



NEWSLETTER

April 2017

Sehr geehrte Damen und Herren,

hier kommt unser zweiter Newsletter für das Jahr 2017.

Neben Tipps und Tricks aus den Bereichen Office und Technik finden Sie auch aktuelle Termine sowie weitere News, die für den Einen oder Anderen vielleicht von Interesse sind. Der nächste Newsletter erscheint im Juni 2017.

Wir wünschen Ihnen eine interessante Lektüre.

Ihr Team von Netz-Weise IT-Training

News

Netz-Weise baut um

Nach fast 5 Jahren sind wir den Kinderschuhen entwachsen und der Raum wurde zu eng, so dass wir seit Anfang April weitere Räume in unserem Gebäude dazu gemietet haben. Dieses zog einige Renovierungs- und Umbauarbeiten mit sich, zum Beispiel den Einbau einer Spindeltreppe zur Verbindung der beiden Etagen. Einige Impressionen hierzu finden Sie auf unserer [Homepage](#).

PowerShell User Group

Der Mai steht dieses Mal unter dem Vorzeichen XML und Verschlüsselung. Folgende Vorträge sind am 19.05.2017 hierzu geplant:

Vortrag 1 wird von Andreas Nick gehalten. Er wird uns anhand einiger Beispielscripte zeigen, wie man mit XML-Daten umgeht.
Vortrag 2 wird sich mit verschiedenen Verschlüsselungsmöglichkeiten auseinandersetzen, angefangen von Secure Strings über Public Key-Verschlüsselung zum Einsatz des Azure Key Vaults.
Außerdem werden wir wieder die Rubrik Bring your Script mit Christian Imhorst starten.

Wenn Sie nicht persönlich anwesend sein können, wird die Session auch über Skype verfügbar sein.

Termine

Veranstaltungen

01.06.2017 – IT-Jogging „Windows Server 2016 Storage Spaces Direct Cluster“ mit Marc Grote
19.05.2017 – PowerShell Usergroup-Treffen (jeden 3. Freitag im Monat)

Garantietermine (ein Auszug)

16.05.2017 – Excel 2010 – Grundlagen
24.05.2017 – Excel 2010 – Erweiterte Funktionalitäten nutzen
29.05.-02.06.2017 – Linux Systemadministration
29.05.-02.06.2017 – M20345-1 Administering MS Exchange Server 2016
12.06.-16.06.2017 – Administration von Windows Server 2012 / 2012R2
12.06.-16.06.2017 – VMware vSphere 6.5: Install, Configure, Manage
03.07.-07.07.2017 – PowerShell Kompaktkurs

Weitere Termine finden Sie auf unserer Homepage unter www.netz-weise-it.training

Tipps und Tricks –Office

Was ist PowerPivot?

(Autor: Dirk Röllecke)

Seit kurzem bieten wir einen Trainingstag PowerPivot für Excel 2013 und Excel 2016 an. Das Training wurde auch schon durchgeführt, wir sind der Meinung, dass es sich durchaus lohnen könnte, sich einen Tag mit diesem Excel-AddIn zu beschäftigen. Ähnlich wie beim Solver, der als schon lange AddIn in Excel zu haben ist, den aber kaum jemand wirklich kennt, kann es aber durchaus sein, dass PowerPivot Ihnen bislang kein Begriff ist. Das liegt sicherlich an der nicht ganz offensichtlichen Integration dieses AddIns in Excel. Nutzer der 2010er Excel-Version müssen sich PowerPivot noch als Download aus dem Internet besorgen. Tipps dazu hier:

<https://support.office.com/de-de/article/PowerPivot-Add-In-a9c2c6e2-cc49-4976-a7d7-40896795d045>

Mit den Versionen Excel 2013 und Excel 2016 wird PowerPivot dann als AddIn (Com-Add-In!) zur Verfügung gestellt. Soweit ich das beurteilen kann, aber nicht in jeder Version. Wir benutzen hier die Office 365 ProPlus Suite, da ist es integriert. Versucht man sich bei Microsoft schlau zu lesen, was genau PowerPivot zu leisten im Stande ist, bleiben gerne diverse Fragen offen. Zumindest wurde mir nicht ohne weiteres klar, wo genau die Unterschiede zwischen dem "normalen Exceln" und dem "Pivotisieren mit Power" nun genau liegen. Siehe dazu z.B. diese Webadresse:

<https://support.office.com/de-de/article/Power-Pivot-Leistungsstarke-Datenanalyse-und-Datenmodellierung-in-Excel-d7b119ed-1b3b-4f23-b634-445ab141b59b>

Dankenswerter Weise hat der Herdt Verlag nun eine Trainingsunterlage zu PowerPivot veröffentlicht (Thomas Käflein: PowerPivot in Excel professionell einsetzen. Excel 2013/2016. August 2016). Die benutzen wir auch als Trainingsunterlage.

