Welcome to SQL Saturday Denmark

JSON & XML SQL Server 2016

since SQL 2016 it is import
to weight respective advantages

Thanks you our PLATINUM sponsors

Thanks you our GOLD and SILVER sponsors

Microsoft

SQL POWERHOUSE

SCALES

PURESTORAGE by CLOUDIO

NEXTAGENDA
BUSINESS INTELLIGENCE LØSNINGER

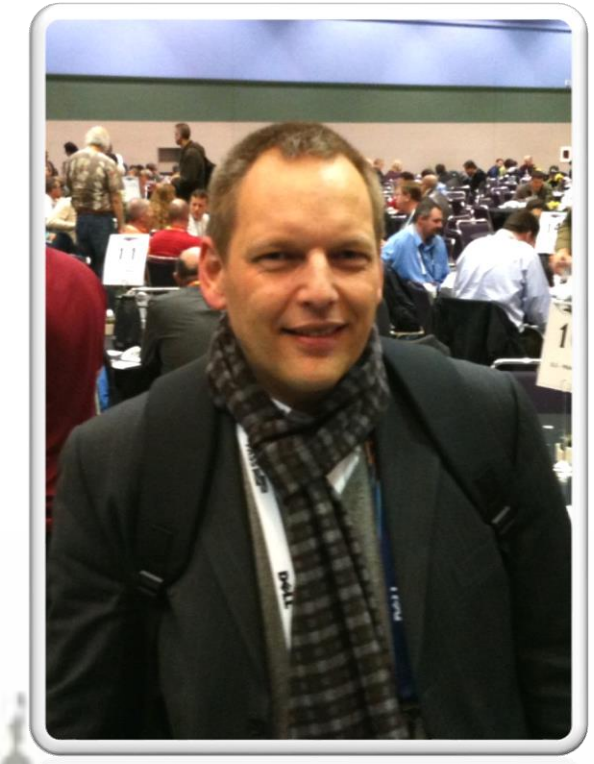
SQLSENTRY

COZYROC™
Go to the next level

imshealth™ | Rehfeld

PASS SQL saturday

About me



Alexander Karl

.net - CDE

SQL + BI Consultant

Microsoft CERTIFIED Trainer

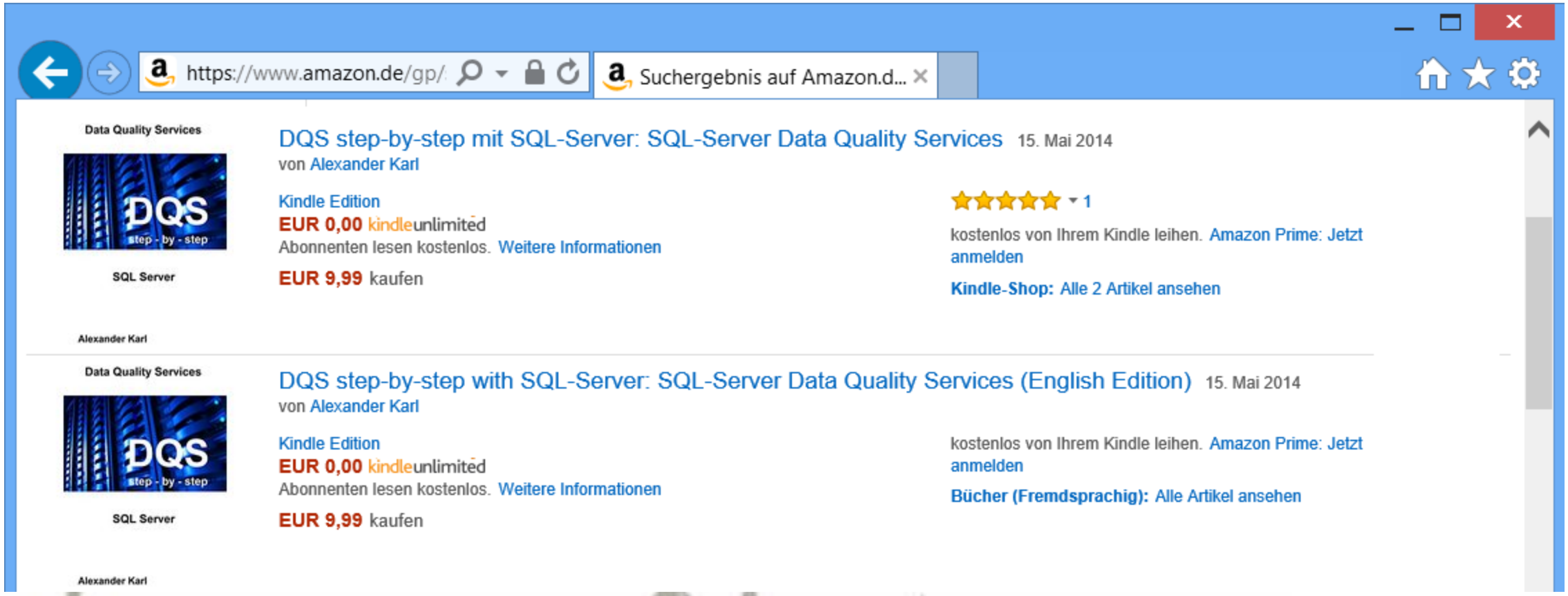
Microsoft CERTIFIED Solutions Associate SQL Server 2012/2014

Microsoft CERTIFIED Solutions Expert Data Platform

PASS SQL saturday

publications

publications



Agenda

- JSON „new feature in SQL Server 2016“
- appropriate tools
- syntax For JSON / For XML
- syntax OpenJSON / OpenXML
- datatype ? JSON vs. XML
- indexes on datatype

