VBA voor Doe het Zelvers deel 21

Handleiding van Helpmij.nl

Auteur: leofact

Oktober 2015

“ Dé grootste en gratis computerhelpdesk van Nederland ”

Vorige aflevering

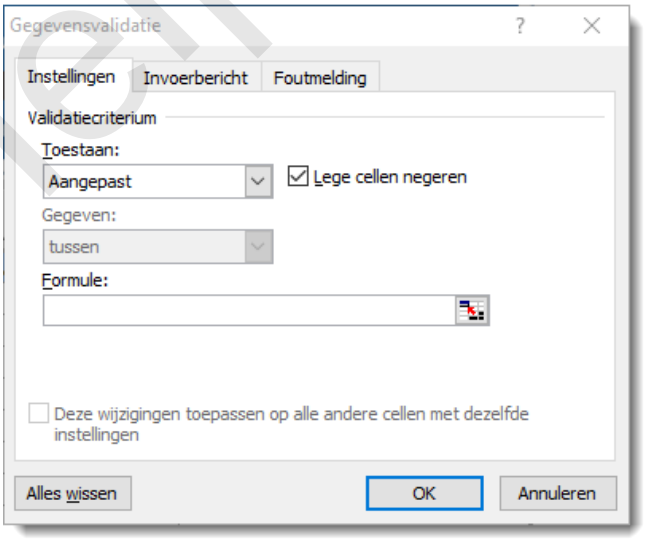
In deel 20 werd uitgelegd wat een operator is. Er werden vijf soorten operators benoemd; namelijk de rekenkundige, de vergelijkende, de aaneenschakelende, de logische en als laatste de object-operatoren. Van iedere soort werden de belangrijkste operatoren gegeven met de wijze waarop deze gebruikt kunnen worden. Daarnaast werden een aantal voorbeeldjes gegeven hoe deze in een procedure verwerkt kunnen worden.

Deze aflevering

In dit deel worden de regular expressions behandeld. Deze expressies maken het mogelijk om op eenvoudige wijze invoermaskers te maken waarmee invoer op het juiste format is te controleren. Microsoft heeft de expressies beschikbaar gesteld in de VBScript Regular Expressions-bibliotheek. Deze wordt in deze aflevering gebruikt om de juiste invoer te controleren van de postcode, een mobiel telefoonnummer, een e-mailadres, een BSN en een Nederlandse IBAN. Bij de laatste twee wordt naast de controle op het format ook een rekenkundige controle gegeven om de juistheid van het nummer te kunnen verifiëren. Samen met de invoer met behulp van een een inputbox wordt ook behandeld hoe je de functie in een werkblad kunt opnemen. Dit alles wordt uitgewerkt in de bijlage, die hier is te downloaden.

Gegevensvalidatie

Het komt regelmatig voor dat je met bepaalde gegevensinvoer verder wilt werken in een bepaald project. Het kan daarbij van belang zijn dat de ingevoerde gegevens aan bepaalde voorwaarden voldoen zodat je deze kunt gebruiken in een formule of in het verdere verloop van de procedure. Wanneer er gerekend wordt, is het bijvoorbeeld noodzakelijk dat de invoer als getal wordt herkend. Een ander voorbeeld is het gebruik van datums; de invoer dient dan voor Excel als datum herkenbaar te zijn. Excel kent hiervoor al een handige functie; namelijk gegevensvalidatie. Deze functie start je vanuit het lint > tab **Gegevens** > **Gegevensvalidatie**:

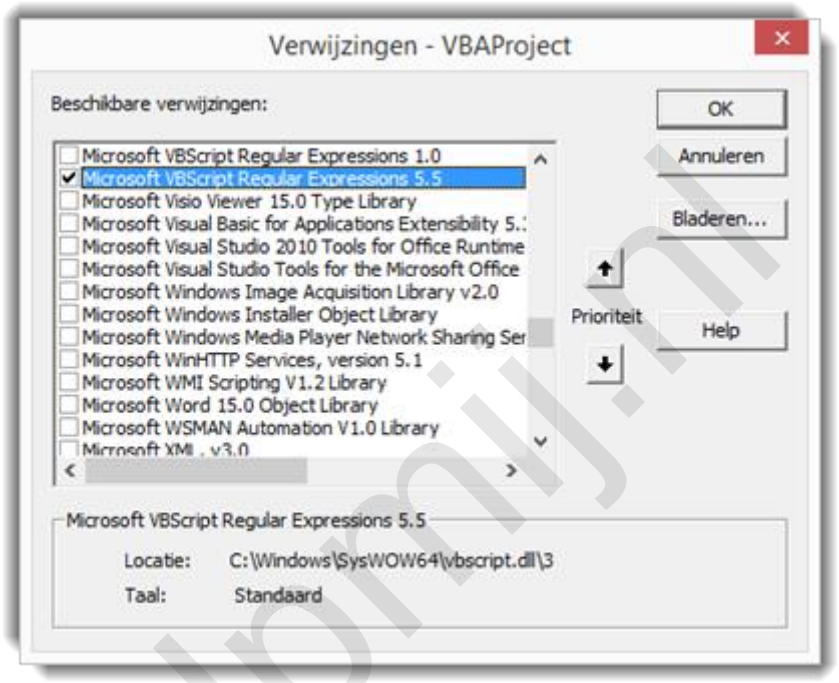


In dit venster kun je instellen welke controle je wilt laten uitvoeren. Daarnaast kun je de gebruiker bijstaan met de juiste invoer door middel van een invoerbericht. Bij foutieve invoer kun je dan een foutmeldingsbericht laten geven. Meer informatie over gegevensvalidatie vind je hier.

Met deze flexibele en krachtige functie kun je op vele manieren invoer controleren op juistheid. Toch zijn er situaties denkbaar dat gegevensvalidatie niet bruikbaar is, of niet handig ingezet kan worden. Dat is onder meer het geval wanneer er geen celinvoer wordt gebruikt, zoals bijvoorbeeld bij het gebruik van inputboxen of userforms. Met regular expressions (afgekort tot regex of regexp) kun je op relatief eenvoudige wijze de invoer hiervan op het juiste format controleren.

Regular expressions

Een reguliere expressie (in het Nederlands) maakt het mogelijk om te omschrijven aan wat voor patroon een bepaalde (invoer)string moet voldoen. De regular expressions (afgekort regex of regexp) werden al in de vijftiger jaren ontwikkeld en worden nog altijd door veel programmeertalen ondersteund, waaronder VBA. De bibliotheek met de regular expressions maken we beschikbaar op de inmiddels bekend vooronderstelde wijze; door deze via het menu **Extra** > **Verwijzingen** te selecteren in de beschikbare bibliotheken:



Door het gebruik van early binding kunnen we bij het programmeren van de regex op de programmeerhulp Intellisense rekenen (zie daarvoor eventueel deel 13).

Expressie-opbouw

De regex kent een eigen syntax waarmee de expressies kunnen worden opgebouwd. Deze kent de volgende eigenschappen:

- *Pattern*; string-expressie welke het controle patroon bevat.
- *Global*; booleanwaarde waarmee aangegeven wordt of er na de eerste overeenkomst gestopt wordt met vergelijken (False), of dat er doorgezocht wordt naar alle overeenkomsten (True).
- *IngoreCase*; eveneens een booleanwaarde waarmee geschakeld wordt tussen hoofdletter-ongevoelig zoeken (True) of hoofdletter-gevoelig zoeken (False).

En deze methodes:

- *Execute*; voert de vergelijking uit op de te testen tekst in de doel-string. Het resultaat is een Match-object voor iedere gevonden overeenkomst.
- *Replace*; hiermee wordt ieder gevonden overeenkomst vervangen door de gespecificeerde vervang-tekst.
- *Test*; booleanwaarde waarmee getest wordt of een vergelijking mogelijk is (True), of niet (False).

