Cloud Computing

Microsoft Azure

Agenda

- Brief Recap
- Azure Deployment Models – ASM and ARM
- Hands-on Activities on Azure cloud
  - Create and access a virtual machine
  - Deploying an application to Azure cloud and app service
  - Working with Azure SQL database
  - Working with Redis Cache on Azure
  - Content Delivery Network
- Azure Calculator

Recap

- Why Cloud computing
- Cloud Deployment Models
- Cloud Service Models
- Architecture and Design Concepts
  - High Availability
  - Disaster Recovery
  - Multi Tenancy

Azure infrastructure provisioning models or deployment models

- Azure Service Management or Classic (ASM)
- Azure Resource Manager (ARM)

Azure Deployment models

ASM

- It is old model
- Works on old portal with backward compatibility on new portal
- Primarily originated from PaaS mindset, e.g. Cloud Service based approach
- Does not work very well on customized scenarios and non .NET environments
- Deployments can be done by portal, SDKs and REST APIs
- Not recommended from Microsoft for any future development

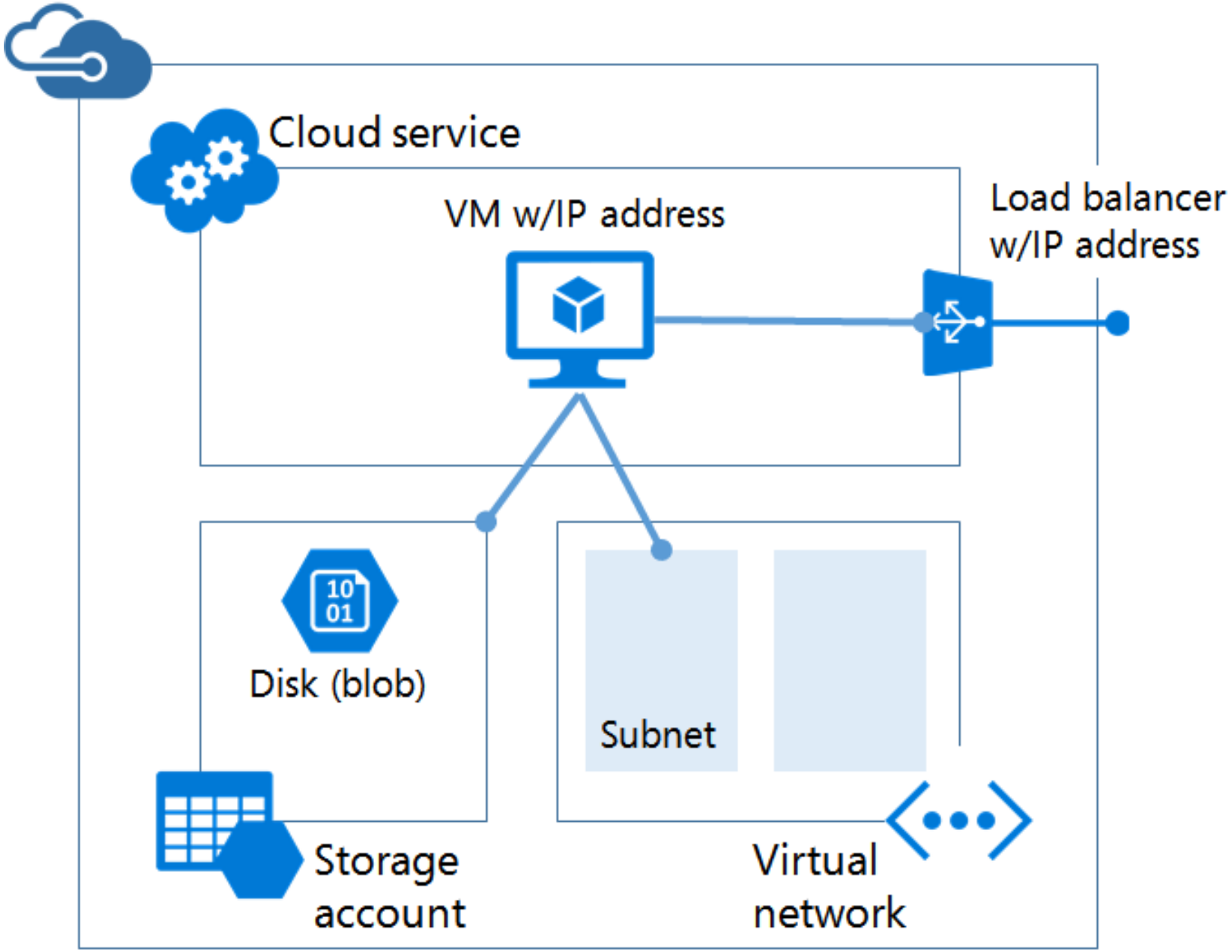
ARM

- It is a new model
- Works only on new portal
- Redesign to allow IaaS based mindset
  - compute, storage, networking all as distinguished components
- Works in all scenarios regardless of technologies
- Deployments can be done via
  - new portal, SDKs, REST APIs and JSON templates
- Recommended for all future development

ASM or Classic

- Cloud service that acts as a container for hosting virtual machines
  - Virtual machines are automatically provided with a network interface card (NIC) and an IP address assigned by Azure.
  - The cloud service contains an external load balancer instance, a public IP address, and default endpoints to allow remote desktop and remote PowerShell and Secure Shell (SSH) traffic.
- A storage account that stores the VHDs for a virtual machine
  - It includes the operating system, temporary, and additional data disks.
- An optional virtual network that acts as an additional container, in which you can create a sub-netted structure and designate the subnet on which the virtual machine is located (network).