PASS SQL saturday

{JSON} „round trip“

Built-in functions
ISJSON
JSON_VALUE
JSON_QUERY

OPENJSON
Transforms JSON text to table

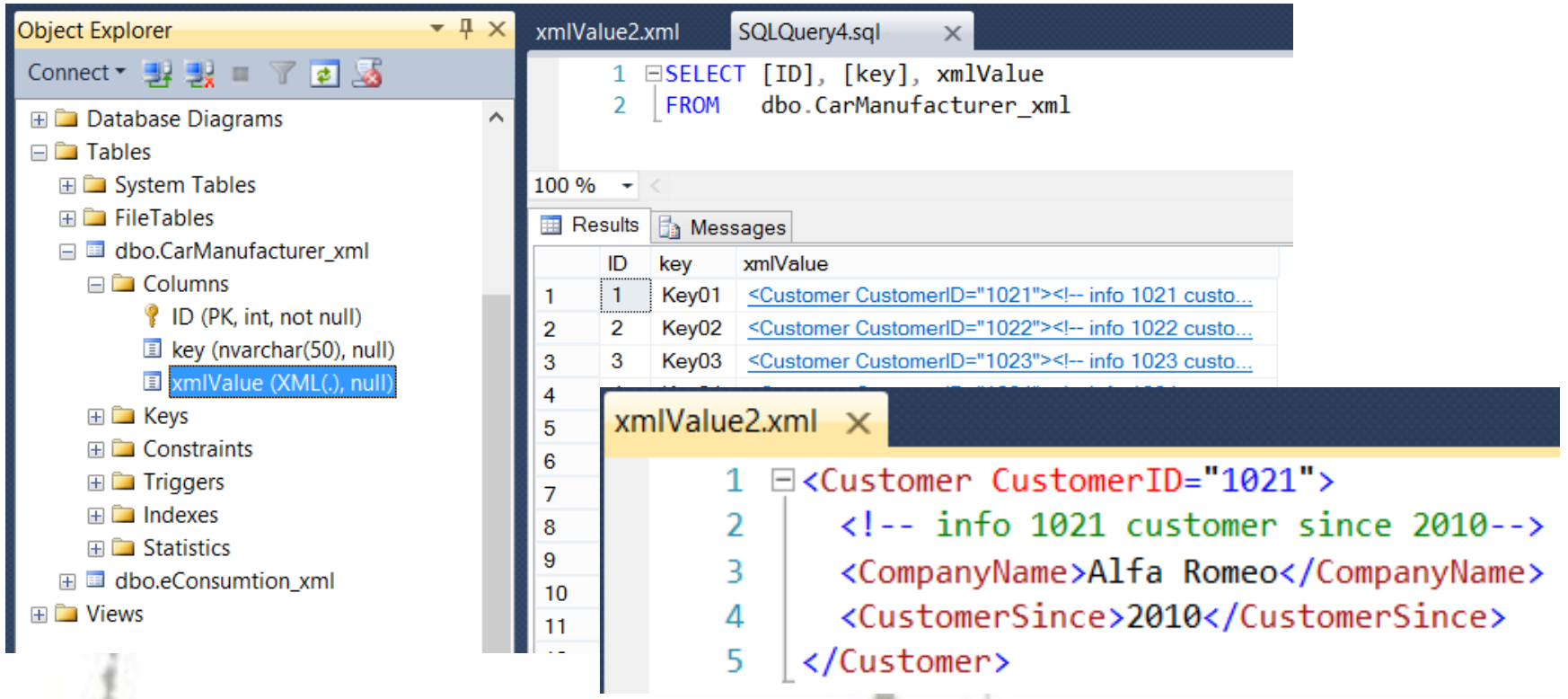
```
[
  {
     "Number":"SO43659",
     "Date":"2011-05-31T00:00:00"
     "AccountNumber":"AW29825",
     "Price":59.99,
     "Quantity":1
   },
   {
     "Number":"SO43661",
      "Date":"2011-06-01T00:00:00"
       "AccountNumber":"AW73565",
       "Price":24.99,
        "Quantity":3
   }
]
```

Number	Date	Customer	Price	Quantity
SO43659	2011-05-31T00:00:00	MSFT	59.99	1
SO43661	2011-06-01T00:00:00	Nokia	24.99	3

FOR JSON
Formats result set as JSON text.

https://msdn.microsoft.com/en-us/library/dn921897.aspx

<xml/> tooling


Object Explorer
Connect
Database Diagrams
Tables
System Tables
FileTables
dbo.CarManufacturer_xml
Columns
ID (PK, int, not null)
key (nvarchar(50), null)
xmlValue (XML(.), null)
Keys
Constraints
Triggers
Indexes
Statistics
dbo.eConsumtion_xml
Views
xmlValue2.xml
SQLQuery4.sql
1 SELECT [ID], [key], xmlValue
2 FROM dbo.CarManufacturer_xml
100 %
Results
Messages
ID key xmlValue
1 1 Key01 <Customer CustomerID="1021"><!-- info 1021 custo...
2 2 Key02 <Customer CustomerID="1022"><!-- info 1022 custo...
3 3 Key03 <Customer CustomerID="1023"><!-- info 1023 custo...
xmlValue2.xml
1 <Customer CustomerID="1021">
2 <!-- info 1021 customer since 2010-->
3 <CompanyName>Alfa Romeo</CompanyName>
4 <CustomerSince>2010</CustomerSince>
5 </Customer>

{JSON} tooling

```
1  SELECT name                  as 'name'
2       , database_id           as 'database_id'
3       , create_date           as 'create_date'
4       , is_published          as 'replication.is_published'
5       , is_subscribed         as 'replication.is_subscribed'
6       , is_merge_published    as 'replication.is_merge_published'
7       , is_distributor        as 'replication.is_distributor'
8  FROM   sys.databases
9  FOR    JSON  PATH, ROOT('databases'), Include_NULL_Values;
10
```

