zien. Probeert u zo veel mogelijk te leren door te doen!  *Ingeschatte tijdsduur: 15 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  **2**  **Bestuderen**   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Eigenschappen van JavaScript** De geboorte van JavaScript vindt plaats in 1995, als scripttaal voor de in die tijd zeer populaire browser Netscape. In Flanagan vindt u in hoofdstuk 1 een kader met de titel *JavaScript: Names and Versions*. Bestudeer deze tekst en onthoud de relatie tussen JavaScript, JScript en ECMAScript.  *Ingeschatte tijdsduur: 30 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  3  **Oefenen**   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Eerste kennismaking met JavaScript** Om een indruk te krijgen hoe JavaScript eruitziet en wat u ermee kunt, is er online een schitterend hulpmiddel, de [JavaScript-tutorial](http://www.w3schools.com/js/default.asp" \t "_blank).  In de volgende oefenopgaven past u de geleerde kennis toe. Het is erg belangrijk de oefenopgaven eerst zelf te maken, voordat u de uitwerkingen bekijkt.  Ga naar de pagina [JS Introduction](http://www.w3schools.com/js/js_intro.asp). Op deze pagina staan voorbeelden die u kunt proberen door op de knop Try it Yourself te klikken. Klik op deze knop bij het eerste voorbeeld. Er opent een nieuw venster dat in tweeën is gesplitst. Links staat de HTML-broncode, rechts de pagina die de browser van deze code maakt.  Maak opgave 1 t/m 3, de opgaven vindt u [hier](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20oefenopgaven/Oefenopgaven_hoofdstuk_2.pdf).  Controleer uzelf aan de hand van [de uitwerkingen](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20oefenopgaven/Uitwerkingen_oefenopgaven_H2.pdf).  *Ingeschatte tijdsduur: 45 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  **Bestuderen**   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Client-server** Het opvragen van een webpagina is een voorbeeld van het client-server mechanisme. Heel simpel verteld, komt het hierop neer: de browser op uw computer is de client. Ergens op internet staat een webserver. Als u het adres van een pagina in uw browser intikt, gaat er een verzoek vanuit uw browser naar de server om de betreffende pagina terug naar uw browser te sturen. De webpagina bestaat in het algemeen uit HTML-code. De browser leest deze code en toont de pagina opgemaakt en wel op het scherm.  Lees hier meer over de [Client-server](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20SA/Client-server.pdf).  *Ingeschatte tijdsduur: 45 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  **Oefenen**   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | In de volgende oefenopgaven past u de zojuist geleerde kennis toe. Het is erg belangrijk de oefenopgaven eerst zelf te maken, voordat u de uitwerkingen bekijkt.  Maak opgave 4 en 5, de opgaven vindt u [hier](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20oefenopgaven/Oefenopgaven_hoofdstuk_2.pdf#page=2).  Controleer uzelf aan de hand van [deze uitwerkingen](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20oefenopgaven/Uitwerkingen_oefenopgaven_H2.pdf#page=2).  *Ingeschatte tijdsduur: 30 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  **Bestuderen**   |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | Programma's werken vaak met talloze hoeveelheden gegevens of waarden. Gegevens zijn er in verschillende soorten: getallen als 1, -32 en 3.14, tekst als "Klik op deze knop" en logische waarden als true en false. Deze soorten heten typen. Omdat ze heel elementair zijn, heten ze primitieve typen.  Veel andere programmeertalen maken onderscheid tussen gehele getallen en getallen met een decimale punt, JavaScript doet dat niet. Alle getallen behoren tot het type *number*. Tekst behoort tot het type *string* en logische waarden tot het type*boolean*.  