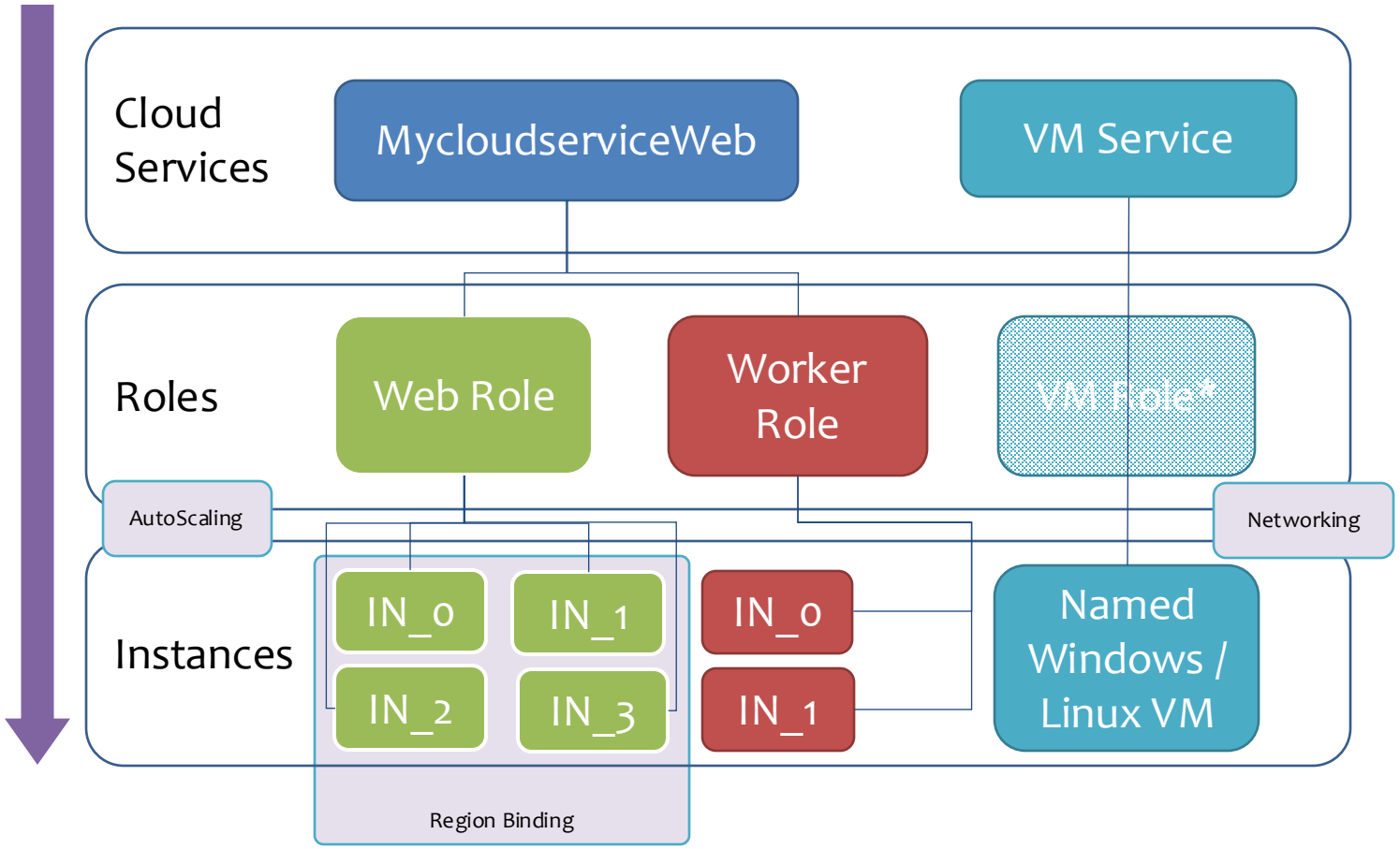
ASM


Cloud service
VM w/IP address
Load balancer
w/IP address
10
01
Disk (blob)
Subnet
Storage
account
Virtual
network

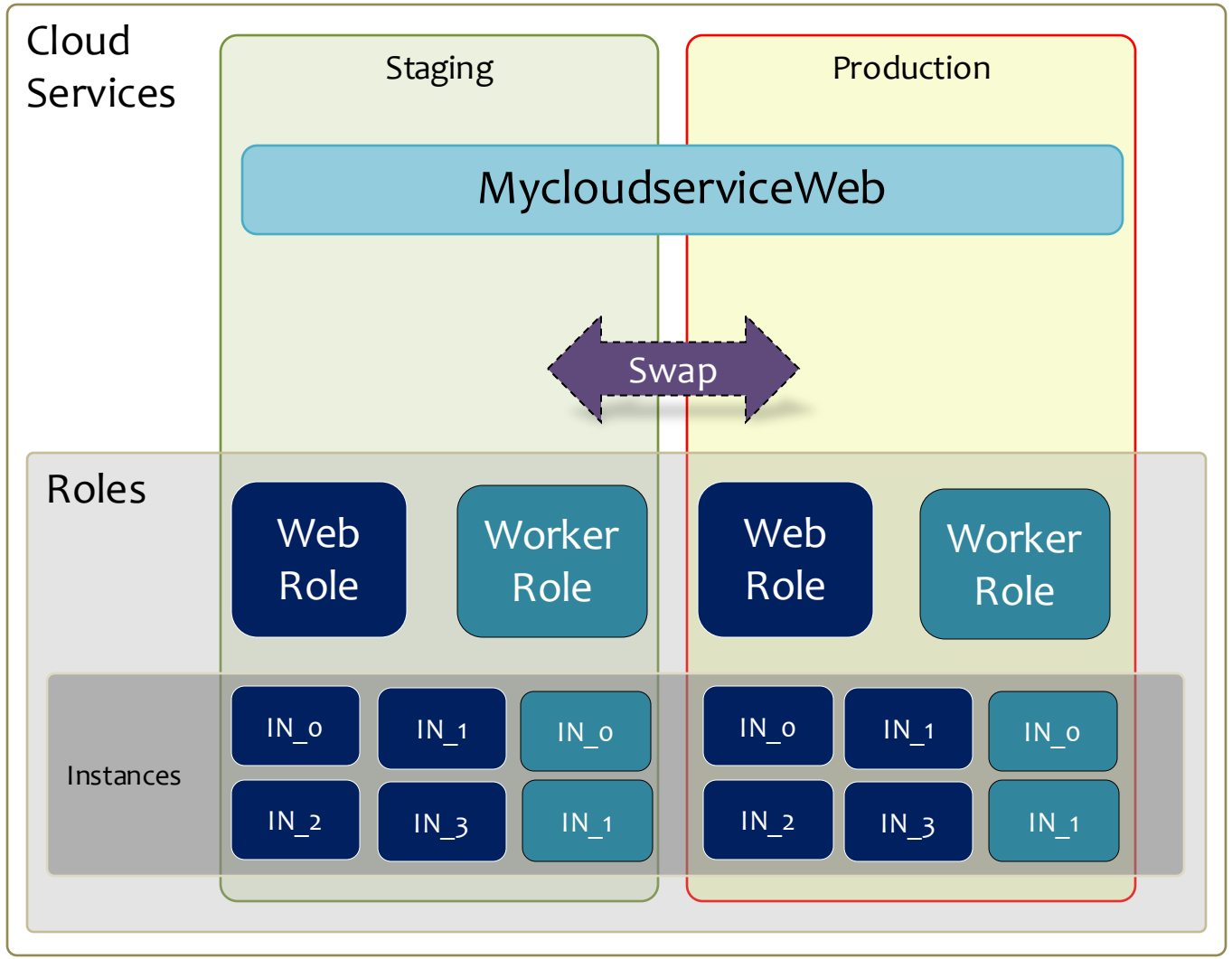
Cloud Service Topology: Roles vs Instances

- Role is a logical boundary of virtual machine.
- A role can have multiple virtual machines in it.
- All VMs created must have a role associated.
- A role provides following functionalities as OOB feature:
  - AutoScaling
  - Inter VM networking
  - Region binding
- There are 3 roles
  - Web role
  - Worker role
  - VM role (now deprecated)