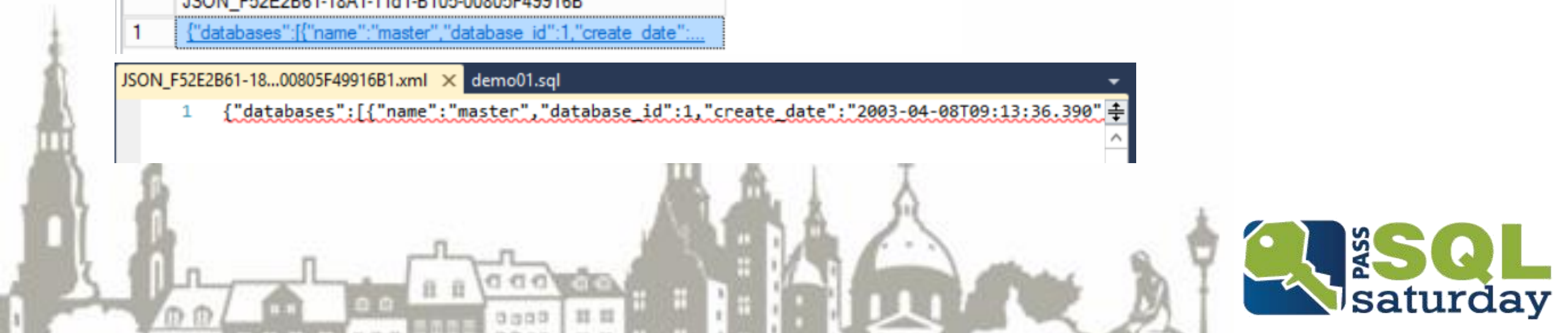
100 %

Results | Messages

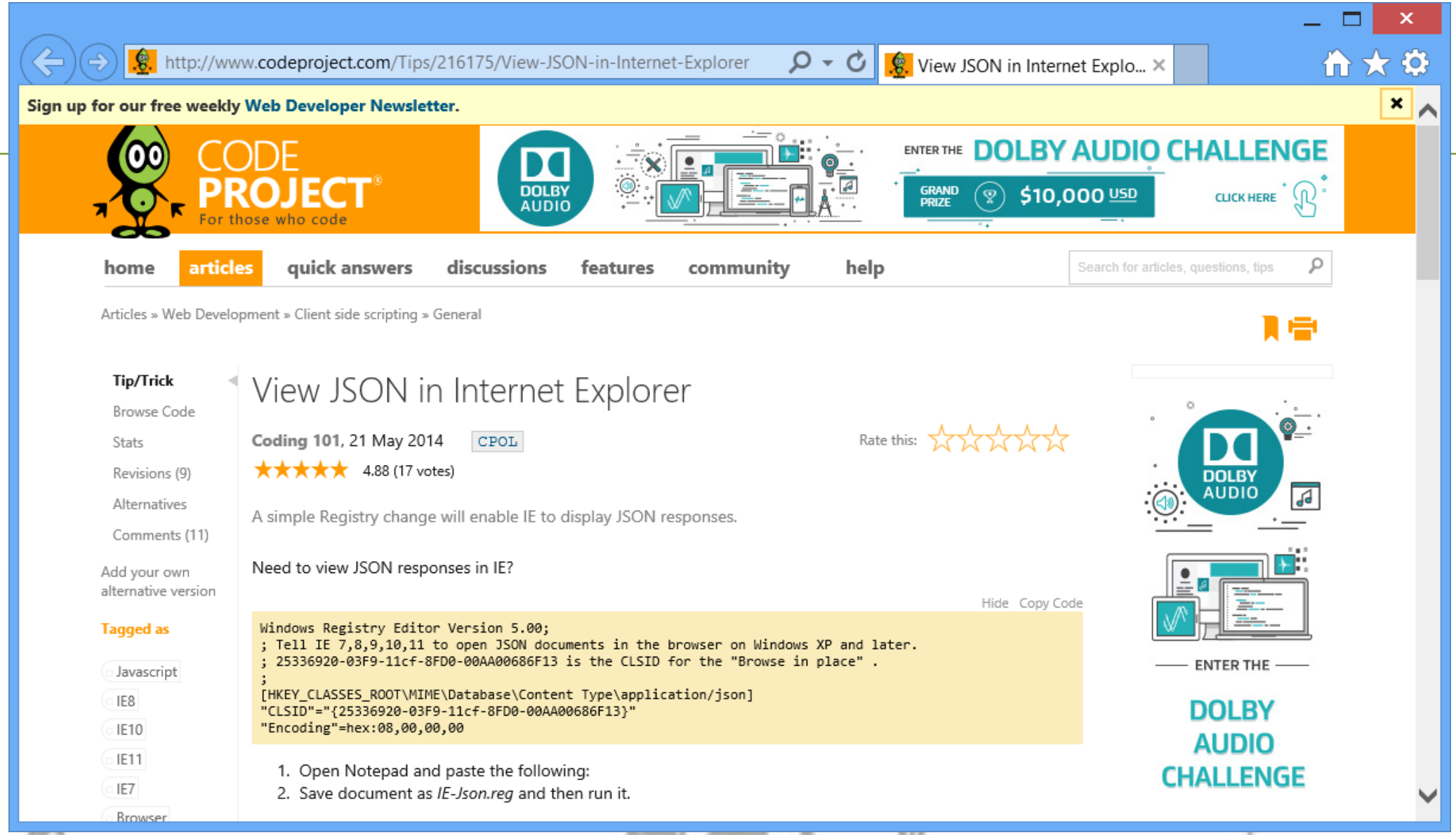
	JSON_F52E2B61-18A1-11d1-B105-00805F49916B
1	{"databases":[{"name":"master","database_id":1,"create_date":...

JSON_F52E2B61-18...00805F49916B1.xml | demo01.sql

```
1  {"databases":[{"name":"master","database_id":1,"create_date":"2003-04-08T09:13:36.390"
```

