



Opleiding Applicatie Ontwikkelaar

Leerlijn Interface

Reguliere Expressies

Domein C Level 2

Auteur: Aminah Balfaqih

Datum: 9-10-2019

Versie: 2.0

Inhoudsopgave

Overzicht	3
Voorkennis.....	3
Materialen	3
Instructie	3
Doelen	3
Beoordeling	3
Studieblok 1.....	4
Bestuderen	4
Opdrachten.....	6
Studieblok 2.....	7
Bestudeer	7
Opdrachten.....	7
Studieblok 3.....	9
Bestudeer	9
Opdrachten.....	9
Studieblok 4.....	10
Bestudeer	10
Opdrachten.....	10
Inleveropdrachten	11
Inleveren.....	12
Bronnen	13

Overzicht

Level: Domein C Level 2
Duur: 3 weken
Methode: Weekplanning

Voorkennis

HTML en CSS,

C2 Javascript

Materialen

Er wordt gebruik gemaakt van de volgende studiematerialen:

- www.w3schools.com
- Jouw laptop

Instructie

Het is het aan te bevelen om dit document van begin tot eind te volgen. Probeer zoveel mogelijk opdrachten uit. Laat tussendoor zoveel mogelijk aan de docent zien om te kijken of je op de goede weg zit.

Doelen

Je kan gebruik maken van reguliere expressies om data te valideren.

Je kan gebruik maken van JQuery.

Beoordeling

Deze module wordt afgesloten met twee eindopdrachten. Deze moet je inleveren in één zip bestand in Magister

Studieblok 1

Bestuderen

Van Wikipedia:

Reguliere expressie

Een **reguliere expressie** (uit het Engels, *regular expression*, afgekort tot “regexp”, “regex” of RE) is een manier om patronen te beschrijven waarmee een [computer](#) tekst kan herkennen. Er bestaat hiervoor een formele [syntaxis](#), die deels is gestandaardiseerd.

Reguliere expressies worden bijvoorbeeld in [teksteditors](#) gebruikt om stukken tekst te doorzoeken, te manipuleren, in andere programma's worden ze gebruikt om te controleren, dat bepaalde patronen voorkomen. Veel [programmeertalen](#) ondersteunen reguliere expressies voor tekstmanipulatie. Sommige, zoals [Perl](#) en [JavaScript](#), hebben ze zelfs in hun syntaxis ingebouwd. Reguliere expressies zijn vooral populair geworden door de hulpprogramma's van het besturingssysteem [Unix](#), zoals [sed](#) en [grep](#).

Een eenvoudige variant van de reguliere expressie is in veel [besturingssystemen](#) te vinden als de [jokertekens](#) die gebruikt kunnen worden bij het zoeken naar [bestandsnamen](#).

Grondbegrippen

Een reguliere expressie omschrijft een verzameling tekenreeksen ([strings](#)) zonder ze allemaal op te noemen. De drie strings *Handel*, *Händel* en *Haendel* kunnen bijvoorbeeld beschreven worden met het patroon “H(a|ä|ae)ndel”.

Gewone letters en cijfers in de reguliere expressie herkennen hetzelfde teken in de te vinden tekenreeks. Een aantal tekens hebben speciale betekenis:

- Een punt (.) staat voor een willekeurig teken, behalve het teken voor een newline (`\n`).
- Vierkante haken geven een lijst van mogelijke tekens: `[abc]`.
 - Binnen vierkante haken staat een minteken voor een reeks: `[a-zA-Z]` is het patroon waarmee alle letters “gevangen” worden.
 - Een dakje als eerste teken binnen de vierkante haken verandert de tekenverzameling in het omgekeerde: `[^0-9]` herkent alles wat geen cijfer is.
- Een dakje `^` staat voor het begin van de regel.
- Een dollarteken `$` staat voor het eind van de regel.

Deze basiselementen kunnen worden gecombineerd met de volgende constructies:

Keuze

Een verticale balk scheidt de alternatieven, bijvoorbeeld “groen|rood” herkent “groen” of “rood”.



Kwantificatie

Een kwantor achter een teken geeft aan hoe vaak dat teken voor mag komen. De meest voorkomende kwantoren zijn +, ? en *:

- + Een plusteken geeft aan dat het voorafgaande teken ten minste één keer moet voorkomen, bijvoorbeeld “goo+gle” herkent *google*, *gooogle*, *gooooogle*, enz.
- ? Een vraagteken geeft aan dat het voorafgaande teken ten hoogste één keer mag voorkomen, bijvoorbeeld “De Bruij?n” herkent “De Bruin” en “De Bruijn”.
- * Een sterretje geeft aan dat het voorafgaande teken nul of meer keer mag voorkomen, bijvoorbeeld “0*42” herkent *42*, *042*, *0042*, enzovoort.