Cloud Service Topology



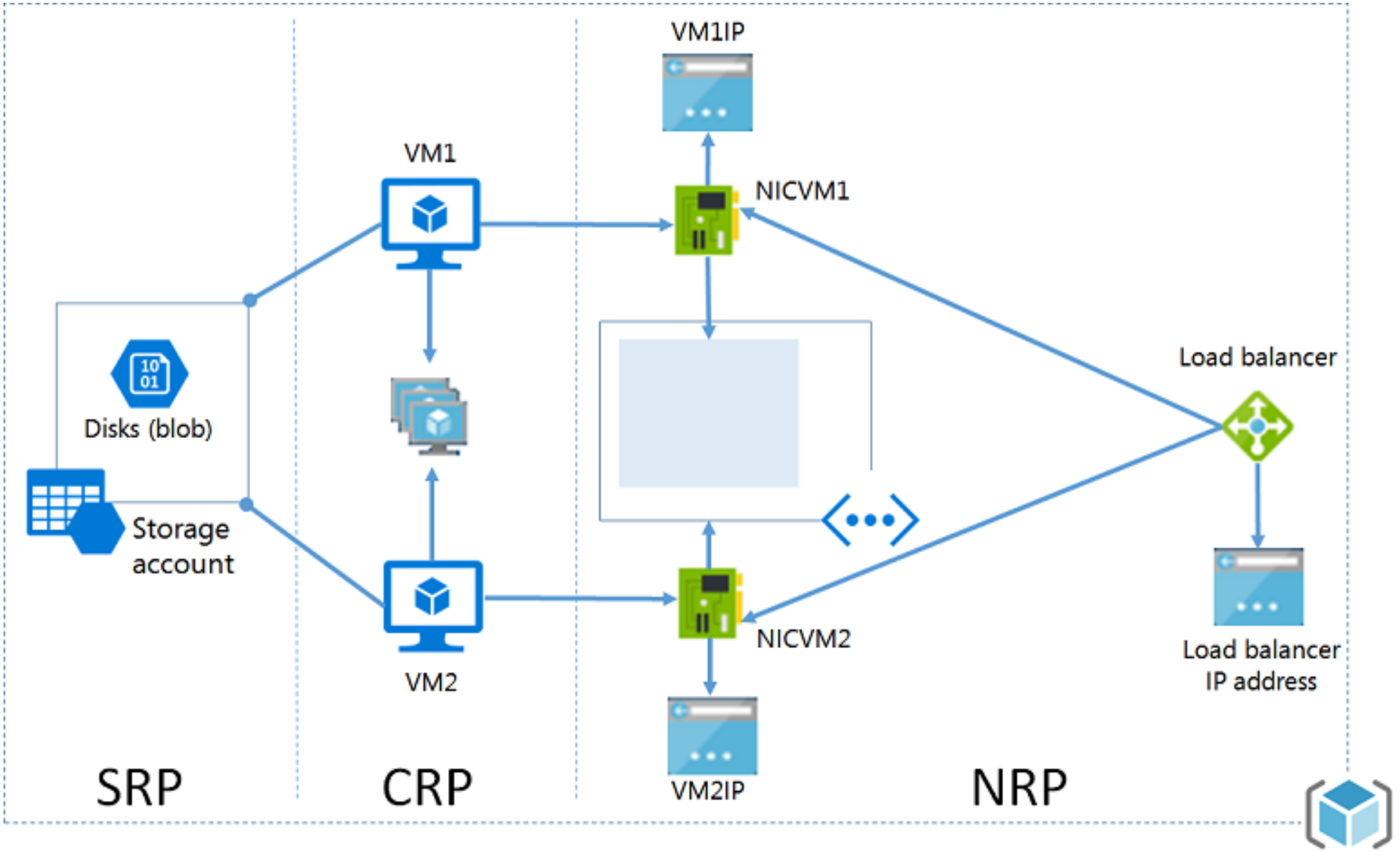
Cloud Service Topology



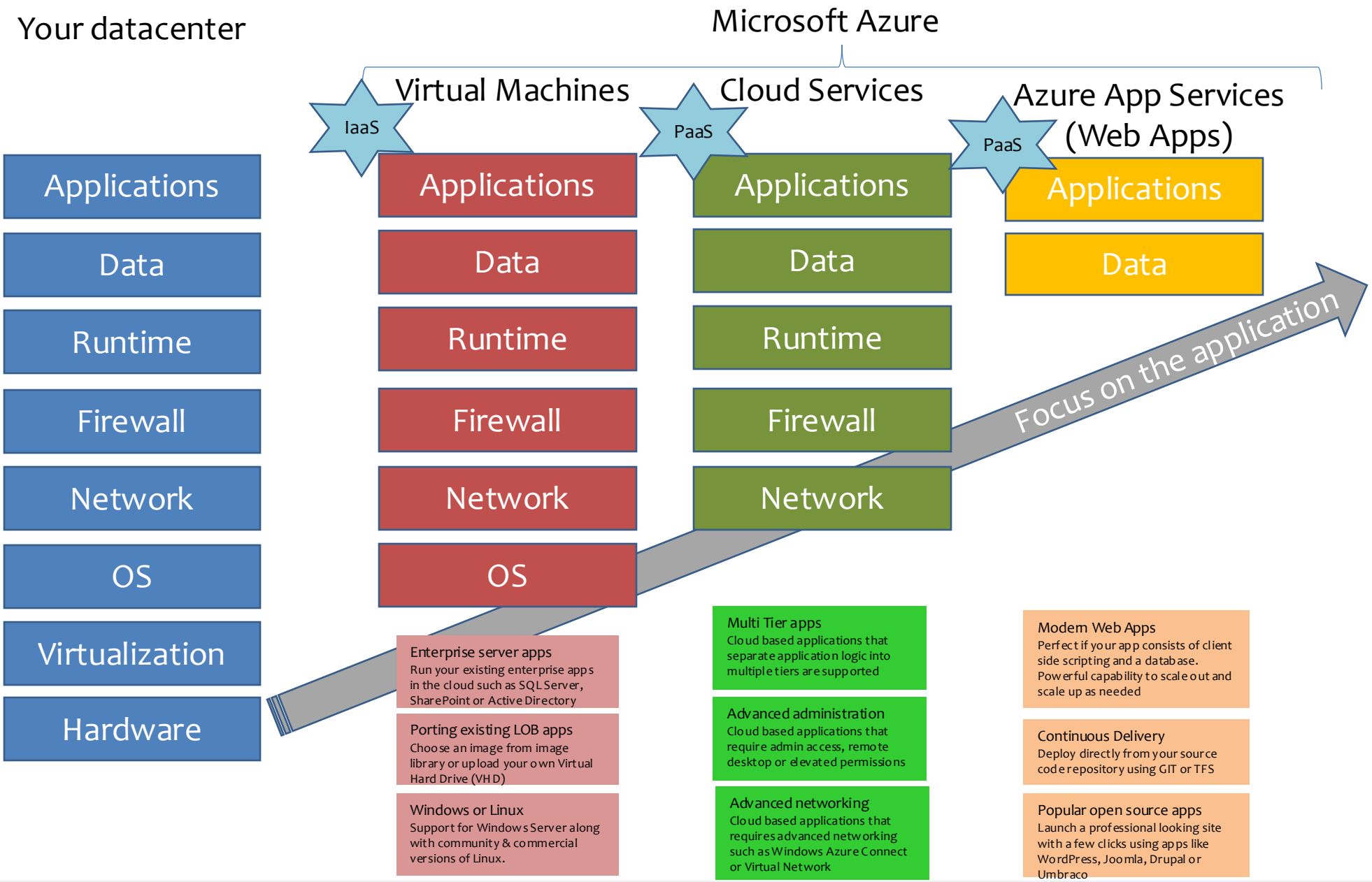
ARM

- **Compute** - Supports instances of virtual machines and optional availability sets.
- **Storage** - Supports required storage accounts that store the VHDs for virtual machines, including their operating system and additional data disks.
- **Network** - Supports required NICs, virtual machine IP addresses, and subnets within virtual networks and optional load balancers, load balancer IP addresses, and Network Security Groups.

ARM


VM1IP
VM1
NICVM1
Load balancer
Disks (blob)
Storage account
NICVM2
Load balancer IP address
VM2
VM2IP
SRP
CRP
NRP