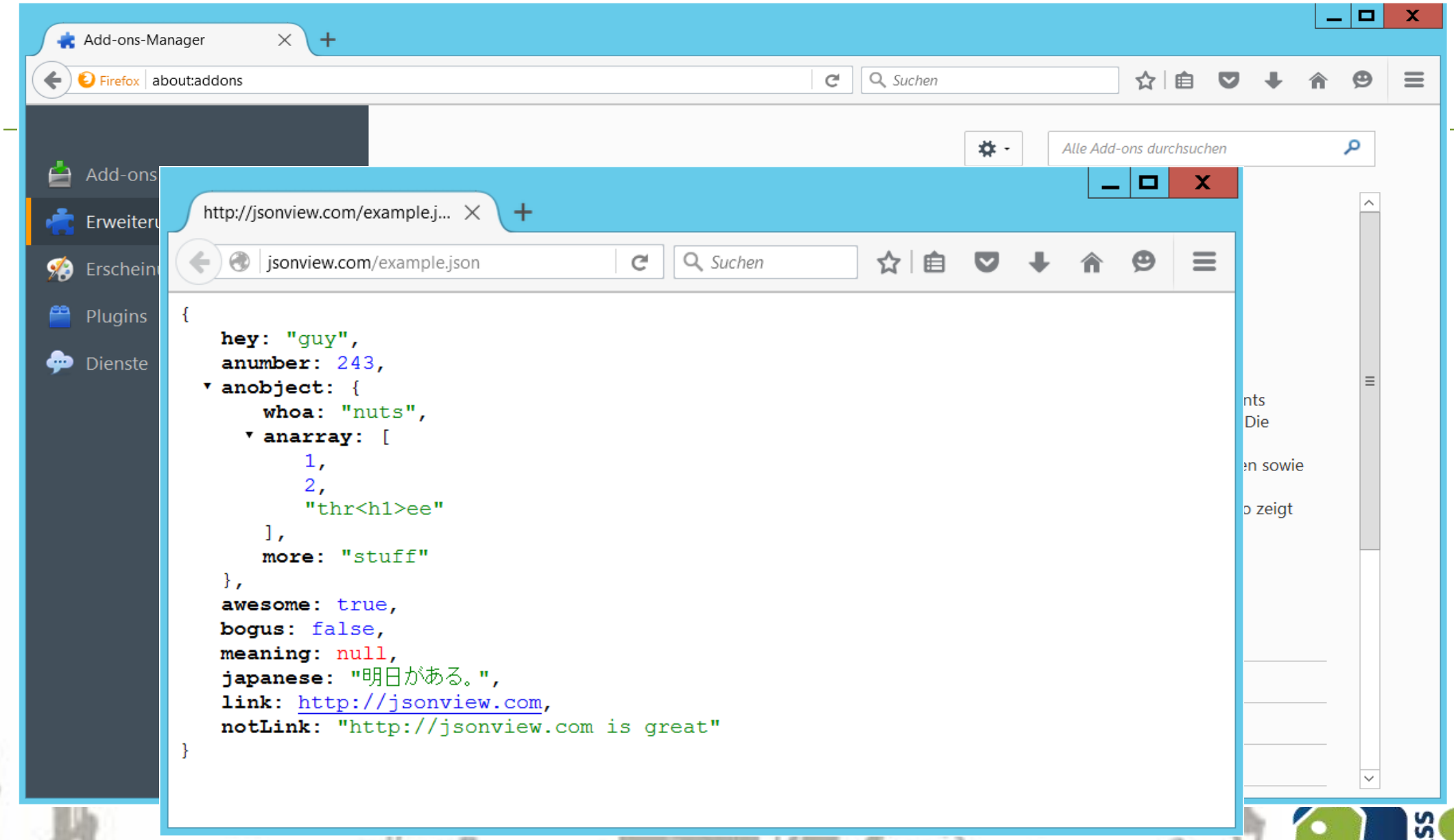


PASS SQL saturday

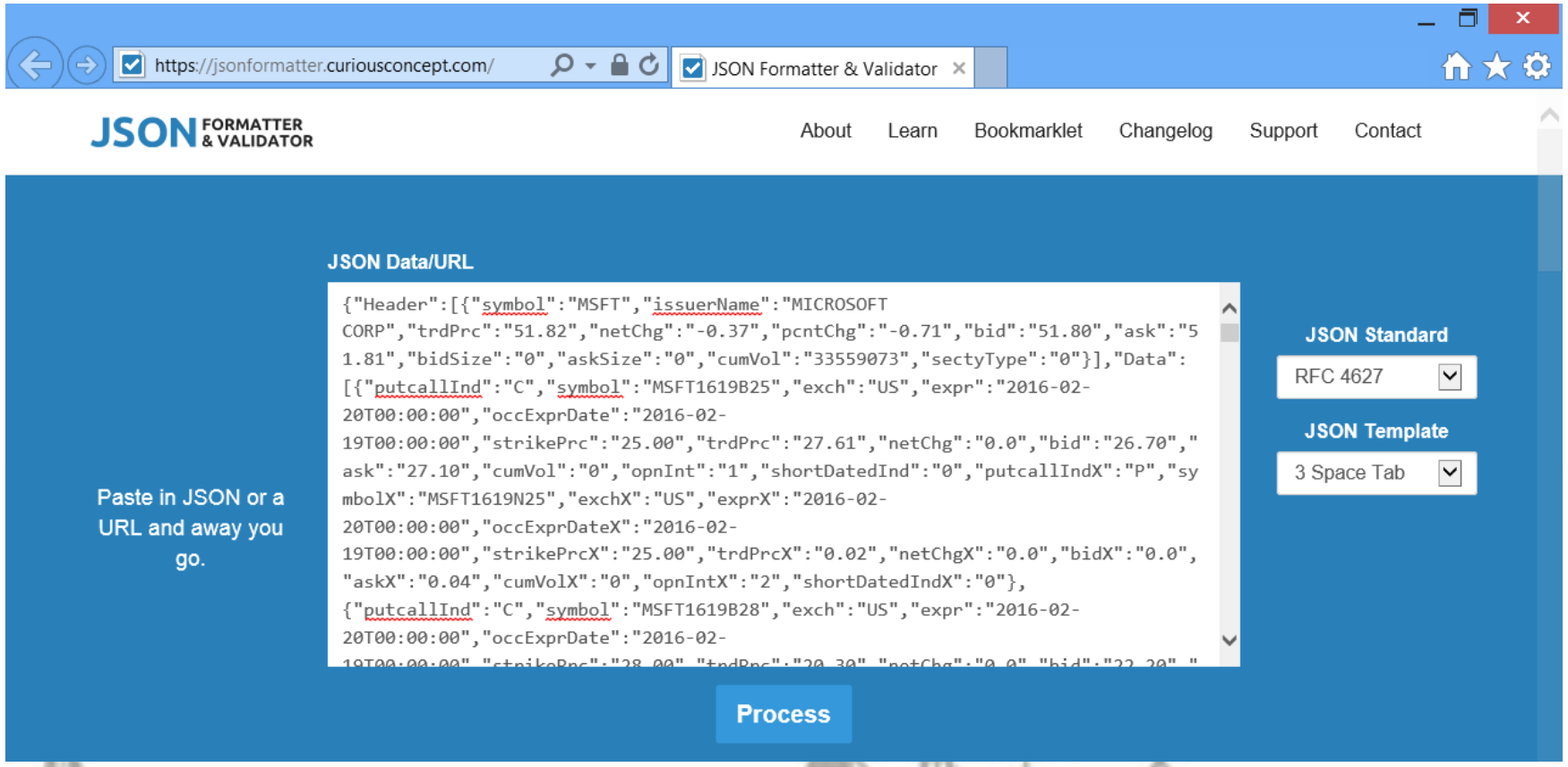

http://www.codeproject.com/Tips/216175/View-JSON-in-Internet-Explorer
View JSON in Internet Explo...
Sign up for our free weekly Web Developer Newsletter.
CODE PROJECT
For those who code
DOLBY AUDIO
ENTER THE DOLBY AUDIO CHALLENGE
GRAND PRIZE $10,000 USD
CLICK HERE
home
articles
quick answers
discussions
features
community
help
Search for articles, questions, tips
Articles » Web Development » Client side scripting » General
Tip/Trick
Browse Code
Stats
Revisions (9)
Alternatives
Comments (11)
Add your own alternative version
Tagged as
Javascript
IE8
IE10
IE11
IE7
Browser
View JSON in Internet Explorer
Coding 101, 21 May 2014 CPOL
4.88 (17 votes)
Rate this:
A simple Registry change will enable IE to display JSON responses.
Need to view JSON responses in IE?
Hide Copy Code
Windows Registry Editor Version 5.00;
; Tell IE 7,8,9,10,11 to open JSON documents in the browser on Windows XP and later.
; 25336920-03F9-11cf-8FD0-00AA00686F13 is the CLSID for the "Browse in place" .
;
[HKEY_CLASSES_ROOT\MIME\Database\Content Type\application/json]
"CLSID"="{25336920-03F9-11cf-8FD0-00AA00686F13}"
"Encoding"=hex:08,00,00,00
1. Open Notepad and paste the following:
2. Save document as IE-Json.reg and then run it.
DOLBY AUDIO
ENTER THE
DOLBY AUDIO CHALLENGE