Andere, meer complexe typen zijn vaak opgebouwd met behulp van primitieve typen. Die complexere typen zijn objecten. Ze behoren tot het type *object*. We zullen verderop in deze module veel voorbeelden daarvan zien. Omdat JavaScript veel met objecten werkt, heet het ook wel een objectgeoriënteerde taal.  Om makkelijk met waarden te kunnen werken, is het handig ze op te slaan in een variabele. Een variabele is technisch gesproken een stukje geheugen waarin u een waarde kunt opbergen. Dat stukje geheugen geeft u een naam, zodat u via die naam kunt refereren aan de waarde die onder die naam is opgeborgen. Een term voor de naam van een variabele is *identifier*. Om een variabele te kunnen gebruiken, moet u hem eerst definiëren; dat heet de declaratie van een variabele.  Bij de declaratie kunt u een variabele meteen een waarde geven, dat heet*initialisatie*. Hier zijn drie voorbeelden van declaraties:   |  |  | | --- | --- | | var x; | // de variabele x is gedeclareerd, // maar heeft nog geen waarde | | x = 3; | // de variabele x krijgt de waarde 3 | | var t = "OK"; | // declaratie van t en initialisatie met "OK" | | var y = x + 5; | // declaratie van y en initialisatie met // de waarde van x+5, dus: y=3+5=8 |   Merk op dat de verschillende declaraties eindigen met een puntkomma, net zoals andere opdrachten (statements) in JavaScript.  **Statement** Een statement is een opdracht in JavaScript. Een statement eindigt meestal met een puntkomma.  Het =-teken waarmee u een variabele een waarde geeft, zoals in var x = 3, is een*operator.* De officiële naam is *assignment operator*(toekenningsoperator). In var x = 3 wordt met behulp van de operator = de waarde 3 aan de variabele x toegekend. U kunt het best de operator = uitspreken als 'wordt' of als 'krijgt de waarde'. Een voorbeeld:   |  |  | | --- | --- | | var x = 3; | // de variabele x krijgt de waarde 3 | | x = x + 1; | // x krijgt de waarde van x+1, dus x=3+1=4 |   De laatste opdracht moet u lezen als: de nieuwe waarde van x wordt gelijk aan de huidige waarde van x plus een.  Kijk [op deze website](http://www.w3schools.com/js/js_variables.asp) voor meer voorbeelden. Neem ook de tips ter harte die op deze pagina bij de gloeilampjes staan, met name deze tip: het is een goede gewoonte alle variabelen zo veel mogelijk aan het begin van een script te definiëren.  **Undefined** Een variabele die wel is gedeclareerd, maar geen waarde heeft, heeft de waarde undefined.  *Ingeschatte tijdsduur: 30 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  Onderkant formulier |   **Oefenen**   |  | | --- | | In de volgende oefenopgaven past u de zojuist geleerde kennis toe.  Het is erg belangrijk de oefenopgaven eerst zelf te maken, voordat u de uitwerkingen bekijkt.  Maak opgave 6 en 7, de opgaven vindt u [hier](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20oefenopgaven/Oefenopgaven_hoofdstuk_2.pdf#page=2).  Controleer uzelf aan de hand van [deze uitwerkingen](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20oefenopgaven/Uitwerkingen_oefenopgaven_H2.pdf#page=2).  *Ingeschatte tijdsduur: 30 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  Onderkant formulier |   **Bestuderen**   |  | | --- | | Namen van variabelen (identifiers) mogen letters, cijfers, onderstreeptekens en dollartekens bevatten. En verder geldt:   * Namen van variabelen mogen niet met een cijfer beginnen. * Ze kunnen met een $ of \_ beginnen (maar dat is ongebruikelijk). * Namen van variabelen zijn case sensitive (dat wil zeggen dat hoofd- en kleine letters verschillende tekens zijn: dus x and X zijn twee verschillende namen, en ook saldo en Saldo zijn twee verschillende namen). * Gereserveerde woorden (zoals de JavaScript-keywords) kunnen geen namen zijn van variabelen.   