Azure VM vs Cloud Services vs App Services




KEEP
CALM
AND
HAVE
A TEA BREAK

Cloud Computing

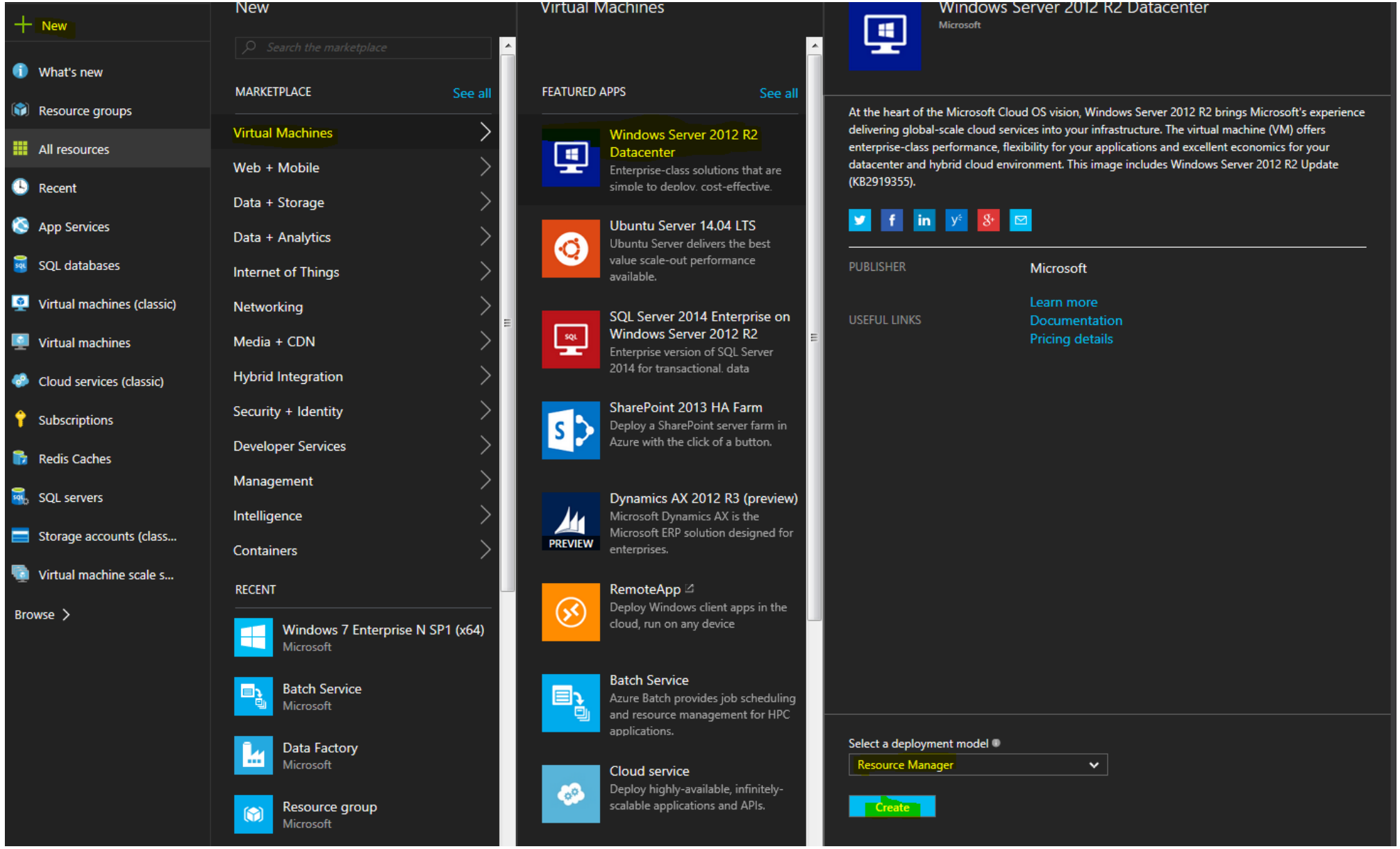
Part – II

Demo and Hands on Activities

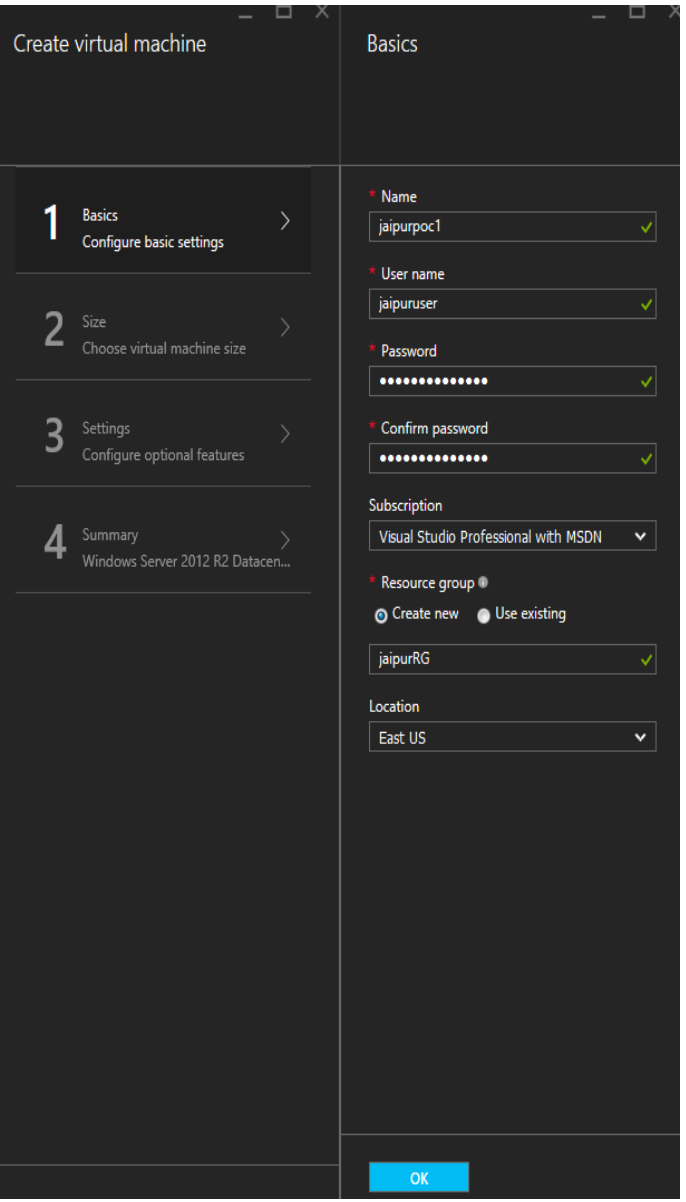
Lets create a

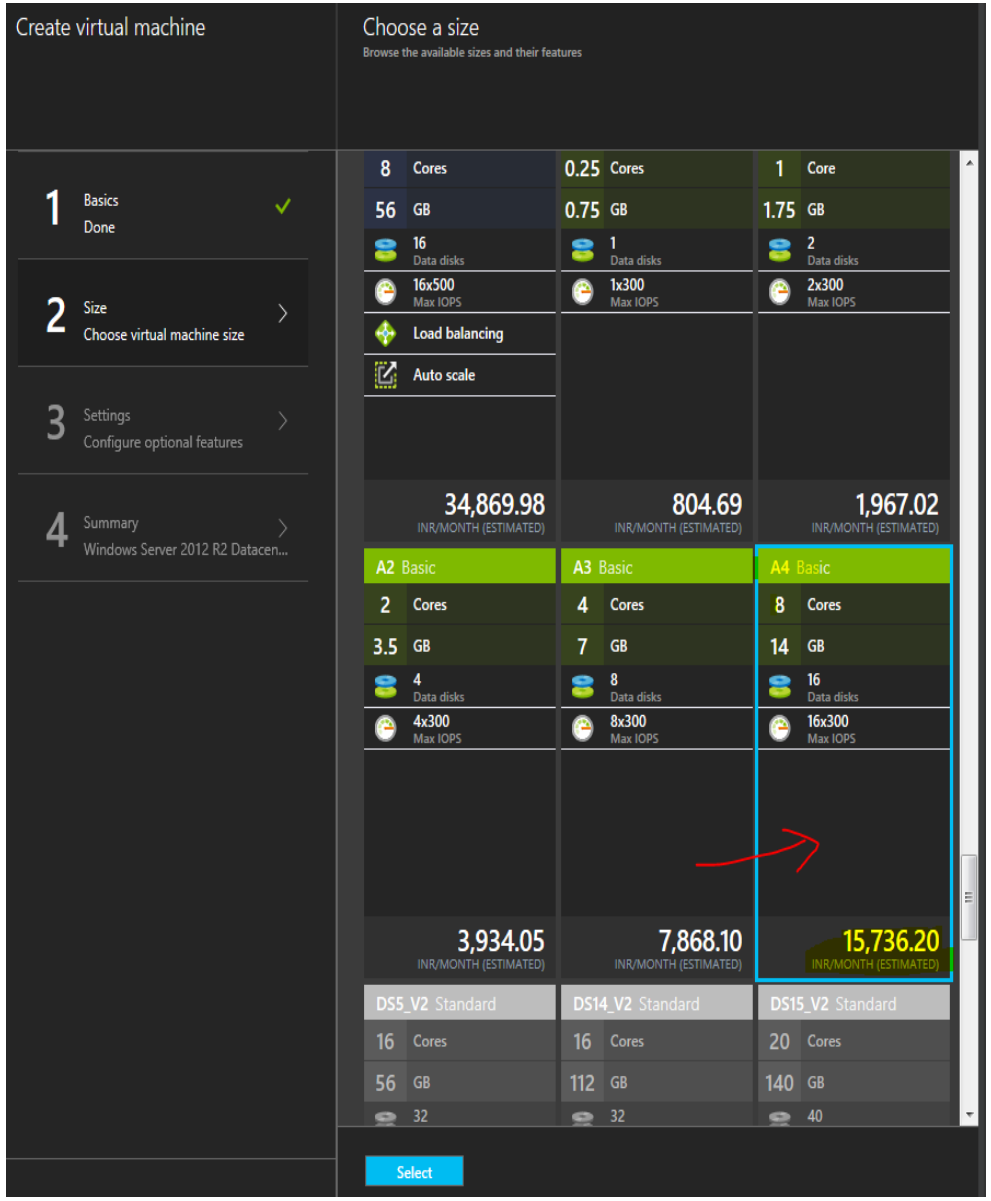
Virtual Machine

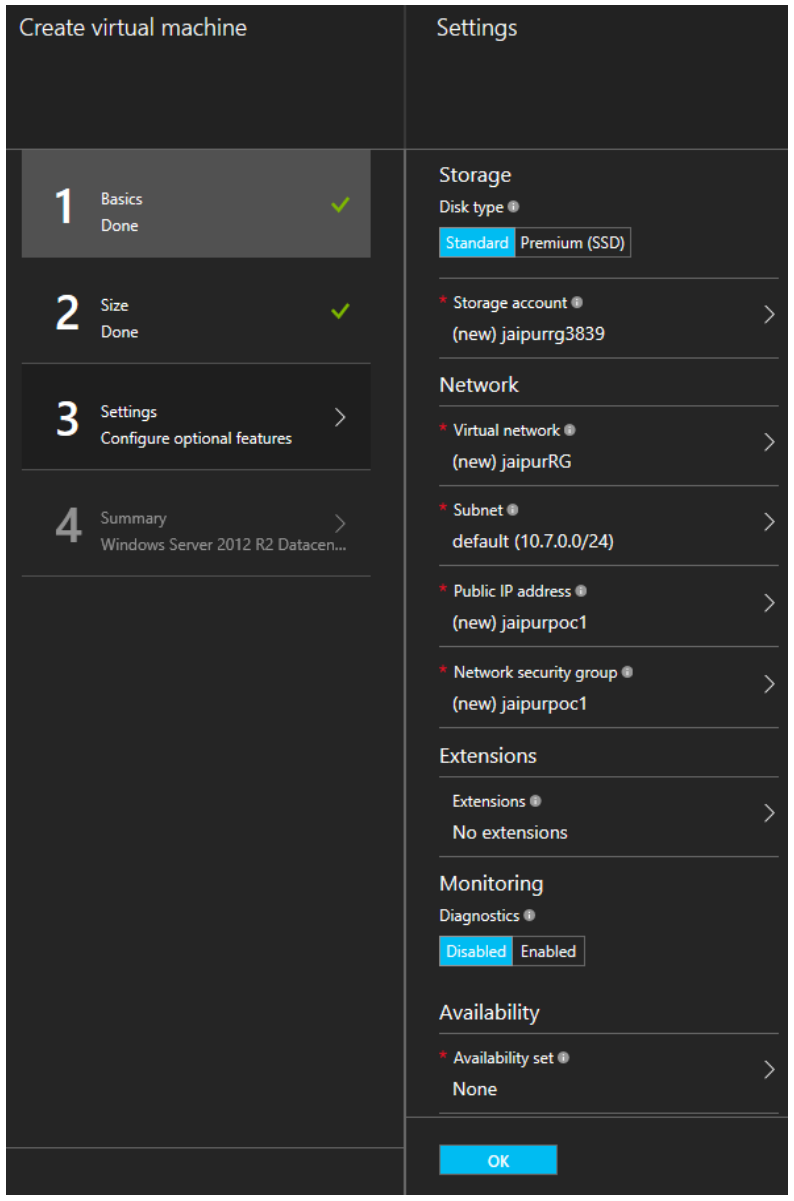
Azure Virtual Machine



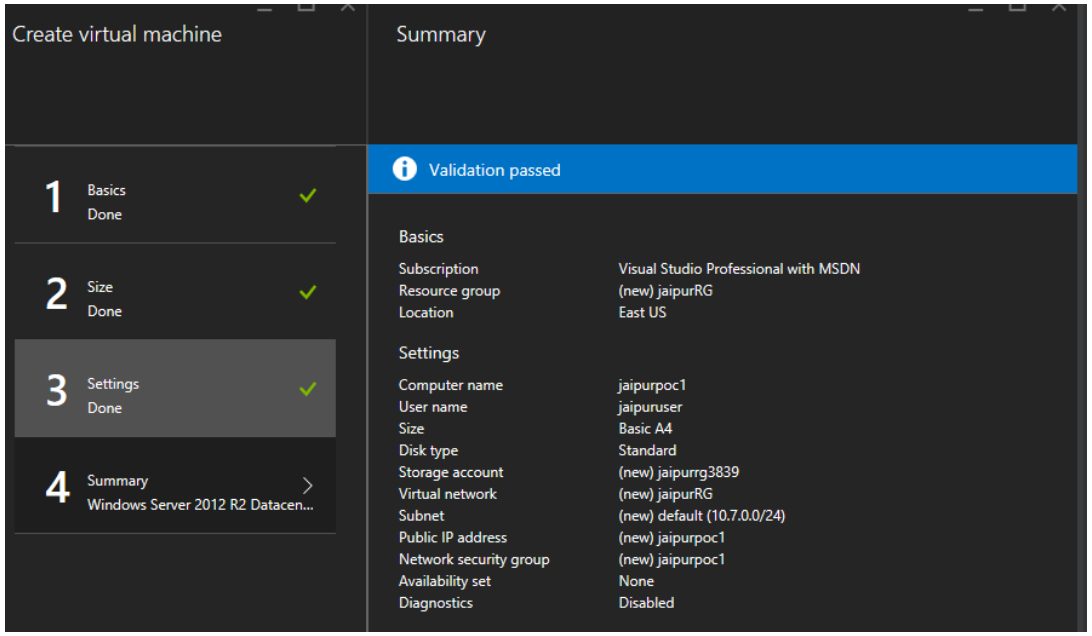
Azure Virtual Machine

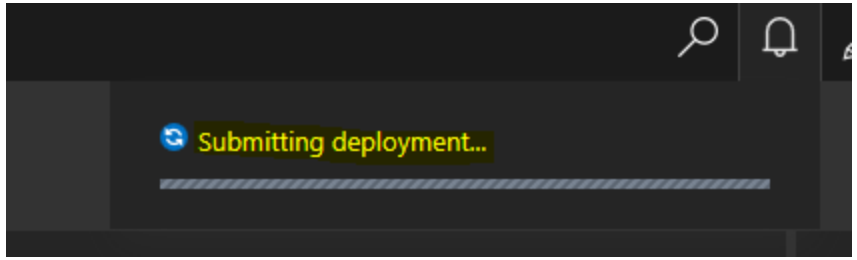


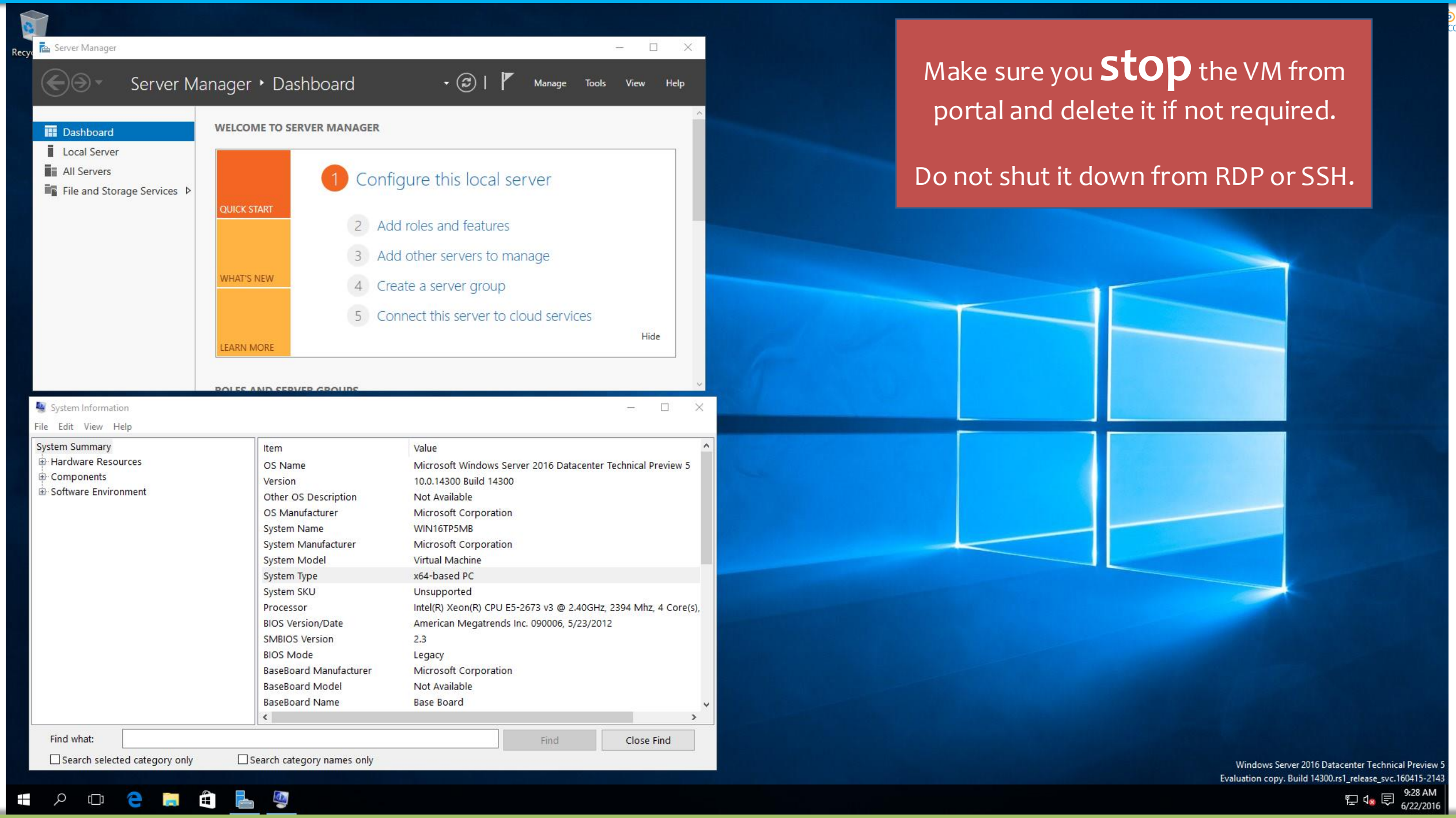




Azure Virtual Machine






Server Manager
Server Manager ▸ Dashboard
Manage
Tools
View
Help
Dashboard
Local Server
All Servers
File and Storage Services
WELCOME TO SERVER MANAGER
QUICK START
WHAT'S NEW
LEARN MORE
1 Configure this local server
2 Add roles and features
3 Add other servers to manage
4 Create a server group