http://www.codeproject.com/Tips/216175/View-JSON-in-Internet-Explorer
View JSON in Internet Explo...
Sign up for our free weekly Web Developer Newsletter.
CODE PROJECT
For those who code
DOLBY AUDIO
ENTER THE DOLBY AUDIO CHALLENGE
GRAND PRIZE $10,000 USD
CLICK HERE
http://jsonview.com/example.json
jsonview.com
{"hey": "guy","anumber": 243,"anobject": {"whoa": "nuts","anarray": [1,2,"thr
ee"], "more":"stuff"},"awesome": true,"bogus": false,"meaning": null,
"japanese":"明日がある。", "link": "http://jsonview.com", "notLink":
"http://jsonview.com is great"}
Tagged as
Javascript
IE8
IE10
IE11
IE7
Browser
Hide Copy Code
Windows Registry Editor Version 5.00;
; Tell IE 7,8,9,10,11 to open JSON documents in the browser on Windows XP and later.
; 25336920-03F9-11cf-8FD0-00AA00686F13 is the CLSID for the "Browse in place" .
;
[HKEY_CLASSES_ROOT\MIME\Database\Content Type\application/json]
"CLSID"="{25336920-03F9-11cf-8FD0-00AA00686F13}"
"Encoding"=hex:08,00,00,00
1. Open Notepad and paste the following:
2. Save document as IE-Json.reg and then run it.
ENTER THE
DOLBY AUDIO CHALLENGE

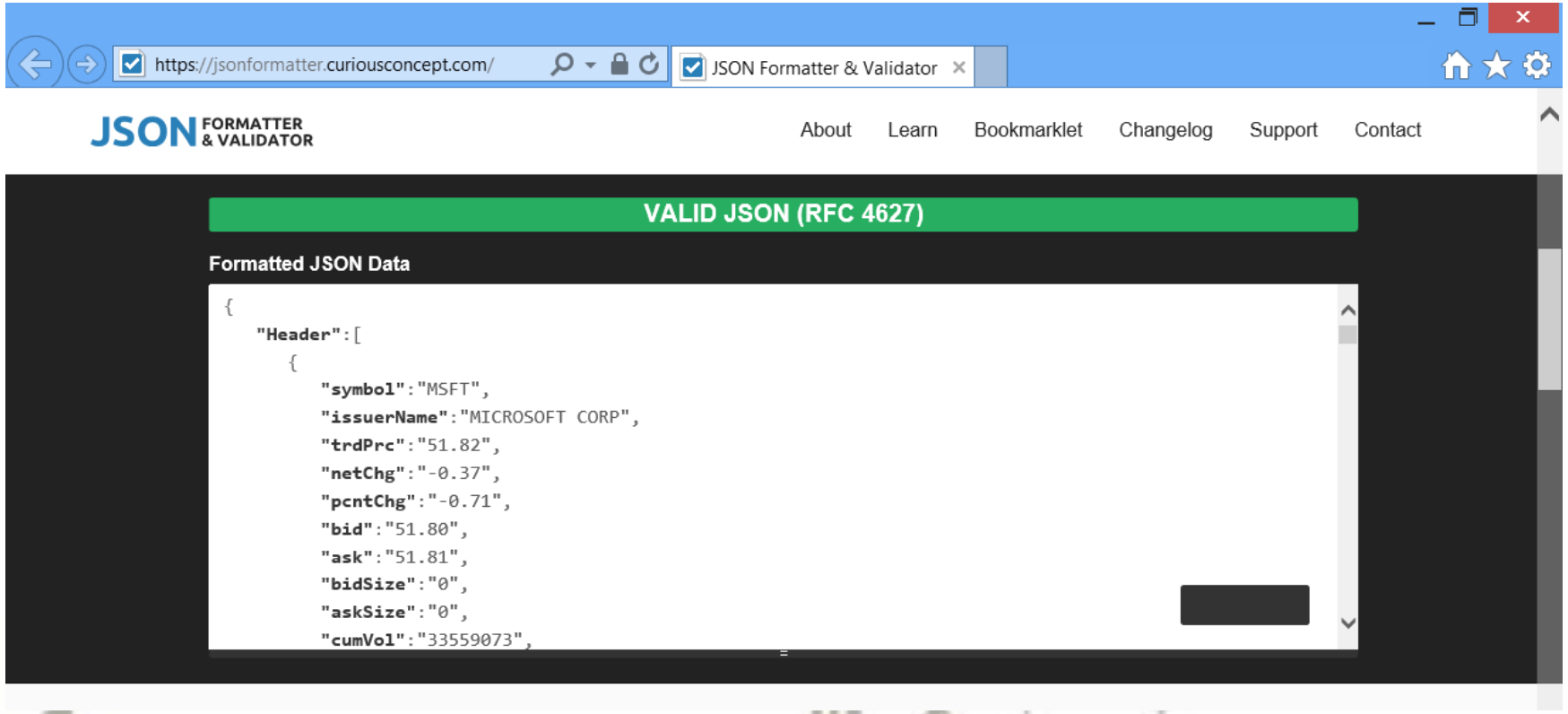

Add-ons-Manager
Firefox about:addons
Suchen
Alle Add-ons durchsuchen
Add-ons suchen
Erweiterungen
Erscheinungsbild
Plugins
Dienste
JSONView 1.1.0
Von Ben Hollis
JSON-Dokumente im Browser anzeigen.
Normalerweise bietet Firefox beim Aufrufen eines JSON-Dokuments (Content-Type "application/json") das Abspeichern der Datei an. Die JSONView-Erweiterung stellt JSON-Dokumente analog wie XML-Dokumente dar. Das Dokument verfügt über Formatierungen sowie Hervorhebungen und Felder und Objekte können erweitert und zusammengefasst werden. Enthält das JSON-Dokument Fehler, so zeigt JSONView nur den Quelltext an.
Um JSONView in Aktion zu sehen, rufen Sie nach der Installation http://benhollis.net/software/jsonview/example.json auf.
Automatische Updates
Standard
Ein
Aus
Zuletzt aktualisiert
Sunday, February 21, 2016
Homepage
http://jsonview.com/
Bewertung
143 Bewertungen


Add-ons-Manager
Firefox about:addons
Suchen
Alle Add-ons durchsuchen
Add-ons
Erweiteru
Erschein
Plugins
Dienste
http://jsonview.com/example.j...
jsonview.com/example.json
Suchen
{
hey: "guy",
anumber: 243,
anobject: {
whoa: "nuts",
anarray: [
1,
2,
"thr<h1>ee"
],
more: "stuff"
},
awesome: true,
bogus: false,
meaning: null,
japanese: "明日がある。",
link: http://jsonview.com,
notLink: "http://jsonview.com is great"
}