In deutscher Sprache sind zu genau diesem Thema auch (noch) nicht allzu viele Werke erhältlich, bei nicht allzu intensivem Googeln schaffe ich es auf zwei bis drei Buchtitel, die sich ausschließlich mit PowerPivot beschäftigen. Auf YouTube hat sich in deutscher Sprache bislang Andreas Thehos in diversen Tutorials mit diesem Thema beschäftigt.

Was kann aber jetzt PowerPivot, was Excel nicht kann?

Erstens können Sie mit PowerPivot Tabellen in ein sogenanntes Datenmodell laden, die weit mehr als die für Excel Arbeitsblätter höchstens möglichen 1 MIO Datensätze enthalten. Die Obergrenze für solche Tabellen wird nicht in der Anzahl der Datensätze, sondern mit der höchstmöglichen Speicherkapazität von 2GB angegeben. Die Tabellen, die Sie in das Datenmodell laden, können in der vorliegenden Excel-Datei enthalten sein, dürfen aber auch aus externen Quellen stammen und mannigfache Formate haben.

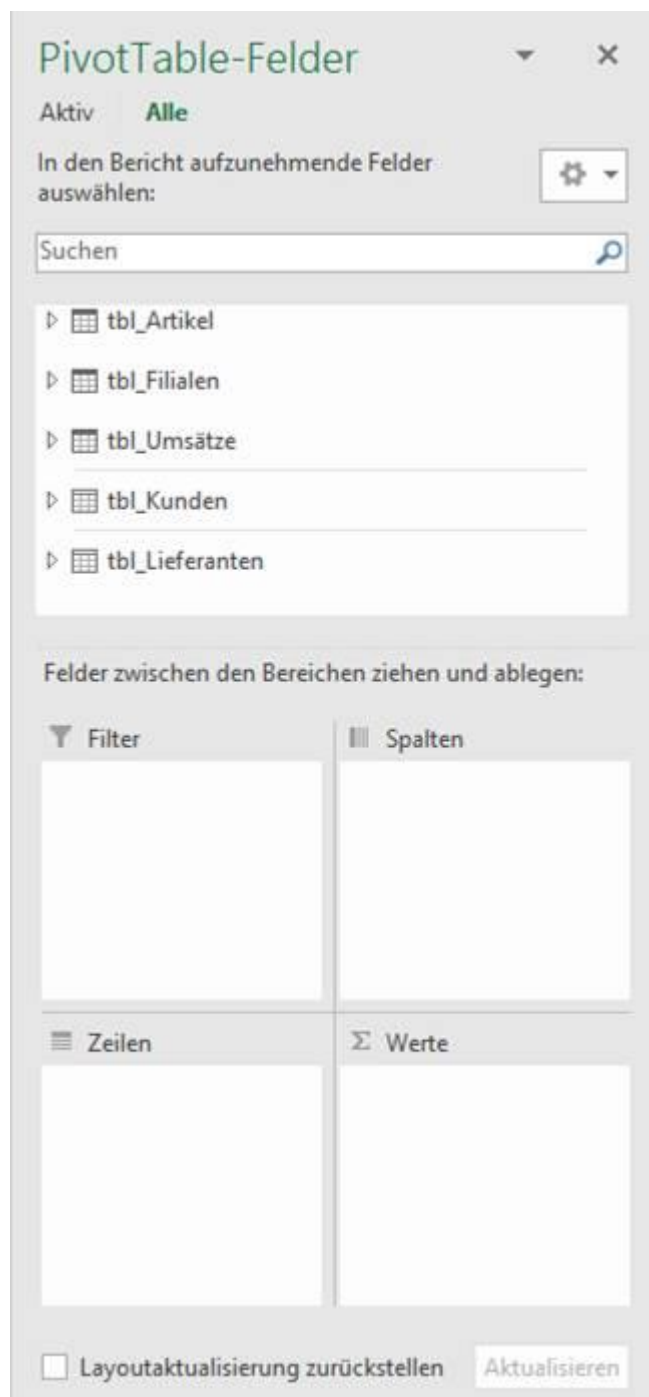
Zweitens können Sie in PowerPivot-Tabellen berechnete Felder (Measures) und berechnete Spalten mit anderen Funktionen als in Excel anlegen. Diese sogenannten DAX-Funktionen (Data Analysis Expressions) ticken in Englisch und ähneln damit eher den Funktionen in Access. Microsoft verspricht uns performanteres Rechnen in der PowerPivot-Umgebung. Das ist allerdings bei meinen Beispieltabellen schwierig messbar. Wir wollen es mal glauben.