5 Connect this server to cloud services
Hide
ROLES AND SERVER GROUPS
System Information
File Edit View Help
System Summary
Hardware Resources
Components
Software Environment
Item
Value
OS Name
Microsoft Windows Server 2016 Datacenter Technical Preview 5
Version
10.0.14300 Build 14300
Other OS Description
Not Available
OS Manufacturer
Microsoft Corporation
System Name
WIN16TP5MB
System Manufacturer
Microsoft Corporation
System Model
Virtual Machine
System Type
x64-based PC
System SKU
Unsupported
Processor
Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz, 2394 Mhz, 4 Core(s),
BIOS Version/Date
American Megatrends Inc. 090006, 5/23/2012
SMBIOS Version
2.3
BIOS Mode
Legacy
BaseBoard Manufacturer
Microsoft Corporation
BaseBoard Model
Not Available
BaseBoard Name
Base Board
Find what:
Find
Close Find
Search selected category only
Search category names only
Make sure you stop the VM from portal and delete it if not required.
Do not shut it down from RDP or SSH.
Windows Server 2016 Datacenter Technical Preview 5
Evaluation copy. Build 14300.rs1_release_svc.160415-2143
9:28 AM
6/22/2016

Now.. Lets get into some coding

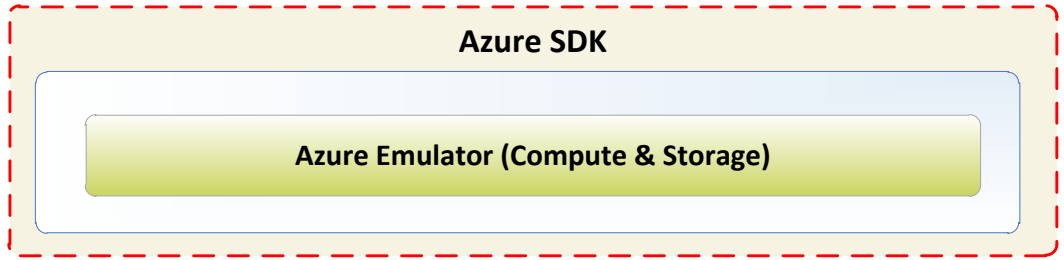
Deploy an application on Azure cloud service

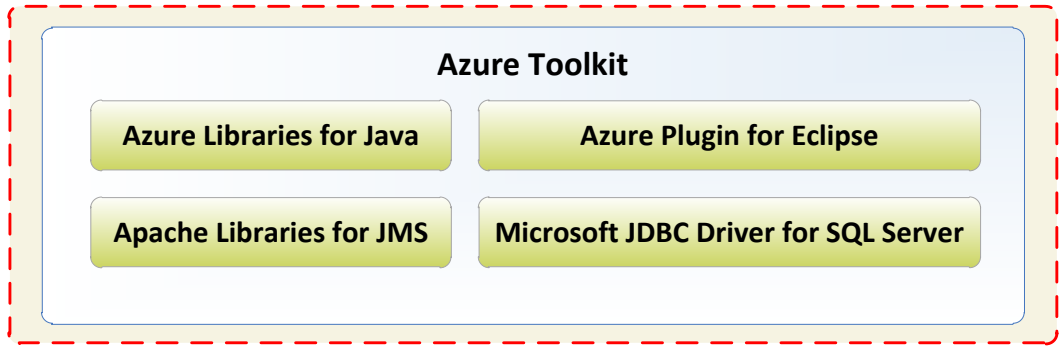
Azure environment setup for Java

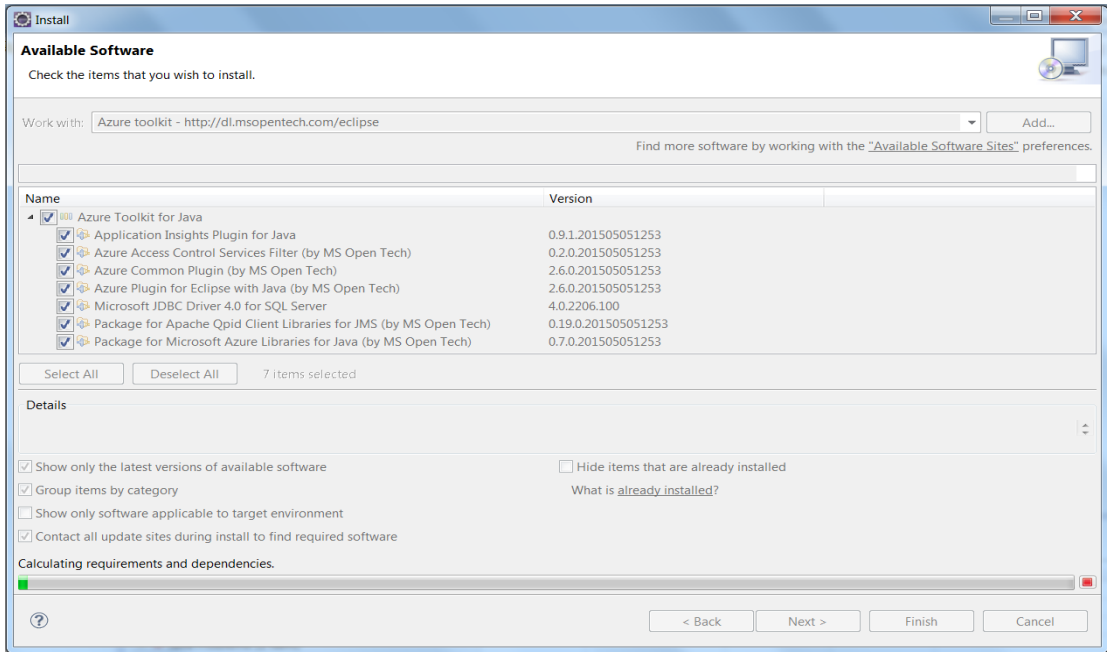
First install Azure SDK by using Web Platform Installer (WPI) http://go.microsoft.com/fwlink/?LinkID=252838

Then install Azure Toolkit for Eclipse https://msdn.microsoft.com/en-us/library/azure/hh690946.aspx

Azure Plugin contains Windows Azure project creation wizard & utility scripts.