{JSON} Formatter & Validator



{JSON} Formatter & Validator



Demo

PASS SQL saturday

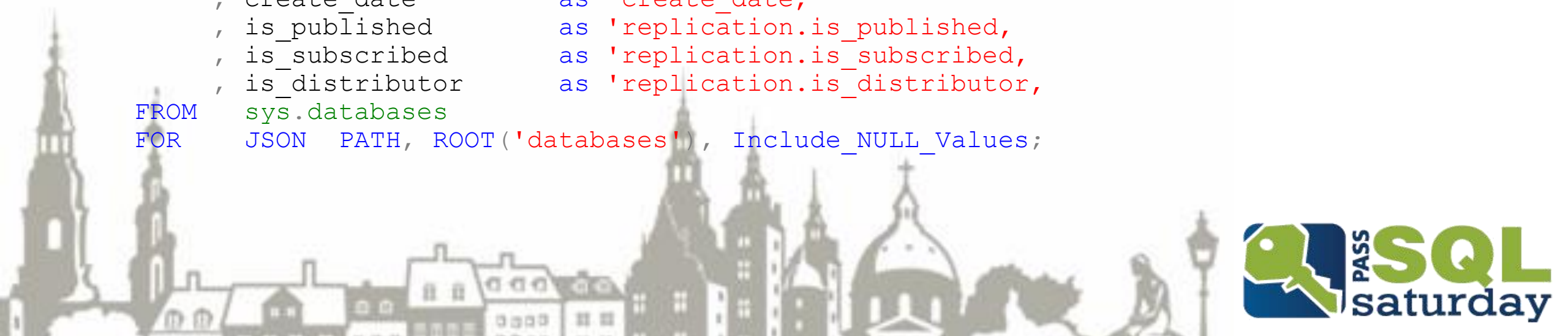
{JSON} FOR JSON

- AUTO Mode

```
SELECT name, database_id, create_date
FROM   sys.databases
FOR    JSON  Auto
```

- PATH Mode

```
SELECT name                 as 'name,
     , database_id          as 'database_id,
     , create_date          as 'create_date,
     , is_published         as 'replication.is_published,
     , is_subscribed        as 'replication.is_subscribed,
     , is_distributor       as 'replication.is_distributor,
FROM   sys.databases
FOR    JSON  PATH, ROOT('databases'), Include_NULL_Values;
```



PASS SQL saturday

{JSON} FOR JSON

- AUTO Mode

```
SELECT
FROM
FOR
```

file:///C:/ |. /JSON_demo/autoMode.json


```
[
  ▾ {
        name: "master",
        database_id: 1,
        create_date: "2003-04-08T09:13:36.390"
    },
  ▾ {
        name: "tempdb",
        database_id: 2,
        create_date: "2016-02-24T11:40:54.827"
    },
  ▾ {
        name: "model",
        database_id: 3,
        create_date: "2003-04-08T09:13:36.390"
    },
```


- PATH Mode

```
SELECT
      ,
      ,
      ,
      ,
      ,
FROM
FOR
```

```
ublished,
ubscribed,
istributor,

JLL_Values;
```

PASS SQL saturday

{JSON} FOR JSON

- AU

```
SELE
FROM
FOR
```

file:///C:/ ... /JSON_demo/autoMode.json

- PA

```
SELE
```

file:///C:/ ... /JSON_demo/pathMode.json

```
FROM
FOR
```


```
{
  databases: [
    {
      name: "master",
      database_id: 1,
      create_date: "2003-04-08T09:13:36.390",
      replication: {
        is_published: false,
        is_subscribed: false,
        is_merge_published: false,
        is_distributor: false
      }
    },
    {
      name: "tempdb",
      database_id: 2,
      create_date: "2016-02-24T11:40:54.827",
      replication: {
        is_published: false,
        is_subscribed: false,
        is_merge_published: false,
        is_distributor: false
      }
    },
```


PASS SQL saturday

<xml/> FOR XML

- AUTO Mode

```
SELECT name, database_id, create_date
FROM   sys.databases
FOR    XML  Auto ;
```

- PATH Mode

```
SELECT name                 as 'name,
     , database_id          as 'database_id,
     , create_date          as 'create_date,
     , is_published         as 'replication/is_published,
     , is_subscribed        as 'replication/is_subscribed,
     , is_distributor       as 'replication/is_distributor,
FROM   sys.databases
FOR    XML  PATH('db'), ROOT('databases');
```

PASS SQL saturday

{JSON} OpenJSON

```
DECLARE @json NVARCHAR(MAX)
SET @json ='{
        "Name"      :"PASS Deutschland e.V.",
        "addInfo"   : null,
        "ID"        : 828,
        "Current"   : true,
        "Skills"    :["SQL","SSIS","SSRS",42,"MDX"],
        "Region"    :{"Country":"Germany","Territory":"Hessen"}
        }';

SELECT * -- [key], [value], type
FROM   OpenJSON(@json);

SELECT *
FROM   OpenJSON(@json, '$.Skills')

SELECT *
FROM   OpenJSON(@json, '$.Region')
```

PASS SQL saturday

{JSON} OpenJSON

```
DECLARE @json NVARCHAR(MAX)
SET @json ='{
        "Name"
        "addInfo"
        "ID"
        "Current"
        "Skills"
        "Region"
        }';

SELECT * -- [key]
FROM   OpenJSON((

SELECT *
FROM   OpenJSON((

SELECT *
FROM   OpenJSON((
```

Results | Messages

	key	value	type
1	Name	PASS Deutschland e.V.	1
2	addInfo	NULL	0
3	ID	828	2
4	Current	true	3
5	Skills	["SQL","SSIS","SSRS",42,"MDX"]	4
6	Region	{"Country":"Germany","Territory":"Hessen"}	5

	key	value	type
1	0	SQL	1
2	1	SSIS	1
3	2	SSRS	1
4	3	42	2
5	4	MDX	1

	key	value	type
1	Country	Germany	1
2	Territory	Hessen	1

PASS SQL saturday

{JSON} OpenJSON Data Type

```
1  DECLARE @json NVARCHAR(MAX)
2  SET @json = '{
3                 "Name"     :"PASS Deutschland e.V.",
4                 "addInfo"  : null,
5                 "ID"       : 828,
6                 "Current"  : true,
7                 "Skills"   :["SQL","SSIS","SSRS",42,"MDX"],
8                 "Region"   :{"Country":"Germany","Territory":"Hessen"}
9             }';
10
```