Een veelvoorkomende constructie is .* dat alle tekst vindt.

Groepering

Haakjes maken een eenheid van het patroon waar ze omheen staan, bijvoorbeeld “(va|moe)der” is hetzelfde als “vader|moeder” en “(groot)?vader” herkent zowel “vader” als “grootvader”.

Deze constructies kunnen worden gecombineerd in hetzelfde patroon, zodat “H(ae?|ä)ndel” hetzelfde is als “H(a|ae|ä)ndel”.

De precieze syntaxis varieert enigszins tussen de verschillende programma’s, maar meestal worden de bovenstaande gebruikt.

Bestudeer op w3schools.com:

- https://www.w3schools.com/jsref/jsref_obj_regexp.asp
- https://www.w3schools.com/js/js_regexp.asp

Opdrachten

Opdracht 1

Maak een reguliere expressie voor de volgende onderdelen:

- A. Nederlandse postcode
- B. Nederlands telefoonnummer
- C. Email
- D. Adres
- E. Man of vrouw invoer

Op de volgende websites kan je jouw reguliere expressie uitproberen:

- <http://www.regexpal.com/>
- <https://regex101.com/>

Opdracht 2

Test al jouw reguliere expressie van opdracht 1 in javascript.

Deze website kan je helpen: https://www.w3schools.com/js/js_regex.asp

Opdracht 3

Bestudeer de volgende gegevens en maak hiervoor een reguliere expressie:

0023
0240
2302
4002
0240

Studieblok 2

Bestudeer

Bestudeer op w3schools.com (jQuery tutorial):

- Home
- Intro
- Get Started
- Syntax
- Selectors
- jQuery Effects Hide/Show

Opdrachten

Opdracht 1

Maak een simpele html pagina met een aantal `` tags erin. Bijvoorbeeld een wensenlijstje voor je verjaardag. Zorg ervoor dat je jQuery kunt gebruiken in je html pagina. Zet er ook een button op

- Zorg dat met één klik op de knop alle `` onzichtbaar worden
- Voeg nog een knop toe en zorg ervoor dat de ene knop alle `` verbergt en de andere alle `` weer laat zien.
- Geef een paar `` tags de klasse naam "test". Voeg een stukje jQuery toe die na een klik op een derde knop alle `` met klassenaam test laat verdwijnen.
- * Maak met 1 knop alle `` blauw en met de andere knop alle `` weer normaal.

* Deze opdracht is lastiger, kijk hoe je css dingen doet in jQuery.

Opdracht 2

Maak een simpele html pagina met een aantal `` tags erin en daarom heen een aantal `<div>` tags.

- Maak een knop die alle `` elementen blauw maakt maar alleen als de css klasse van die div kleurAanpasbaar is.
- Maak nog een knop die zorgt dat de elementen weer terug gaan naar hun oorspronkelijke staat.

Opdracht 3

Gebruik het vorige html bestand.

- Maak een knop die alle `` tags blauw maakt die binnen een div met klasse kleurAanpasbaar heeft en alleen als het `` element zelf een klasse "aanpasbaar" heeft.
- Maak nog een knop die zorgt dat de elementen weer terug gaan naar hun oorspronkelijke staat.



Opdracht 4

Maak een html bestand met vier rijen van <div> waarin weer 4 <div> elementen komen. Zorg dat je een net vierkant krijgt met vier rijen van vier blokjes van bijvoorbeeld 100px. Dit kun je in CSS doen.

- a. Maak een knop die aan elk blokje een willekeurige kleur geeft, waarbij je kunt kiezen uit 4 kleuren.
- b. Als je op een van de blokjes klikt moeten alle blokjes die dezelfde kleur hebben weer wit worden. Hint: gebruik kleuren die een gewone naam hebben...

Studieblok 3

Bestudeer

Bestudeer van w3schools.com:

- Events

Opdrachten

Opdracht 1

Maak een html bestand met drie knoppen en een `<div>`.

- Zorg dat je als je op een button klikt dat de naam van de button in de div komt te staan
- Doe hetzelfde maar dan als je over de knop gaat met de muis
- Pas je bestand aan zodat als je over de eerste knop gaat met de muis, je programma willekeurig een van de twee andere knoppen klikt.

Opdracht 2

Maak een html bestand met één knop en een `<div>`, de opmaak van de div mag je in css doen of in jquery.

Maak in jQuery een programma dat er voor zorgt dat de div afwisselend rood en groen van kleur wordt als je er op klikt.

Opdracht 3

Maak een simpele form in html met minimaal 3 invoervelden en zorg er voor dat als de cursor in een van de velden staat de rand een opvallende kleur krijgt en anders de standaard kleur.

