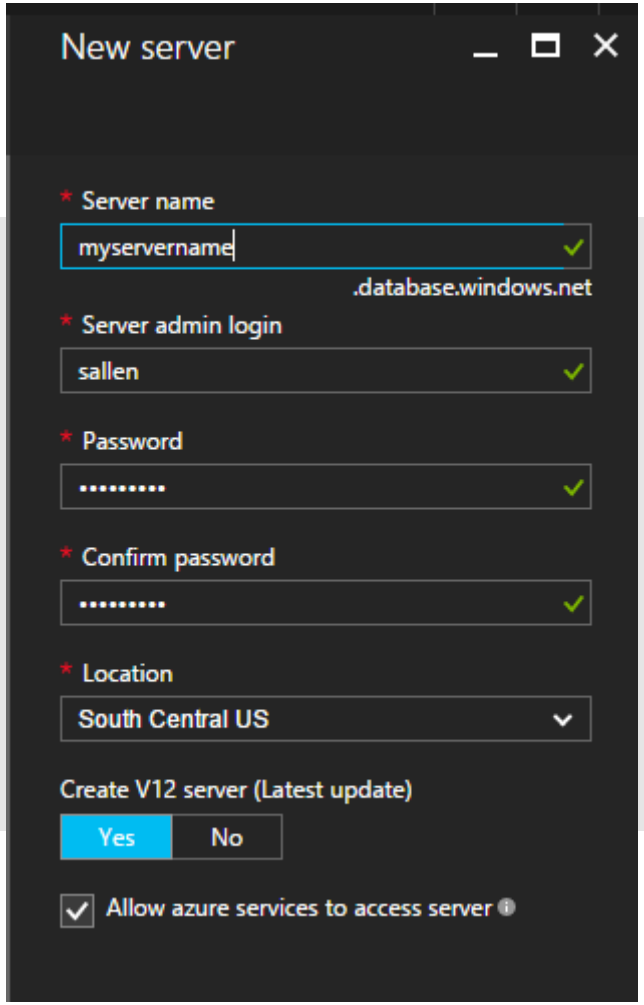
Using Cloud Databases

Scott Allen

@OdeToCode – http://OdeToCode.com

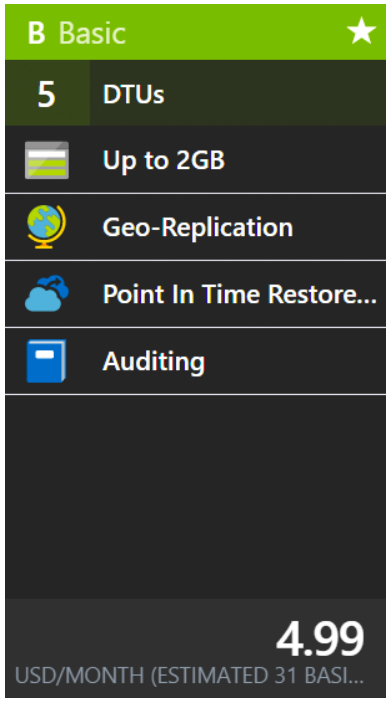
Azure SQL
Database

DocumentDB
Database

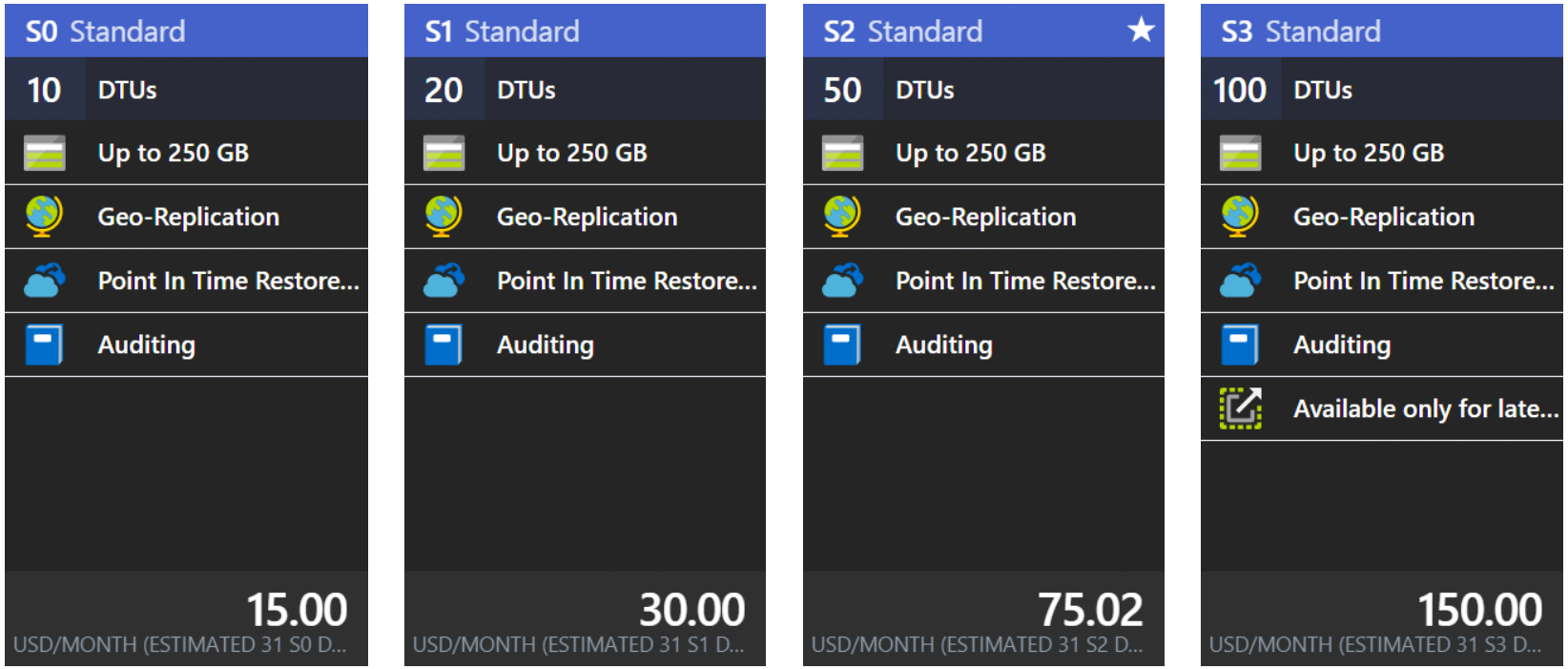

New server
* Server name
myservername
.database.windows.net
* Server admin login
sallen
* Password
* Confirm password
* Location
South Central US
Create V12 server (Latest update)
Yes
No
Allow azure services to access server

Server

SQL
SQL
SQL


B Basic
5
DTUs
Up to 2GB
Geo-Replication
Point In Time Restore...
Auditing
4.99
USD/MONTH (ESTIMATED 31 BASI...