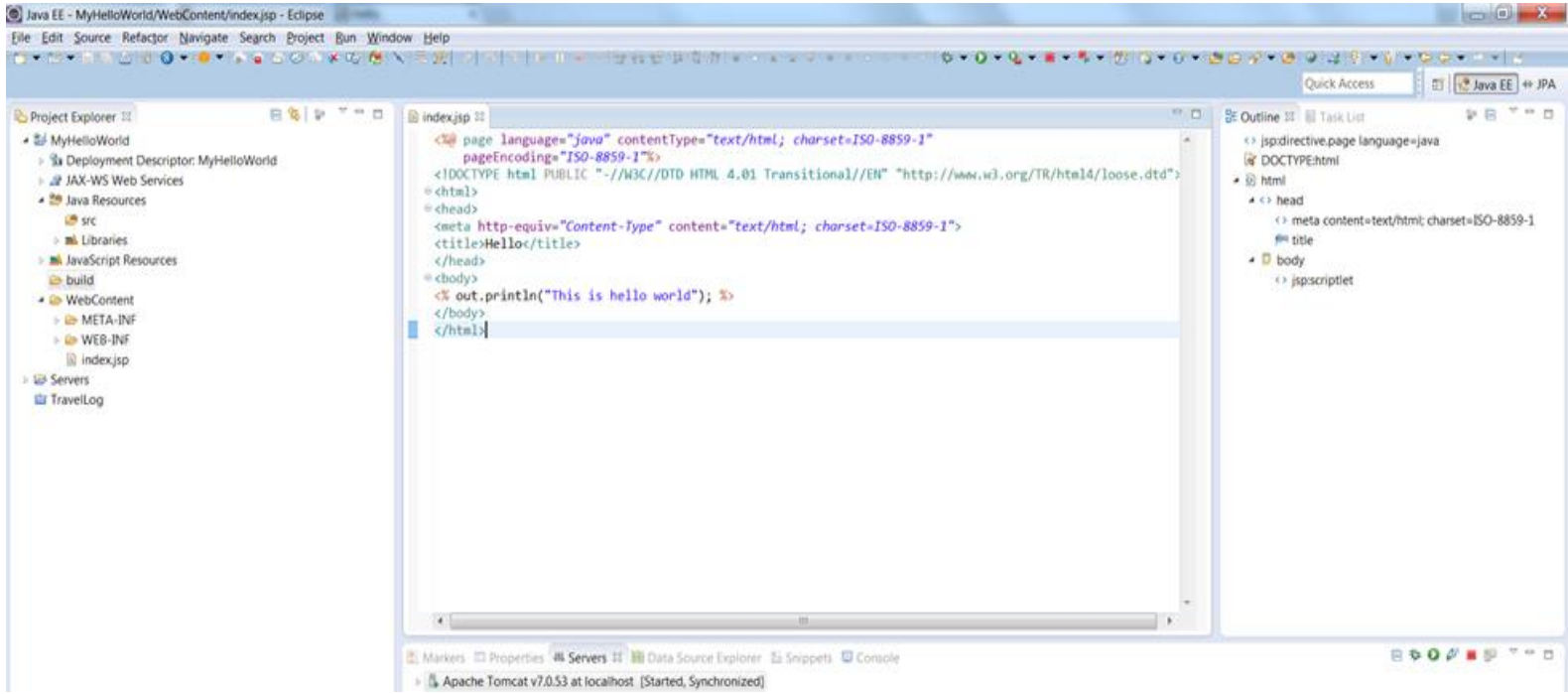
Java Hands-on | Task 1

Deploy Java App on Azure Cloud Service (as Worker Role)

Deploy Java App as Azure Cloud Service

Create a Dynamic Web Project (named MyHelloWorld) and add a JSP file (index.jsp)

Update the JSP file with sample code to print “This is hello world”



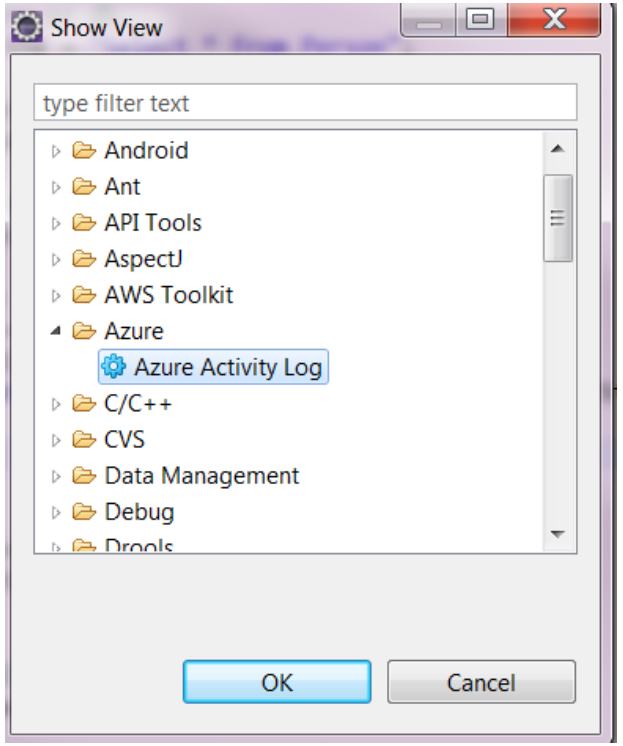
Run it locally and verify the output:

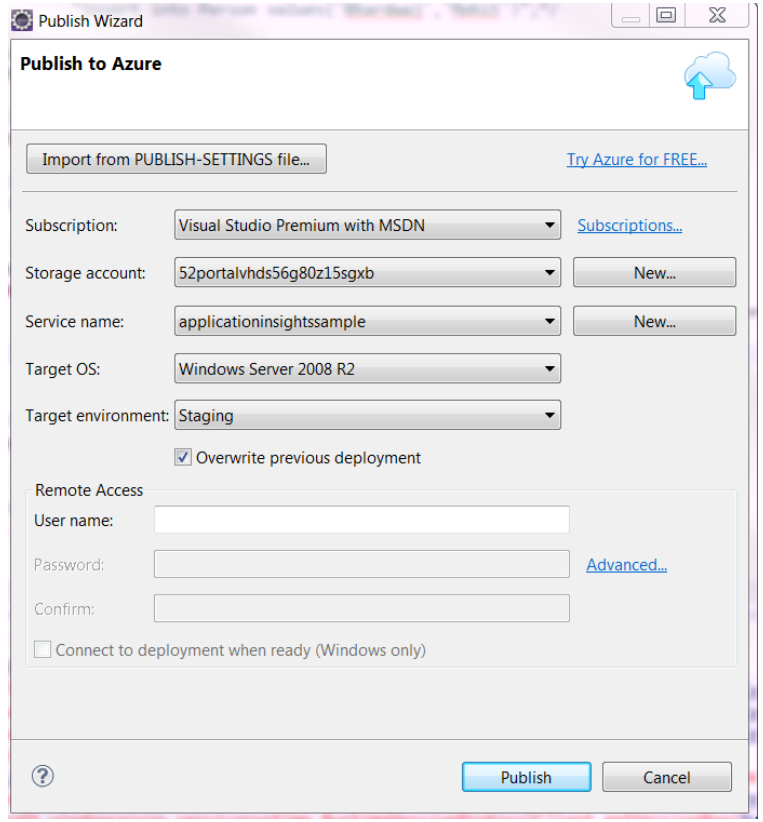
Deploy Java App as Azure Cloud Service

Select the project and on Eclipse toolbar, click **"Publish as Cloud Service"** button or choose from right click menu -> Azure -> Publish to Azure option.