Drittens kann man mit PowerPivot Beziehungen darstellen. Es gibt in PowerPivot eine "Diagrammsicht", die dem Access-Beziehungsfenster ähnelt, und dort kann man zwischen Tabellen 1-zu-n-Beziehungen herstellen. Dazu braucht man allerdings zumindest Grundkenntnisse in der Erstellung relationaler Datenbanken. Und gut vorbereitete Tabellen (Primärschlüssel- und Fremdschlüsselfelder usw.)!

Viertens kann ich auf die im PowerPivot Datenmodell abgelegten Tabellen, die gemeinsam mit der Excel-Datei abgespeichert, aber in einem eigenen Programmfenster bearbeitet werden, per Pivot-Tabelle gleichzeitig zugreifen! Ich sehe also eine Feldliste mit mehreren Tabellen, aus denen ich eine Pivot-Tabelle zusammenstellen kann. (Siehe Abbildung!). Microsoft scheint diese Möglichkeit so überzeugend zu sein, dass man seit Access 2013 keine Pivot Ansichten mehr in diesem Datenbankprogramm findet.

Man wird in PowerPivot noch mit einigen neuen Begriffen versorgt, die aber nicht immer neues enthalten. So würde ich den "Key-Performance-Indicator" (KPI) eher als abgewandelte bedingte Formatierung einschätzen und die "Perspektive" entspricht in etwa dem, was man in älteren Excel-Versionen als benutzerdefinierte Ansicht kannte.

Falls Sie einen der vier oben genannten Aspekte für Ihre Arbeit als zuträglich empfinden sollten, könnte es sein, dass wir uns demnächst in einem Training "PowerPivot für Excel 2013/2016" wiedersehen werden. Ich würde mich freuen!



Tipps und Tricks – Technik

Komplexe Kennwörter generieren in Powershell

(Autor: Holger Voges)

Kennwörter braucht man in Skripten an allen Ecken und Enden, speziell dann, wenn man Benutzer erstellen möchten. Bei einer größeren Anzahl von Konten kann da ein Skript ganz hilfreich sein, dass zufällige Kennwörter generiert. Das ist mit Powershell relativ einfach gemacht. Alles, was man benötigt, ist eine Funktionalität zum Erstellen von Zufallswerten und die Möglichkeit, Zahlen in Buchstaben umzuwandeln. Das Cmdlet Get-Random liefert Zufallszahlen zurück:

```
Get-Random -Min 10 -Max 100
```

Die vollständige Auflistung der Funktionen Get-Random können Sie in der Powershell-Hilfe nachlesen. Für uns interessant ist der Parameter -InputObject, über den man Get-Random ein Array mit Werten übergeben kann, aus denen der Zielwert gewählt wird.

```
$ZufallsListe = 65..90  
$Zufallszahl = Get-Random -InputObject $Zufallsliste
```

Die Zahlen 65-90 entsprechen den Ascii-Codes der Großbuchstaben von A-Z. Wir können die Zahlen in einen ASCII-Code umwandeln, indem wir Sie in ein CHAR[]-Array konvertieren.

```
[CHAR[]]$Zufallszahl
```

Wenn man den Code in einer Schleife ausführt, bekommt man ein zufällig generiertes Kennwort:

```
$Length = 12  
$Zufallsliste = 65..90  
For ( $i = 0; $i -lt 12; $i++)  
{  
    $password += Get-Random -InputObject $Zufallsliste  
}
```