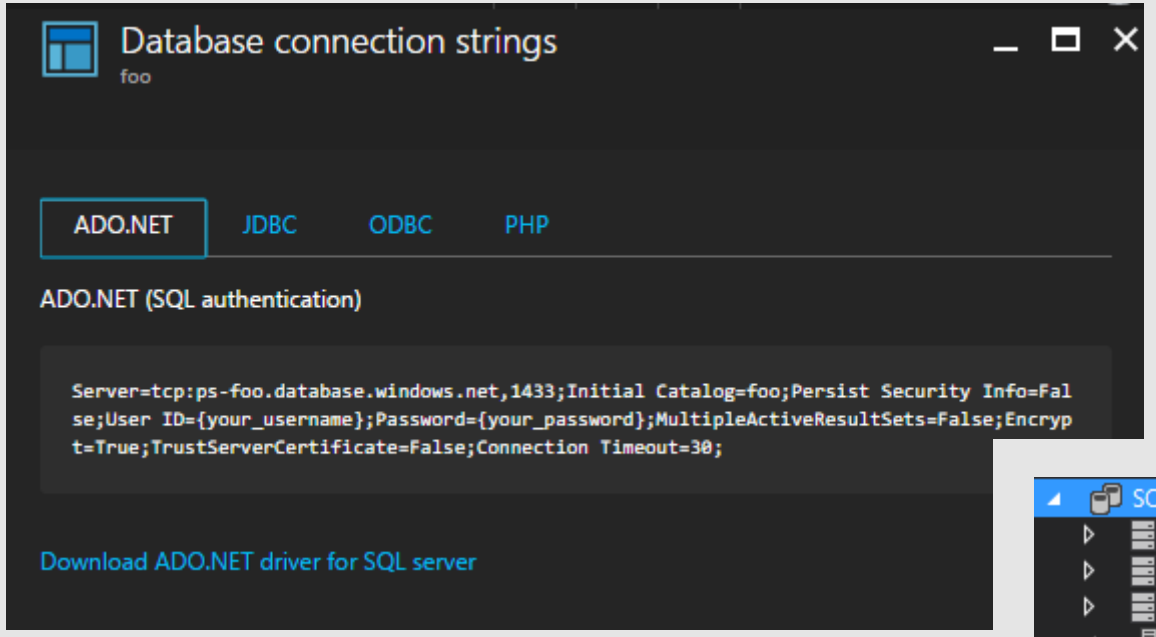
5

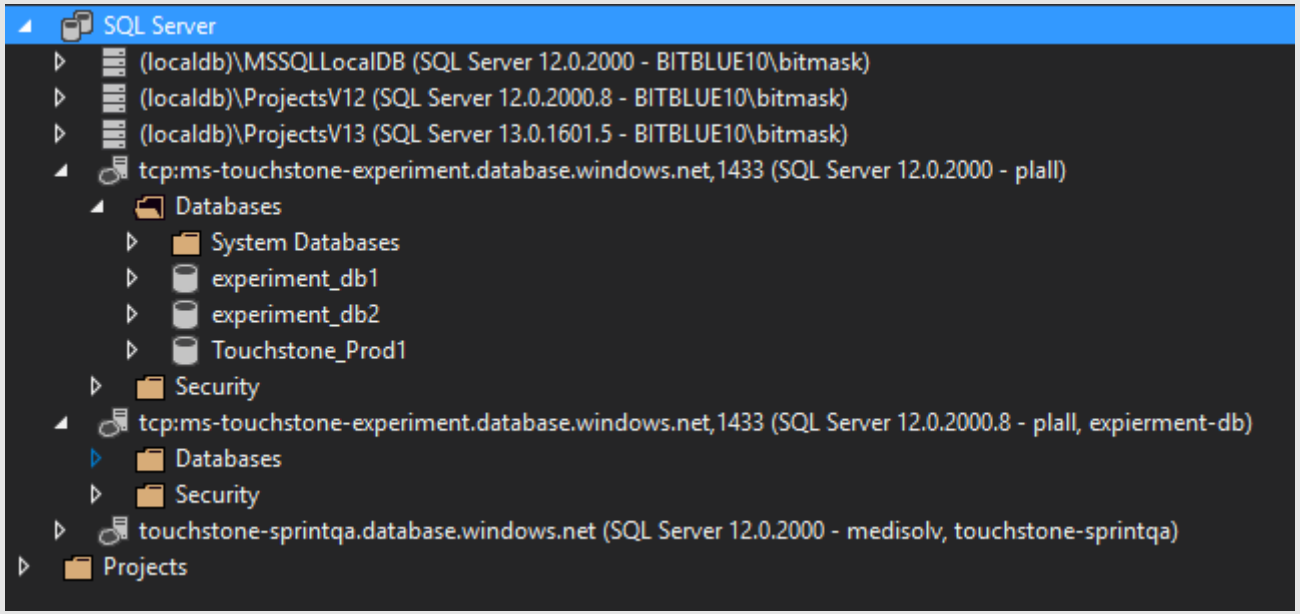

S0 Standard
10 DTUs
Up to 250 GB
Geo-Replication
Point In Time Restore...
Auditing
15.00
USD/MONTH (ESTIMATED 31 S0 D...
S1 Standard
20 DTUs
Up to 250 GB
Geo-Replication
Point In Time Restore...
Auditing
30.00
USD/MONTH (ESTIMATED 31 S1 D...
S2 Standard
50 DTUs
Up to 250 GB
Geo-Replication
Point In Time Restore...
Auditing
75.02
USD/MONTH (ESTIMATED 31 S2 D...
S3 Standard
100 DTUs
Up to 250 GB
Geo-Replication
Point In Time Restore...
Auditing
Available only for late...
150.00
USD/MONTH (ESTIMATED 31 S3 D...