Metacharacters

Met de metacharacters of speciale tekens wordt het patroon opgebouwd waarmee de vergelijking wordt uitgevoerd. Een aantal belangrijke voorbeelden:

- []: vierkante haakjes; geeft exacte overeenkomst met één of meerdere karakters
  - [abc] geeft overeenkomst als a, b of c voorkomt in de testreeks
- (): haakjes; worden gebruikt om patroondefinities te groeperen (net als bij rekenen)
- ^: dakje; geeft het begin van de input of teststring
  - ^A; het eerste karakter moet de hoofdletter A zijn
- $: dollarteken; komt overeen met het laatste teken van de input of teststring
  - $0; het laatste karakter dient een nul te zijn
- -: minteken; wordt gebruikt om een reeks aan te duiden
  - [a-z]; geeft overeenkomst met alle kleine letters van a tot z
- {}: accolades; hiermee wordt het aantal herhalingen van het voorafgaande aangeduid. Er kan een minimum en een maximum worden opgegeven.
  - [A-Z]{2}; geeft overeenkomst bij een reeks van twee hoofdletters
  - [0-9]{2,5}; geeft overeenkomst bij een getal van 2 tot 5 cijfers
- |: pipe; staat voor of (Or) en is een scheidingsteken tussen twee patronen
  - auto|fiets geeft overeenkomst bij auto of bij fiets
- [^]: Dakje binnen haakjes; (Not) geeft de tegengestelde overeenkomst
  - [^0-9]; het karakter mag geen getal zijn
- .: punt: dit is de wildcard, deze staat voor ieder teken
- +: plus; herhaling. dit geeft aan dat het voorafgaande één of meerdere keren mag worden herhaald
  - ba+s geeft overeenkomst bij bas, baas, baaas etcetera en niet bij bs
- ?: vraagteken; geeft aan dat het voorafgaande afwezig mag zijn, of maximaal één keer mag voorkomen
  - br?oek; zowel boek als broek geven overeenkomst, brroek wordt echter niet herkend
- *: sterretje; geeft aan dat het voorafgaande afwezig mag zijn, maar ook één of meerdere keren mag voorkomen
  - bro*s; geeft overeenkomst bij: brs, bros, broos enzovoorts
- \: backslash; geeft de letterlijke betekenis van de speciale tekens
  - \& komt overeen met het dollarteken
  - \- komt overeen met het minteken

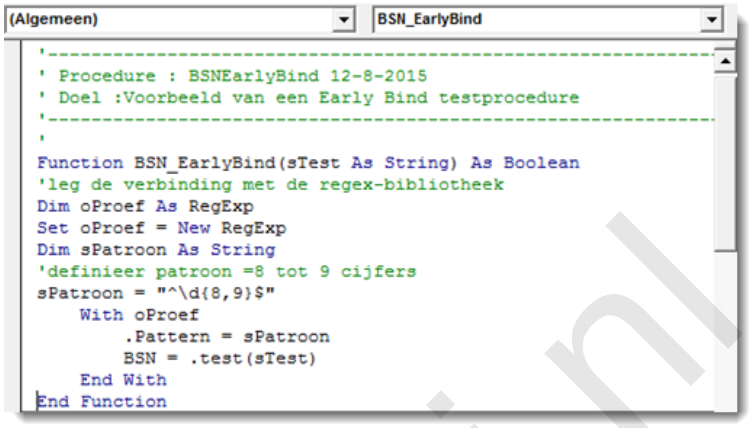
De backslash kan ook voorafgaan aan een gewone letter en geeft dan een voorgedefinieerde overeenkomst. Een aantal voorbeelden hiervan volgen hieronder:

- \d; komt overeen met een cijfer = [0-9]
- \D; komt overeen met een alfa-teken (geen cijfer) = [a-zA-Z]
- \w; komt overeen met een alfanummeriek teken inclusief underscore =[a-zA-Z0-9_]
- \W; komt overeen met de niet-alfanummerieke tekens = [^a-zA-Z0-9_]
- \n; new line of nieuwe regel
- \s; voor overeenkomst met ieder spatiëring-teken, inclusief new line en return
- \S; voor overeenkomst met elk teken, behalve spatiëring-tekens

Met deze set aan speciale tekens in gedachten kunnen we nu aan de gang om de reguliere expressie te formuleren. Als eerste is er een routine nodig waarmee de vergelijking wordt uitgevoerd. We maken daarvoor een functie. Het handige van een functie is dat deze gelijk ook in de werkbladen als functie beschikbaar is. Het is mogelijk om de functie zodanig op te bouwen dat de test-string en het herkenningspatroon worden meegegeven als variabele. De functie is dan universeel voor iedere test in te zetten. In de praktijk kan het echter ook handig zijn om voor iedere aparte controle een aparte functie te maken. Alleen de teststring hoeft dan doorgegeven te worden naar de functie. Op deze manier zijn de volgende voorbeelden geprogrammeerd.