Gereserveerde woorden zijn woorden die door JavaScript zelf gebruikt worden en die u niet kunt gebruiken als naam voor een variabele. Zie [JS Reserved Words](http://www.w3schools.com/js/js_reserved.asp) voor het overzicht. Er staan nogal veel woorden op deze pagina die u moet vermijden (of beter kunt vermijden) voor eigen gebruik. Het is natuurlijk niet de bedoeling dat u ze uit uw hoofd leert, maar wel dat u af en toe deze lijst raadpleegt. Voordeel is dat het uitsluitend Engelse termen zijn, indien u voor de namen van uw eigen variabelen Nederlandse namen kiest, is de kans klein dat er verwarring optreedt.  **camelCase** Zoals u in de regels over namen van variabelen ziet, is een spatie in een identifier niet toegestaan. Toch kan het vaak handig zijn een naam te gebruiken die uit meer dan één woord bestaat. Zoals *vorig saldo* of *oppervlakte van cirkel*. Een onder programmeurs veelgebruikte manier is *camelCase*: schrijf de woorden aan elkaar vast, het eerste woord met een kleine letter en begin elk volgend woord met een hoofdletter. Dus:  vorigSaldo oppervlakteVanCirkel  Op deze manier voldoet u aan de regels waar identifiers aan moeten voldoen en is de tekst goed leesbaar. Het woord camelCase is een verwijzing naar een kameel: de hoofdletters in de identifier doen denken aan de bulten op zijn rug.  *Ingeschatte tijdsduur: 30 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid  Onderkant formulier |   **Oefenen**   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | U kunt uw parate kennis over *namen van variabelen* testen op de volgende pagina's:   * [Variables for Numbers](http://www.asmarterwaytolearn.com/js/3.html). * [Variable Names Legal and Illegal](http://www.asmarterwaytolearn.com/js/4.html).   *Ingeschatte tijdsduur: 60 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  **Bestuderen**   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | Bij veel onderwerpen biedt het boek van Flanagan achtergrondinformatie. Houd er rekening mee dat Flanagan niet in de eerste plaats een leerboek is, maar een naslagwerk. Er worden soms termen gebruikt die u nog niet kent. Naarmate u meer van JavaScript weet, zult u het boek beter kunnen begrijpen. Dat neemt niet weg dat het belangrijk is in dit boek te lezen over onderwerpen waar u nu al iets van weet.  Bestudeer over getallen:   * Flanagan: paragraaf 3.1 (Numbers)  tot en met 3.1.3 (Arithmetic in JavaScript).   *Ingeschatte tijdsduur: 45 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  **Bestuderen**   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | Om rekenkundige uitdrukkingen (expressies) met getallen  te kunnen maken, zijn er enkele operatoren.  Lees [hier](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20SA/Operatoren.pdf) meer over deze operatoren.  *Ingeschatte tijdsduur: 30 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  **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 8, de opgave vindt u [hier](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20oefenopgaven/Oefenopgaven_hoofdstuk_2.pdf#page=2).  Controleer uzelf aan de hand van [deze uitwerkingen](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20oefenopgaven/Uitwerkingen_oefenopgaven_H2.pdf#page=2).  *Ingeschatte tijdsduur: 30 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  **Oefenen**   |  | | --- | | Test uw parate kennis over operatoren op de volgende pagina's:   * [Math Expressions: familiar operators](http://www.asmarterwaytolearn.com/js/5.html). * [Math Expressions: unfamiliar operators](http://www.asmarterwaytolearn.com/js/6.html). * [Math Expressions: eliminating ambiguity](http://www.asmarterwaytolearn.com/js/7.html).   *Ingeschatte tijdsduur: 90 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  Onderkant formulier |   **Bestuderen**   |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Strings**  Strings gebruikt u om tekst op te slaan en te bewerken. In JavaScript-code zet u een string tussen dubbele aanhalingstekens of tussen apostrofs. Ook een string kunt u in een variabele opbergen, net als getallen.  var s1 = "roos"; var s2 = "tulp";   * Bestudeer paragraaf 4.7 (Operator Overview) van Flanagan.   *Ingeschatte tijdsduur: 70 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  **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 9, de opgave vindt u [hier](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20oefenopgaven/Oefenopgaven_hoofdstuk_2.pdf#page=2).  Controleer uzelf aan de hand van [deze uitwerkingen](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20oefenopgaven/Uitwerkingen_oefenopgaven_H2.pdf#page=3).  *Ingeschatte tijdsduur: 15 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  Onderkant formulier |   **Bestuderen**   |  | | --- | | **Concatenatie** Met het plusteken kunt u, behalve getallen optellen, ook twee strings achter elkaar plakken, zodat ze een lange string vormen. Dit heet concatenatie.  var s1 = "Goede"; var s2 = "morgen"; var groet = s1 + s2;      // "Goedemorgen"  Ook een string en een getal (number) kunt u op die manier aan elkaar plakken:  var naam = "Thermae "; var jaar = 2000; var s = naam + jaar;      // "Thermae 2000"  Wanneer u meer dan één getal achter een string plakt, worden de getallen achter elkaar gezet, en niet opgeteld, zoals in het volgende voorbeeld:  var inleg = 550; var rente = 11; var s = "saldo = " + inleg + rente; // "saldo = 55011"  Als u wilt dat de getallen eerst worden opgeteld, zet u ze tussen haakjes:  var s = "saldo = " + (inleg+rente); // "saldo = 561"  Concatenatie is het achter elkaar plakken van strings. Ook een string en een number kunt u achter elkaar plakken met het plusteken. Het resultaat is een string.  **De lege string** Een speciale string is de lege string, een string zonder tekens. Zo maakt u de lege string:  var s = "";  Zo'n lege string wordt vaak gebruikt om andere, niet-lege strings te maken met behulp van de += operator. De += operator kunt u niet alleen gebruiken om iets bij een getal op te tellen, zoals we in de paragrafen over numbers hebben gezien, maar ook om twee strings bij elkaar 'op te tellen' (concatenatie). Bijvoorbeeld:  var s = "";    // lege string s += "Maandag"; s += " ";     // spatie s += 5;      // number s += " ";     // spatie s += "december";  Na afloop heeft s de waarde "Maandag 5 december". Dit lijkt misschien op dit moment niet erg nuttig, u zou immers de string meteen de juiste waarde kunnen geven, maar in de praktijk komen de gedeelten van een string vaak van verschillende bronnen en op deze manier kunnen die delen aan elkaar worden geplakt.  Verder wil de lege string nogal eens voorkomen in een tekstvak waar de gebruiker niets intikt, maar wel op Enter drukt. We zullen daar later voorbeelden van zien.  **De methode trim()** Een methode is een stukje code met een naam. Veel methoden in JavaScript zijn ingebouwd, en u kunt ze ook zelf maken. In de volgende hoofdstukken zullen we daar voorbeelden van zien.  Voor het string-type zijn in JavaScript een groot aantal methoden gedefinieerd, u kunt de hele lijst zien op [w3schools.com](http://www.w3schools.com/jsref/jsref_obj_string.asp) onder het kopje String Methods. Al deze methoden kunnen een of andere bewerking op een string doen.  Voorbeeld van zo'n methode is trim(). U kunt een methode herkennen, omdat achter hun naam vaak een openings- en sluithaakje wordt geschreven.  In [dit voorbeeld](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20SA/Voorbeeld%20trim__.pdf)is s een string met twee spaties aan de voorkant en drie aan de achterkant. De uitdrukking s.trim() levert een nieuwe string, waarbij de spaties (whitespace) aan de voor- en achterkant zijn verwijderd. De string s zelf is ongewijzigd.  *Ingeschatte tijdsduur: 20 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  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 10, de opgave vindt u [hier](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20oefenopgaven/Oefenopgaven_hoofdstuk_2.pdf#page=2).  Controleer uzelf aan de hand van [deze uitwerkingen](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20oefenopgaven/Uitwerkingen_oefenopgaven_H2.pdf#page=3).  *Ingeschatte tijdsduur: 15 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  **Oefenen**   |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | Uw parate kennis over strings kunt u testen via:   * [Variables for Strings](http://www.asmarterwaytolearn.com/js/2.html) * [Concatenating text strings](http://www.asmarterwaytolearn.com/js/8.html).   *Ingeschatte tijdsduur: 60 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  **Bestuderen**   |  | | --- | | Bij veel onderwerpen biedt het boek van Flanagan achtergrondinformatie. Houd er rekening mee dat Flanagan een naslagwerk is. Er worden soms termen gebruikt die u nog niet kent.  Bestudeer over de + operator:   * paragraaf 4.8.1 (Flanagan).   *Ingeschatte tijdsduur: 15 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  Onderkant formulier |   **Bestuderen**   |  |  |  | | --- | --- | --- | | **Conversie van number naar string en omgekeerd** Door de concatenatie van een string en een number krijgt u een waarde van het type string. Het getal wordt daarmee omgezet (geconverteerd) naar een string.  Lees [hier](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20SA/Conversie%20van%20number%20naar%20string%20en%20omgekeerd.pdf) meer over de conversie van number naar string en omgekeerd.  *Ingeschatte tijdsduur: 150 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  **Bestuderen**   |  | | --- | | Bij veel onderwerpen biedt het boek van Flanagan achtergrondinformatie. Houd er rekening mee dat Flanagan niet in de eerste plaats een leerboek is, maar een naslagwerk. Dat neemt niet weg dat het belangrijk is in dit boek te lezen over onderwerpen waar u nu al iets van weet.  Bestudeer uit Flanagan:   * paragraaf 3.3 (Boolean Values) * paragraaf 4.10 (Logical Expressions) * paragraaf 5.4 (Conditionals) tot en met paragraaf 5.4.2  (else if).   *Ingeschatte tijdsduur: 70 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  Onderkant formulier |   **Oefenen**   |  | | --- | | In de volgende opgaven past u de zojuist geleerde kennis toe. Het is erg belangrijk de oefenopgaven eerst zelf te maken, voordat u de uitwerking bekijkt.  Maak opgave 11 t/m 13, de opgaven vindt u [hier](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20oefenopgaven/Oefenopgaven_hoofdstuk_2.pdf#page=3).  Controleer uzelf aan de hand van [deze uitwerkingen](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20oefenopgaven/Uitwerkingen_oefenopgaven_H2.pdf#page=3).  *Ingeschatte tijdsduur: 60 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  Onderkant formulier |   Uw parate kennis over conditionele statements kunt u testen via:   * [if statements](http://www.asmarterwaytolearn.com/js/10.html) * [Comparison operators](http://www.asmarterwaytolearn.com/js/11.html) * [if...else and else if statements](http://www.asmarterwaytolearn.com/js/12.html).   Ingeschatte tijdsduur: 90 minuten.  Bovenkant formulier Status U hebt deze activiteit voltooid.  Onderkant formulierOnderkant formulier |   **Bestuderen**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | Bestudeer over relationele expressies:   * Flanagan: paragraaf 4.9.1 (Equality an Inequality Operators) * Flanagan: paragraaf 4.9.2 (Comparison Operators).   *Ingeschatte tijdsduur: 45 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  **Bestuderen**   |  |  |  |  | | --- | --- | --- | --- | | **Pop-ups** Er zijn in JavaScript verschillende manieren om iets aan de gebruiker duidelijk te maken. Een van die manieren hebben we in veel van de voorbeelden gezien: door middel van het wijzigen van de tekst in de HTML-code.  Een andere manier om te communiceren is via een pop-upbox. Dat ziet er meestal uit als een klein rechthoekig dialoogvenster met een mededeling die de gebruiker weg moet klikken, alvorens verder te kunnen gaan. Op die manier weet u vrijwel zeker dat de gebruiker de mededeling gezien heeft.  JavaScript kent drie soorten pop-ups:   * alert() * confirm() * prompt()   De pop-ups, en hun onderlinge verschillen, kunt u vinden op [w3schools.com - popup](http://www.w3schools.com/js/js_popup.asp).  **Controle op de invoer (validation)** Tijdens de invoer bijvoorbeeld een getal in een tekstvak, of als antwoord op een prompt kan wel iets misgaan. De gebruiker kan iets invoeren wat geen getal is. Het zou prettig zijn als JavaScript daarop controleert. Dit heet ook wel validatie (*validation*).  Stel dat de pagina een tekstvak bevat met de naam invoervak, waar de gebruiker geacht wordt een getal in te voeren. Waar iets mis kan gaan, is bij de omzetting van de string uit het tekstvak naar een number:  var getal = +invoervak.value;  Als de gebruiker geen getal invoert, maar bijvoorbeeld *abc*, dan kan de conversie niet plaatsvinden en krijgt getal de waarde NaN.  We kunnen controleren of dit het geval is met de functie isNaN(), deze leverttrue als getal gelijk is aan NaN, en anders false. Let op: u kunt niet metgetal==NaN controleren of getal een geldig getal bevat.  var mededeling; if( isNaN( getal ) )   mededeling = "Geen geldig getal ingevoerd"; else   mededeling = "Ingevoerd: " + getal;  document.getElementById("resultaat").innerHTML = mededeling;  *Ingeschatte tijdsduur: 30 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  **Oefenen**   |  | | --- | | In deze oefenopgaven past u de zojuist geleerde kennis toe. Het is erg belangrijk de oefenopgaven eerst zelf te maken, voordat u de uitwerking bekijkt.  Maak opgave 14 en 15, de opgaven vindt u [hier](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20oefenopgaven/Oefenopgaven_hoofdstuk_2.pdf#page=3).  Controleer uzelf aan de hand van [deze uitwerkingen.](https://www5.loi.nl/access/content/group/255889f1-27c3-4e90-944f-11761ec46ec4/Bestanden%20oefenopgaven/Uitwerkingen_oefenopgaven_H2.pdf#page=5)  *Ingeschatte tijdsduur: 30 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  Onderkant formulier |   **Oefenen**   |  | | --- | | Test uw parate kennis over pop-ups via:   * [Alerts](http://www.asmarterwaytolearn.com/js/1.html) * [Prompts](http://www.asmarterwaytolearn.com/js/9.html).   *Ingeschatte tijdsduur: 60 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  Onderkant formulier |   **Inzenden**   |  | | --- | | JavaScript, variabelen en gegevens besluit met een inzendopgave die uit drie delen bestaat. Test uw oplossing goed voor u deze inzendt. Stuur niet alleen het script, maar het volledige HTML-bestand met het script in.  U ontvangt het beoordeelde werk binnen enkele dagen retour. Het resultaat kunt u vinden in de huiswerkresultaten.  *Ingeschatte tijdsduur: 210 minuten.*  Bovenkant formulier  **Status**  U hebt deze activiteit voltooid.  Onderkant formulier |   Te doen   * [Inzendopgave 242S1](https://www5.loi.nl/portal/tool/b4bd16be-c04d-4da9-829a-6131f20245ac/?userId=f15516b9-d75c-4412-9fef-6401482740e1&learningActionId=61298&modulePLSiteId=255889f1-27c3-4e90-944f-11761ec46ec4&moduleTitle=Programmeren%20in%20JavaScript&externalActionEntitylId=81926&panel=Main)   Onderkant formulier |   Onderkant formulier |   **Onderdeel 1** Maak een pagina met daarop een tekstvak waarin de gebruiker een getal *x*, en een tekstvak  waarin de gebruiker een getal y kan invoeren. Voeg een knop toe die tot functie heeft om de  getallen *x* en *y* uit de tekstvakken bij elkaar op te tellen, en het resultaat op de pagina weer  te geven. U mag ervan uitgaan dat er correcte getallen worden ingevoerd.  **Onderdeel 2** Breid de code van het vorige onderdeel zodanig uit dat niet alleen de som, maar ook het verschil,  het product en het quotiënt van beide getallen op de pagina wordt gezet.  **Onderdeel 3** Breid de code van het vorige onderdeel zodanig uit dat op de pagina ook een mededeling  komt of *x* groter dan, kleiner dan, of gelijk is aan *y*.Onderkant formulier |   Onderkant formulier |   Onderkant formulier |   Onderkant formulier |   Onderkant formulier |   Onderkant formulier |   Onderkant formulier |   Onderkant formulier |   Onderkant formulier |   Onderkant formulier |   Onderkant formulier |   Onderkant formulier |