100 %

Results | Messages

	key	value	type	Data Type
1	Name	PASS Deutschland e.V.	1	string
2	addInfo	NULL	0	null
3	ID	828	2	int
4	Current	true	3	true/false
5	Skills	["SQL","SSIS","SSRS",42,"MDX"]	4	array
6	Region	{"Country":"Germany","Territory":"Hessen"}	5	object

PASS SQL saturday

{JSON} Storing JSON

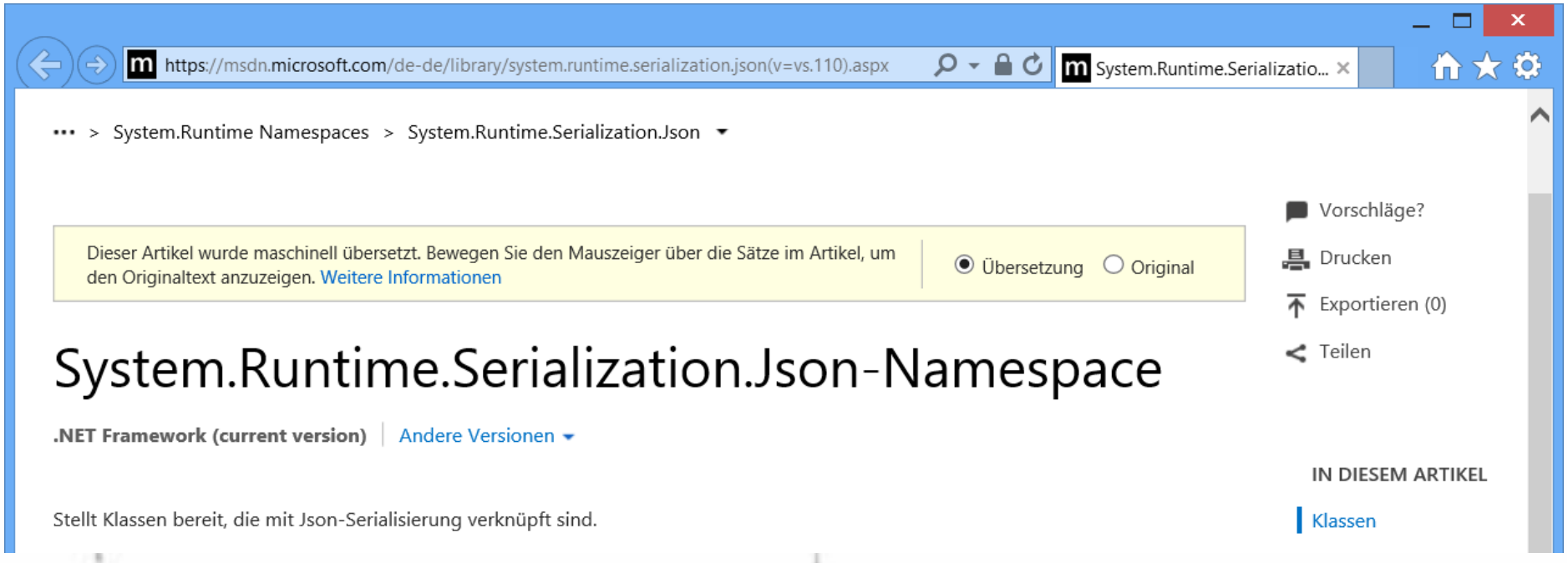
- `NVARCHAR(MAX)`

- Index
  „abgeleitete Spalte“
  mit `JSON_VALUE( @JsonCol, '$. … . …')`

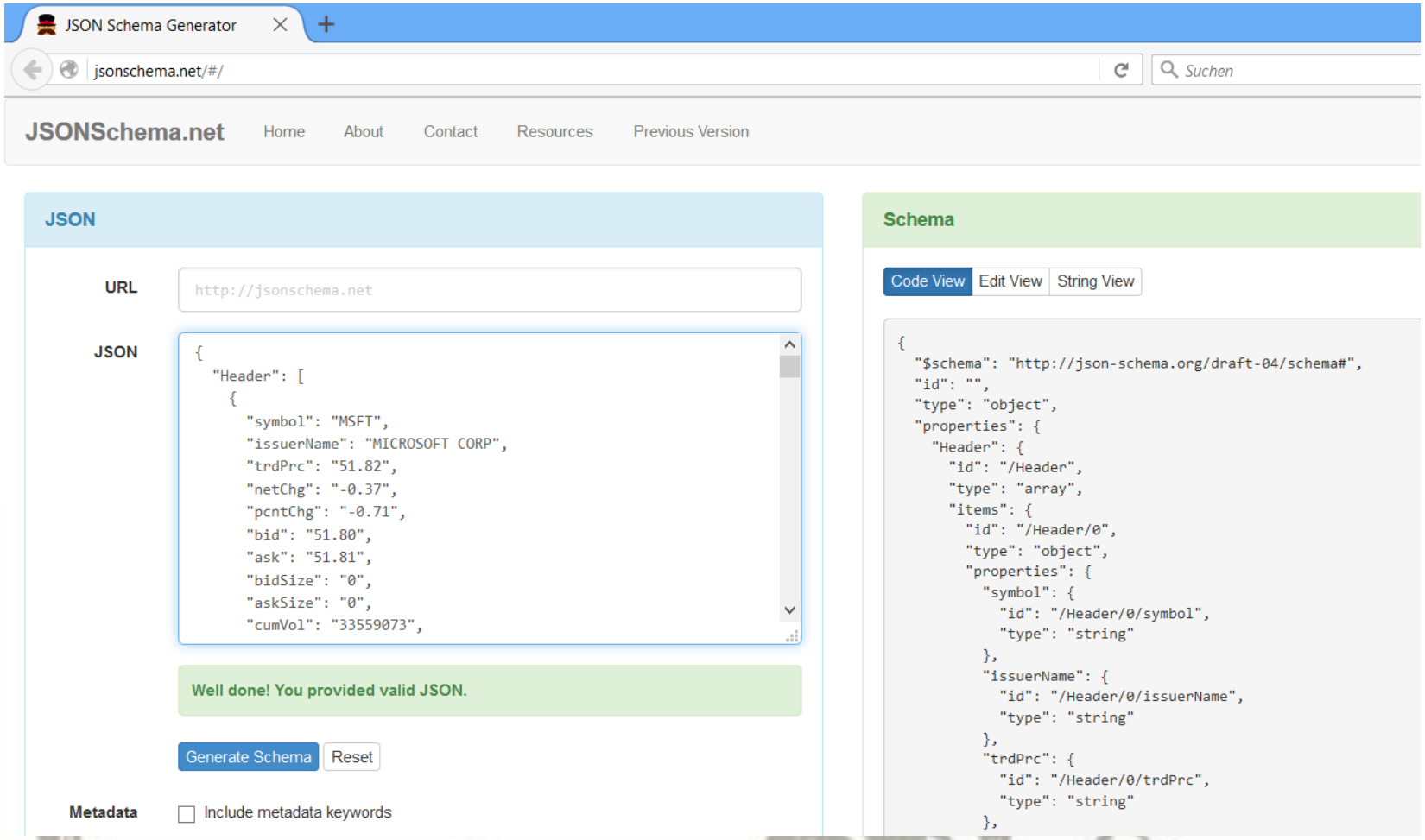
  `>> CREATE INDEX …`

PASS SQL saturday

JSON Namespaces



JSON schema validation



.. and Microsoft ♥ to store JSON data here

{JSON} JSON functions

- ISJSON()
  return 1 if JSON snippet is valid
- JSON_VALUE()
  extract a scalar value from JSON snippet
- JSON_QUERY()
  extract an object or an array from JSON snippet

PASS SQL saturday

<xml/> Select & Ins/Upd/Del

- T-SQL conformer Ins/Upd/Del
- T-SQL XQuery
  - <column>.query (return xml)
  - <column>.value (return sql-Type)
  - <column>.exist (return bit)
  - <column>.nodes

<xml/> Select & Ins/Upd/Del

- T-SQL Xquery Navigation im <xml>
  - child
  - descendant
  - parent
  - Attribut
  - self
  - descendant-or-self

  - .node()
  - .text()

<xml/> CRUD Samples

SQLQuery1.sql

```sql
----  1st Query
SELECT [key], xmlValue.query('*') as Complete_Sequence
FROM   dbo.CarManufacturer_xml

----  2nd Query
SELECT [key], xmlValue.query('data(*)') as Complete_Data
FROM   dbo.CarManufacturer_xml

----  3rd Query  /node()   &   /comment()
SELECT [key]
     , xmlValue.query('/Customer/CompanyName/node()') as CompanyName
     , xmlValue.query('/Customer/comment()')
FROM   dbo.CarManufacturer_xml

```

PASS SQL saturday

<xml/> Indexing

- PRIMARY XML INDEX IX_primaryXml

- CREATE XML INDEX IX_name
  ON sch.table (xmlCol)
  USING XML INDEX IX_primaryXml
  - FOR PATH;
  - FOR VALUE;
  - FOR PROPERTY;

PASS SQL saturday

<xml/> Indexing

xml_Indexes.sql

```sql
CREATE PRIMARY XML INDEX IX_primaryXml
ON  dbo.eConsumtion_xml  ( xmlValue );
GO


CREATE XML INDEX IX_Xml_PATH
ON  dbo.eConsumtion_xml  ( xmlValue )
USING XML INDEX  IX_primaryXml
FOR PATH;

CREATE XML INDEX IX_Xml_PROPERTY
ON  dbo.eConsumtion_xml  ( xmlValue )
USING XML INDEX  IX_primaryXml
FOR PROPERTY;

CREATE XML INDEX IX_Xml_VALUE
ON  dbo.eConsumtion_xml  ( xmlValue )
USING XML INDEX  IX_primaryXml
FOR VALUE;
GO
```

PASS SQL saturday

<xml/> Schema Validation

- XML SCHEMA COLLECTION schemaName

- (table) xmlColumn
  xml (DOCUMENT schemaName)

<xml/> Schema Validation

xml_Schema.sql

```
1  CREATE XML SCHEMA COLLECTION  dbo.bookSchemaCollection
2  AS
3  N'<?xml version="1.0"?>
4  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
5      targetNamespace="http://www.test-fabrik.de/BooksSchema"