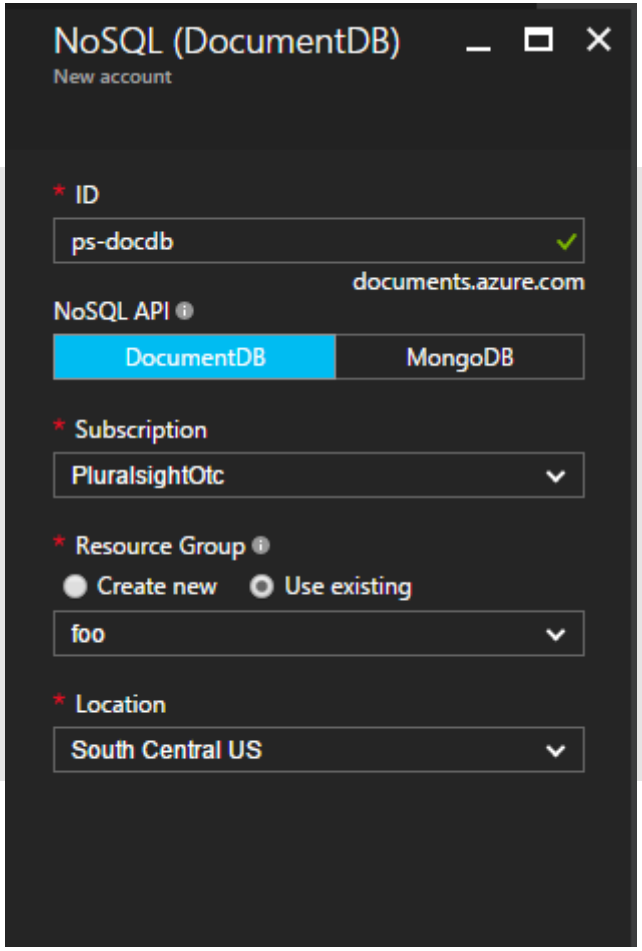
10-100

P1 Premium ★	P2 Premium	P4 Premium	P6 Premium	P11 Premium	P15 Premium
125 DTUs	250 DTUs	500 DTUs	1000 DTUs	1750 DTUs	4000 DTUs
Up to 500 GB	Up to 500 GB	Up to 500 GB	Up to 500 GB	Up to 1024 GB	Up to 1024 GB
Geo-Replication	Geo-Replication	Geo-Replication	Geo-Replication	Geo-Replication	Geo-Replication
Point In Time Restore...	Point In Time Restore...	Point In Time Restore...	Point In Time Restore...	Point In Time Restore...	Point In Time Restore...
Auditing	Auditing	Auditing	Auditing	Auditing	Auditing
465.00 USD/MONTH (ESTIMATED 31 P1 D...	930.00 USD/MONTH (ESTIMATED 31 P2 D...	1,860.00 USD/MONTH (ESTIMATED 31 P4 D...	3,720.00 USD/MONTH (ESTIMATED 31 P6 D...	7,000.11 USD/MONTH (ESTIMATED 31 P11 ...	15,999.97 USD/MONTH (ESTIMATED 31 P15 ...

125 - 4000


Database connection strings
foo
ADO.NET
JDBC
ODBC
PHP
ADO.NET (SQL authentication)
Server=tcp:ps-foo.database.windows.net,1433;Initial Catalog=foo;Persist Security Info=False;User ID={your_username};Password={your_password};MultipleActiveResultSets=False;Encrypt=True;TrustServerCertificate=False;Connection Timeout=30;
Download ADO.NET driver for SQL server


SQL Server
(localdb)\MSSQLLocalDB (SQL Server 12.0.2000 - BITBLUE10\bitmask)
(localdb)\ProjectsV12 (SQL Server 12.0.2000.8 - BITBLUE10\bitmask)
(localdb)\ProjectsV13 (SQL Server 13.0.1601.5 - BITBLUE10\bitmask)
tcp:ms-touchstone-experiment.database.windows.net,1433 (SQL Server 12.0.2000 - plall)
Databases
System Databases
experiment_db1
experiment_db2
Touchstone_Prod1
Security
tcp:ms-touchstone-experiment.database.windows.net,1433 (SQL Server 12.0.2000.8 - plall, expierment-db)
Databases
Security
touchstone-sprintqa.database.windows.net (SQL Server 12.0.2000 - medisolv, touchstone-sprintqa)
Projects