Jetzt kann man noch weitere die Zahlenrepräsentationen für Kleinbuchstaben, Zahlen und Sonderzeichen hinzufügen und alles in eine Funktion packen, und man hat "New-Password":

```
function New-Password {  
  
    [cmdletBinding()]  
    Param  
    (  
        # The Length of the Password  
        [ValidateRange(3,256)]  
        [int]$Length = 12,  
  
        # The Criteria the Password must fulfill. Only Complex enforces the combination of  
        # different Criteria.  
        [ValidateSet('Complex','Numbers','Specials','LowerCaseLetters','UpperCaseLetters')]  
        [String[]]$Type = 'Complex'  
    )  
  
    # ASCII-Codes for different Types of Characters as Arrays  
    $Specials = ( 33..45 ) + ( 58..64 ) + ( 91..95 )  
    $Numbers = ( 48..57 )  
    $UpperCaseLetters = 65..90  
    $LowerCaseLetters = 97.. 122  
  
    # Generate the Array $Charset which contains the Characters from all chosen Criteria  
    Switch ( $Type )
```

```

{
'Complex'      { $Charset = $UpperCaseLetters + $LowerCaseLetters + $Numbers + $Specials; break }
'UpperCaseLetters' { $Charset += $UpperCaseLetters }
'LowerCaseLetters' { $Charset += $LowerCaseLetters }
'Numbers'      { $Charset += $Numbers }
'Specials'     { $Charset += $Specials }
}
Write-Verbose "$Charset"

# Generate the password. The While-Loop ensures that Complex Passwords fullfill at least 3 Criteria.
While ((( $UpperInserted + $LowerInserted + $SpecialsInserted + $NumbersInserted -lt 3 ) -and ( $Type -eq 'Complex' )) `
-or (( $Type -ne 'Complex' ) -and ( $UpperInserted + $LowerInserted + $SpecialsInserted + $NumbersInserted -lt 1 )))

{
# Count-Variables are for reporting purposes
# Inserted-Variables ensure that complex Passwords fullfill always at least 3 Criteria.
# They are checked in the While-Condition.
$UpperCount = 0
$LowerCount = 0
$SpecialsCount = 0
$numbersCount = 0
$UpperInserted = 0
$LowerInserted = 0
$SpecialsInserted = 0
$NumbersInserted = 0
[string]$Password = ""

# Generating the Password
For ( $i=1; $i -le $Length; $i++ )
{
$RandomChar = (Get-Random -InputObject $CharSet)
Switch ( $RandomChar )
{
{ $RandomChar -in $UpperCaseLetters } { $UpperCount++; $UpperInserted = 1 }
{ $RandomChar -in $LowerCaseLetters } { $LowerCount++; $LowerInserted = 1 }
{ $RandomChar -in $Specials } { $SpecialsCount++; $SpecialsInserted = 1 }
{ $RandomChar -in $Numbers } { $NumbersCount++; $NumbersInserted = 1 }
}
[string]$Password += [char[]]$RandomChar
}
}

$Password
Write-Verbose "UpperCaseCount: $UpperCount"
Write-Verbose "LowerCaseCount: $LowerCount"
Write-Verbose "Specials: $SpecialsCount"
Write-Verbose "Numbers: $numbersCount"
}

```

Aber es geht auch einfacher, denn das .NET-Framework bietet bereits eine fertige statische Methode zum Generieren von Kennwörtern, die man in Powershell nur noch laden muss. Die Methode heißt GeneratePassword und befindet sich in der [System.Web]-Assembly. Die Assembly ist in Powershell allerdings nicht standardmäßig geladen. Dies kann man mit dem Cmdlet Add-Type erledigen:

```
Add-Type -AssemblyName System.web
```

Nun können Sie die Methode GeneratePassword aus der Klasse [System.Web.Security.Membership] aufrufen. Der Aufruf erfolgt über [Klasse]::Methodenname(Parameter):

```
[System.Web.Security.Membership]::GeneratePassword(10,3)
```

Die beiden Parameter 10 und 3 bestimmen die Anzahl der Zeichen sowie die nicht-Alphanumerischen Zeichen, die das Kennwort enthalten muss.

In eine Funktion gegossen sieht das dann so aus:

```
function New-Password {  
    param  
    (  
        # The Length of the Password  
        [ValidateRange(3,128)]  
        [String]$Length = 12,  
        # Number of non-Alphanumeric Characters  
        [String]$Specials = '3'  
    )  
    Add-Type -AssemblyName System.web  
    [System.Web.Security.Membership]::GeneratePassword($Length,$Specials)  
}
```

SQL-Server Management Studio mit erweiterten Intellisense Funktionen erweitern über das kostenlose ApexSQL AutoComplete und ApexSQL Refactor

Das SQL-Server Management-Studio bringt bereits eine rudimentäre Form von Intellisense mit, also das automatische Erkennen von SQL-Objekten. Es geht aber noch deutlich besser. Schauen Sie sich z.B. das kostenlose ApexSQL Complete an. Es ergänzt das Sql-Server Managementstudio um so hilfreiche Funktionen wie das automatische auflösen von Wildcards in Spaltennamen (*), das Durchsuchen von Resultsets und Autocomplete für SQL-Statements.

Mit dem ebenfalls kostenlosen Refactor können Sie Ihre Skripte außerdem automatisch formatieren lassen. Dadurch werden unübersichtliche Skripte in deutlich besser lesbare Varianten umgewandelt.

Beide Tools und noch mehr können Sie unter <https://www.apexsql.com/free> herunterladen.

Und wenn Ihnen unser Newsletter gefällt, laden Sie doch auch Ihre Kollegen und Freunde zu uns ein.

Mit freundlichen Grüßen

Ihr Team von

Netz-Weise IT-Training

Netz-Weise IT-Training

Inhaber: Holger Voges

Freundallee 13a
D-30173 Hannover

E-Mail: info@netz-weise.de

Internet: www.netz-weise.de

Telefon +49 511 165 925 0

Fax +49 511 165 925 99



Abmelden von Einladungen:

Sie möchten von uns keine Einladungen mehr per Mail erhalten?

Klicken Sie bitte [hier](#) und wir entfernen Ihre Mail-Adresse aus unserem Verteiler.