Opdracht 4

Maak een html bestand met vier rijen van `<div>` waarin weer 4 `<div>` elementen komen. Zorg dat je een net vierkant krijgt met vier rijen van vier blokjes van bijvoorbeeld 100px. Dit kun je in CSS doen. (Of je maakt natuurlijk een kopie van het bestand van vorige week)

Kies een kleur die een naam heeft (lekker makkelijk), zorg ervoor dat als je met de muis over een vakje gaat dat dat vakje de gekozen kleur krijgt en als de muis er niet boven hangt het vakje weer normaal wordt.

Opdracht 5

Begin met hetzelfde bestand als in opdracht 4

- Bedenk een procedure die er voor zorgt dat alle vakjes een kleur krijgen, steeds twee vakjes dezelfde kleur.

Zorg ervoor dat er een tellertje op het scherm komt, en als je achter elkaar twee keer dezelfde kleur klikt tel je 1 op bij het tellertje.

Studieblok 4

Bestudeer

Bestudeer van w3schools.com:

- jQuery Effects

Opdrachten

Opdracht 1

Maak een simpele pagina en laat onderin een regeltekst lopen van rechts naar links. Net als je bij sommige TV programma's ziet. Basisopmaak mag in CSS, het lopende gedeelte moet in jQuery.

Opdracht 2

Maak weer een simpele html pagina. Zet daar op een div, die aan de rechterkant ongeveer een kwart van je scherm in de breedte in beslag neemt. Zorg dat er maximaal 5 regels tekst inpassen, en vul die 5 regels met willekeurige tekst.

- Maak een knop die aan de onderkant een stukje tekst toevoegt, en dus aan de bovenkant eentje weggooit.
- Zorg dat opdracht a mooier wordt, door als je eentje toevoegt, het bovenste item steeds meer onzichtbaar wordt en dan verwijderd wordt, en daarna het nieuwe item onderaan toegevoegd en langzaam zichtbaar wordt.

Opdracht 3

Maak een rechthoek van divs en vul die met plaatjes door in css dit te regelen.

- Als je met je muis over een plaatje gaat moet hij groter worden
- Als je op het plaatje klikt, vliegt het plaatje op een leuke manier uit beeld. Hoe meer verschillende manier hoe leuker!

Opdracht 4

Maak een webpagina waarop een slider van drie plaatjes komt. Die slider moet je vanzelfsprekend zelf maken. Hij moet doorlopend zijn. Je mag een tutorial gebruiken.

Inleveropdrachten

Opdracht 1

Maak onderstaand voorbeeld (gebruik divs voor de grote velden):

Naam invoeren:	<input type="text"/>	Toevoegen
<div></div>	<div>>>>>>></div> <div><<<<<<</div>	<div></div>

Als je op toevoegen klikt dan komt de ingevulde naam in de linkerkolom te staan. Met de knoppen tussen de kolommen in kan je de naam van links naar rechts schuiven en vice versa. Tip! Maak gebruik van een array.

Zie volgende blad voor opdracht 2!

Opdracht 2

- Maak een formulier die zowel in het Engels als in het Nederlands weergegeven kan worden. De gebruiker moet kunnen kiezen tussen Engels of Nederlands maak hiervoor 2 links of 2 knoppen. De verschillende namen voor de velden moeten opgeslagen worden in een array. Na de keuze gebruik je jquery fade in/out om het formulier weer te geven.
- Er staat een keuze op voor heer of mevrouw. Als een van de opties gekozen wordt dan zal de style veranderen als volgt:
 - Bij een man krijg je een donkergroen achtergrond met lichtgroene tekstvelden.
 - Bij een vrouw krijg je een paarse achtergrond met roze tekstvelden.
- Als er een geboortedatum ingevuld wordt dan wordt de leeftijd automatisch berekent. De leeftijd moet langzaam in beeld komen achter het tekstveld van de geboortedatum. Gebruik hiervoor jquery effects. Zoek op het internet naar een formule voor leeftijdsberekening.
- Alle velden bevatten een test met reguliere expressies. Geef een popup venster bij een eventuele foutmelding.

De volgende velden moeten in het formulier komen:

- Aanhef (heer of mevrouw)
- Naam (regex=alleen letters)
- Adres (regex=alle mogelijke tekens waarin een cijfer moet voorkomen)
- Woonplaats (regex=alleen letters met mogelijk een cijfer)
- Postcode (regex=4 cijfers met opvolgend 2 letters)
- Geboortedatum (regex=notatie: 27-07-1978)
- Nationaliteit (regex=alleen letters)
- Beroep (regex=alleen letters)
- Opmerkingen (textarea)

Alle velden behalve “opmerkingen” zijn verplicht.

Inleveren

Alle bestanden van deze 2 opdrachten moet je inleveren in één zip bestand in Magister

Bronnen

- ✓ <http://javascript.divendo-webs.com/>
- ✓ https://nl.wikipedia.org/wiki/Reguliere_expressie
- ✓ <http://www.w3schools.com>