BSN

Als eerste maken we de functie in early binding zodat Intellisense bruikbaar is. In dit voorbeeld wordt er een format gegeven voor het Burger Service Nummer. Door de early binding kan er makkelijk worden geëxperimenteerd met alle methoden en eigenschappen. Zie daarvoor het volgende voorbeeld:

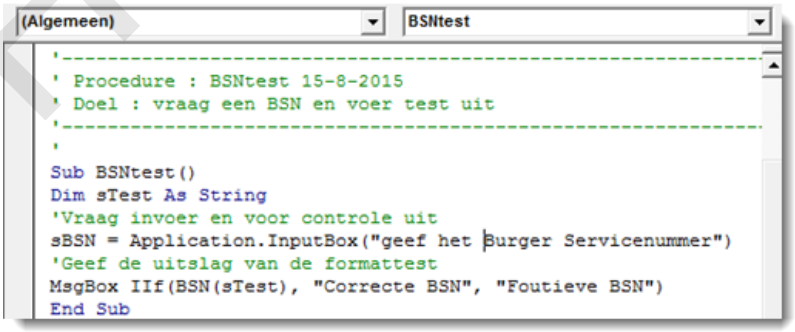


De controle-string checkt of er uitsluitend getallen zijn ingevoerd en of dit getal uit acht of negen cijfers bestaat.

Voor eventuele verspreiding van het werkboek is het natuurlijk verstandig om de functie te herschrijven in late binding. Daarvoor wordt CreateObject gebruikt. Voorbeelden hiervan volgen verderop.

In het werkboek kan de functie als volgt worden ingezet in een cel. Bijvoorbeeld in B17 (invoer in B16):
=ALS(BSN(B16);"correcte BSN";ALS(LENGTE(B16)>0;"foutieve BSN";""))
Hierbij wordt in het eerste deel gekeken of de uitslag van de functie BSN waar is. Zo niet, dan wordt gegeken of er wel invoer is en als dat het geval blijkt wordt er aangegeven dat de BSN niet geldig is.

In VBA kan een inputbox worden ingezet om invoer van het nummer op te vragen bij de gebruiker:



Postcode

In het volgende voorbeeld wordt de invoer van een postcode gecontroleerd. Een spatie tussen de cijfers en letters mag, maar is niet verplicht. Er worden alleen hoofdletters toegestaan, maar het is eenvoudig om ook kleine letters mogelijk te maken:

```
(Algemeen)                                    Postcode
'---------------------------------------------------------------
' Procedure : Postcode 12-8-2015
' Doel : Test Postcoede in Late Binding
'---------------------------------------------------------------
'
Function Postcode(sTest As String) As Boolean
Dim sPatroon As String
'definieer patroon =
'   4 cijfers (eerste geen nul), spatie 2 hoofdletters
'   spatie is niet verplicht
sPatroon = "^[1-9]\d{3} ?[A-Z]{2}$"
'"sPatroon = ^[1-9]\d{3} [A-Z]{2}$" '=> verplichte spatie
    With CreateObject("vbscript.regexp")
        'hoofdletters negeren:
        '.IgnoreCase = True
        .Pattern = sPatroon
        Postcode = .test(sTest)
    End With
End Function
```

De postcode kan bijvoorbeeld als volgt worden opgevraagd:

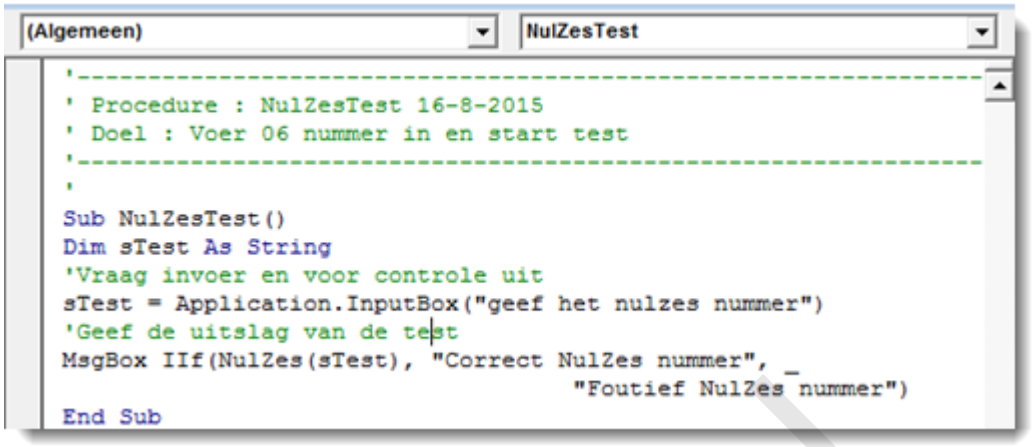
```
(Algemeen)                                    PostcodeTest
'---------------------------------------------------------------
' Procedure : PostcodeTest 16-8-2015
' Doel : vraagt om invoer voor de postcode check
'---------------------------------------------------------------
'
Sub PostcodeTest()
Dim sTest As String
'Vraag invoer en voor controle uit
sTest = Application.InputBox("geef de postcode")
'Geef de uitslag van de test
MsgBox IIf(Postcode(sTest), "Correcte postcode", _
                            "Foutieve postcode")
End Sub
```

Mobiel nummer

Een mobiel of nulzes-nummer kan worden voorafgegaan door het Nederlandse landnummer op verschillende manieren. De volgende controle maakt alle verschillende formats mogelijk:

```
(Algemeen)                                    NulZes
End Sub
'---------------------------------------------------------------
' Procedure : NulZes 12-8-2015
' Doel : Test NulZes nummer op samenstelling
'---------------------------------------------------------------
'
Function NulZes(sTest As String) As Boolean
Dim sPatroon As String
'definieer patroon = landnr +31 of 0031 of 0 dan 6,
' één cijfer geen nul en 7 cijfers incl. 0
sPatroon = "^(\+31|0|0031)6[1-9]\d{7}$"
    With CreateObject("vbscript.regexp")
        .Pattern = sPatroon
        NulZes = .test(sTest)
    End With
End Function
```

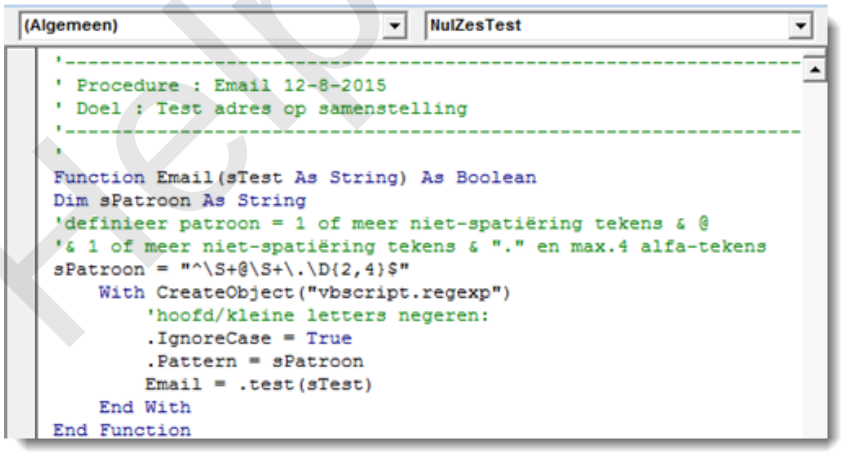
Het nummer kan als volgt worden opgevraagd:



E-mailadres

Over de samenstelling van een e-mailadres zijn veel misverstanden. Er wordt veel meer toegestaan dan gedacht. Meer informatie hierover vind je in deze (Engelse) wiki: en.wikipedia.org/wiki/Email_address#Valid_email_addresses.

Om te voorkomen dat er een correct adres wordt afgekeurd controleert de volgende functie alleen of het adres tekens voor en na de @ bevat, zonder dat daar spatiëring-tekens bij zitten (dat zijn bijvoorbeeld return, linefeed en spatie). Daarnaast wordt er gecheckt of er een punt aanwezig is en een toplevel domein (extensie) van twee tot vier letters. Zie dit voorbeeld:



Deze expressie is natuurlijk geheel aan de eigen wensen aan te passen. Voel je daartoe vooral uitgenodigd.