6      xmlns="http://www.test-fabrik.de/BooksSchema"
7      elementFormDefault="qualified">
8      <xs:element name="Book">
9          <xs:complexType>
10             <xs:attribute name="Title" type="xs:string"/>
11             <xs:attribute name="Price" type="xs:decimal"/>
12         </xs:complexType>
13     </xs:element>
14 </xs:schema>';
15 GO
16
17 CREATE TABLE  dbo.Books_xml(
18                ID        int Identity(1,1)
19              , [key]     nvarchar(50)
20              , xmlRaw    xml
21              , xmlValue  xml(DOCUMENT bookSchemaCollection)
22              );
23 GO
24
```

<xml/> Schema Validation

xml_Schema.sql

```
1  INSERT dbo.Books_xml
2          ( [key] , xmlValue )
3  VALUES ( 'key02', '<Book xmlns="http://www.test-fabrik.de/BooksSchema"
4                           Title="the booksample2"
5                           Price="49.90" />')
6
7  --------------
8  -- !! decimal ,
9  INSERT dbo.Books_xml
10         ( [key] , xmlValue )
11 VALUES ( 'key03', '<Book xmlns="http://www.test-fabrik.de/BooksSchema"
12                          Title="! v ! Book"
13                          Price="49,90" />')
14
15 --Msg 6926, Level 16, State 1, Line 40
16 --XML Validation: Invalid simple type value: '49,90'. Location: /*:Book[1]/@*:Price
```

PASS SQL saturday

<xml/> queries & transformation

- XML *„FLWOR“*
  for **l**et **w**here **o**rder **r**eturn

- Aggregat functions
  - ✓ {count($j)}
  - ✓ {sum($j)}
  - ✓ {avg($j)}

PASS SQL saturday

<xml/> queries & transformation

xml_FLWOR.sql

```sql
-- sample FLWOR
SELECT ID, [key], xmlValue
     , xmlValue.value('(CustomerData/Customer/CustID)[1]', 'nvarchar(40)' )  as 'nvarchar_CustID'
     , xmlValue.query(' for $i in CustomerData/Customer/CustID
                        let $j := CustomerData/Customer/Consumtion/consumtionValue
                        where  avg($j) >= 5
                        return
                        <result>
                           {data($j)}
                        </result>
                      ') as 'data'
     , xmlValue.query(' for $i in CustomerData/Customer/CustID
                        let $j := CustomerData/Customer/Consumtion/consumtionValue
                        where  avg($j) >= 5
                        return
                        <result>
                           <count>  {count($j)} </count>
                           <sum>    {sum($j)}   </sum>
                           <avg>    {avg($j)}   </avg>
                        </result>
                      ') as 'summary'
FROM   dbo.eConsumtion_xml;
```

PASS SQL saturday

???

PASS SQL saturday

Please review the event and sessions

- EVENT

http://speakerscore.com/ZGVX

- SESSION

http://speakerscore.com/FQ1W