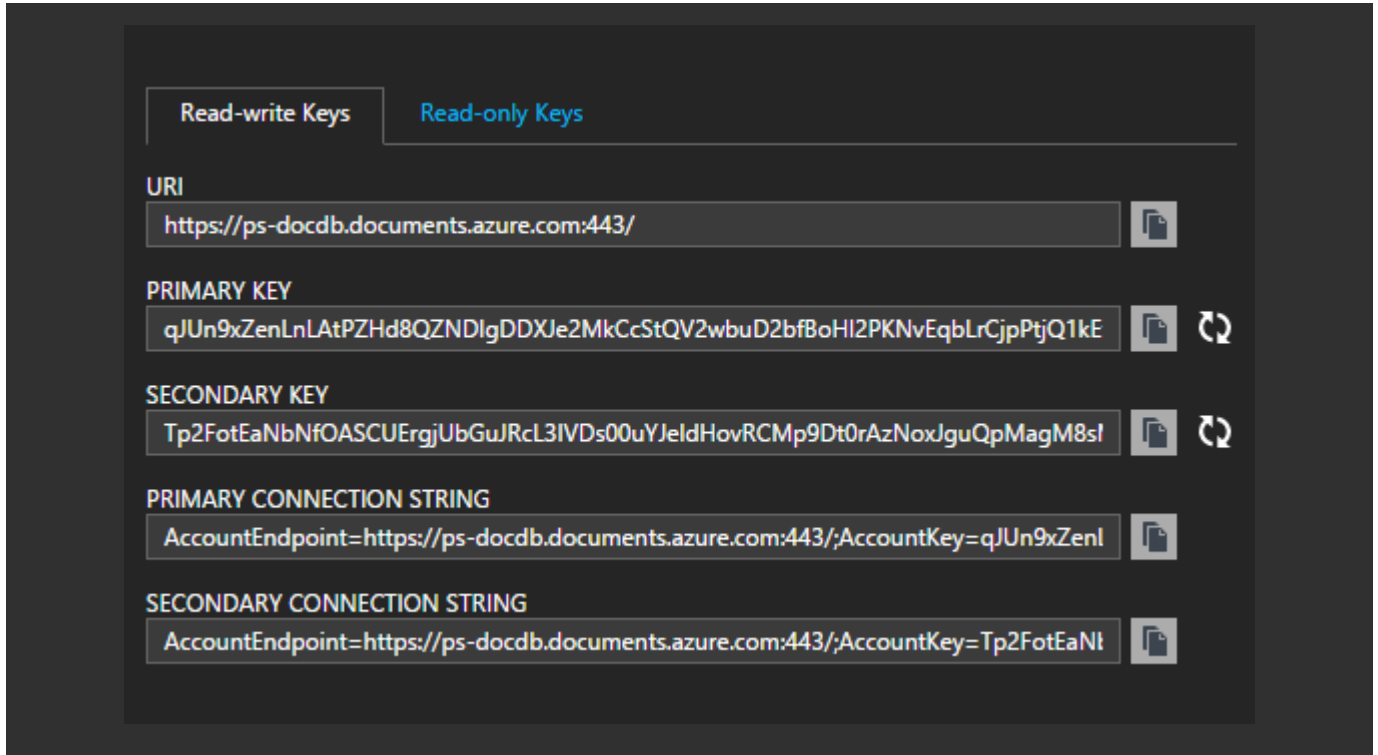

NoSQL (DocumentDB)
New account
* ID
ps-docdb
documents.azure.com
NoSQL API
DocumentDB
MongoDB
* Subscription
PluralsightOtc
* Resource Group
Create new
Use existing
foo
* Location
South Central US

Microsoft Azure DocumentDB Client... 1.11.1

This client library enables client applications to connect to the Azure DocumentDB service. Azure DocumentDB is a NoSQL document database hosted in Microsoft Azure and delivered as a service. For more information, refer to http://azure.microsoft.com/services/documentdb/.

To install Microsoft Azure DocumentDB Client Library, run the following command in the Package Manager Console

```
PM> Install-Package Microsoft.Azure.DocumentDB
```



```
public static T Get(Expression<Func<T, bool>> predicate)
{
    return Client.CreateDocumentQuery<T>(Collection.DocumentsLink)
                 .Where(predicate)
                 .AsEnumerable()
                 .FirstOrDefault();
}

public static IEnumerable<T> Find(Expression<Func<T, bool>> predicate)
{
    var ret = Client.CreateDocumentQuery<T>(Collection.SelfLink)
        .Where(predicate)
        .AsEnumerable();

    return ret;
}
```

```
public IEnumerable<Course> GetAllCourses()
{
    return _client.CreateDocumentQuery<Course>(_coursesLink)
                  .OrderBy(c => c.Title);
}

public async Task InsertCourses(IEnumerable<Course> courses)
{
    foreach (var course in courses)
    {
        await _client.CreateDocumentAsync(_coursesLink, course);
    }
}
```