De functie kan als volgt in het werkblad worden geïmplementeerd. Bijvoorbeeld in cel D17 (invoer in D16):
=ALS(email(D16);"correct e-mailadres";ALS(LENGTE(D16)>0;"foutief e-mailadres";""))

BSN controle

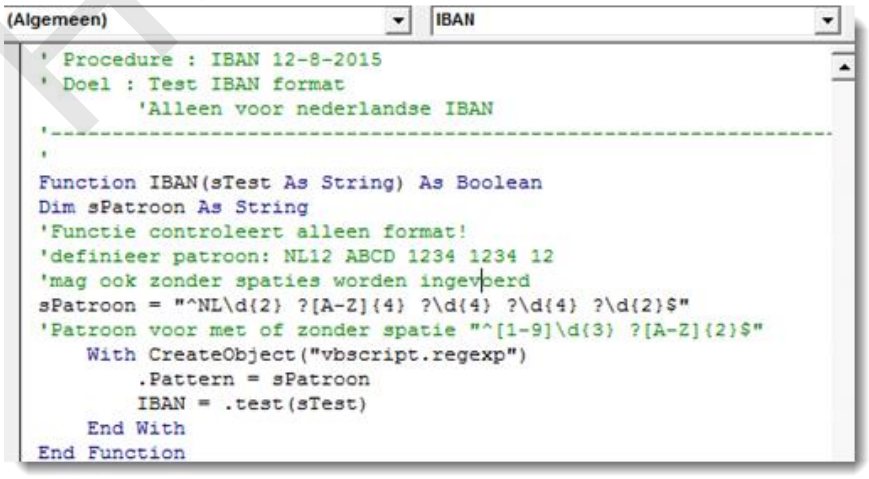
Met de controle van het format van een BSN ben je er niet. Een bestaande BSN dient aan een variant van de 11-proef te voldoen. Meer informatie hierover vind je in deze Wiki: nl.wikipedia.org/wiki/Burgerservicenummer

Dit kan als volgt worden uitgewerkt:

International Bank Account Number (IBAN)

Voor een IBAN geldt hetzelfde. Naast de controle op format is een rekenkundige controle nodig. Zie hiervoor: nl.wikipedia.org/wiki/International_Bank_Account_Number

In de volgende functie wordt het format gecontroleerd met behulp van de regular expression:



Met de volgende routine wordt de opgave van een IBAN gevraagd welke kan worden ingevoerd met of zonder spaties. Hierbij wordt ook de voorgeschreven rekenkundige controle uitgevoerd. Daarbij doet zich een probleem voor: bij het gebruik van de Mod operator met grote getallen krijg je een overloop fout. Deze wordt veroorzaakt doordat de grootte van de getallen de specificaties van Excel te boven gaan. Gelukkig kan dit op verschillende manieren worden omzeild. In de deze procedure wordt de

berekening daarom in verschillende stappen uitgevoerd. De routine is uitsluitend geschikt voor de controle van een Nederlandse IBAN:

Met deze laatste controle van een IBAN wordt deel 21 beëindigd.

Samenvatting

In deze aflevering werden de regular expressions behandeld. Voor VBA zijn deze beschikbaar in de Micrososft VBScript-bibliotheek. Hiermee is het mogelijk om op eenvoudige wijze invoermaskers te maken waarmee invoer op het juiste format is te controleren. Naast uitleg van het gebruik van de methoden, eigenschappen en speciale tekens uit deze bibliotheek werden de volgende voorbeelden uitgewerkt; de postcode, een mobielnummer, een e-mailadres, een BSN en een Nederlandse IBAN. Bij de laatste twee werd naast de controle op het format ook een rekenkundige controle gegeven om de juistheid van het nummer te kunnen verifiëren. Naast de invoer met een inputbox werd ook behandeld hoe je de functie in een werkblad kan opnemen.

De volgende aflevering

Volgende maand slaan we een keertje over. De inhoud van de aflevering daarna hangt af van reacties op de oproep die gedaan is in de vorige aflevering om onderwerpen aan te dragen. Dit kan nog steeds via het e-mailadres van de redactie; redactie@helpmij.nl.

Helpmij.nl