In the Publish to Azure dialog:

- For the first time, **Import Subscription Information** by clicking on **Download Publish Settings file**.
- Then using **Browse** button, select the downloaded subscription file.
- You may change the storage account, service name, target OS and other options.
- Click OK
- See the status using **Azure Activity Log**
- Once published on cloud, you can open the public URL and see the application in working.





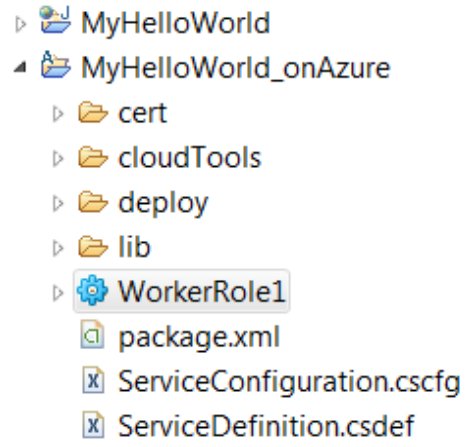
Deploy Java App as Azure Cloud Service

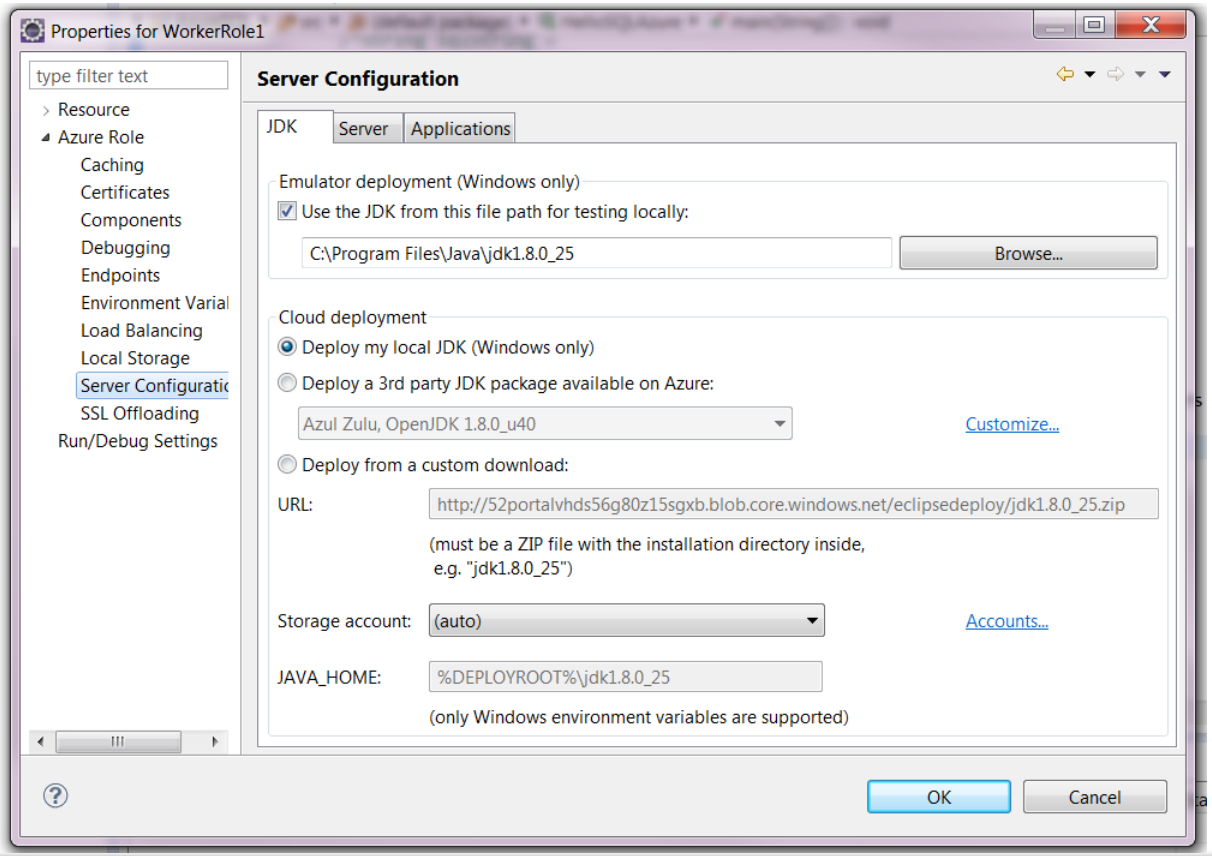
Learnings:

To deploy one or more Java applications, an Azure Deployment Project is needed.

To change the JDK or settings of the project:

- Expand the Azure Wrapper Project
- Right click WorkerRole1
- Expand the Azure submenu
- Click Server Configuration

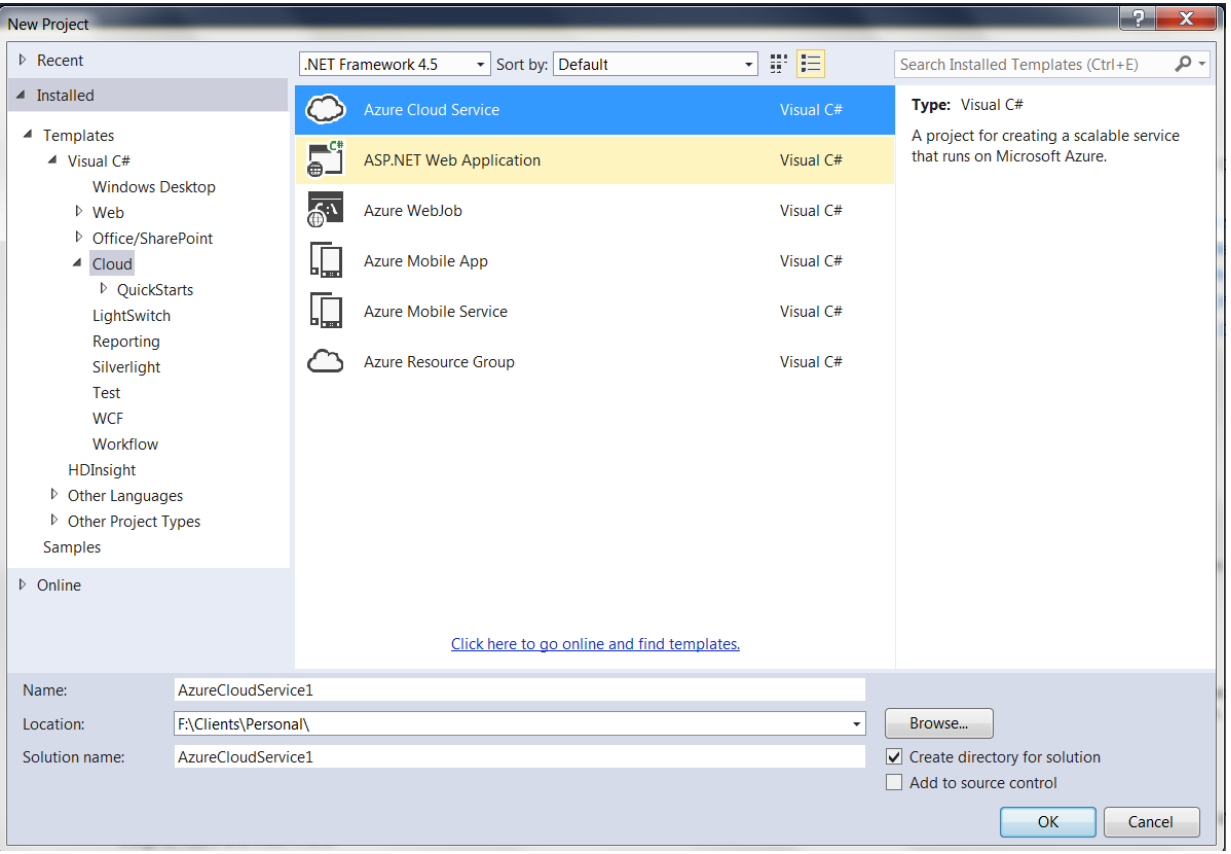


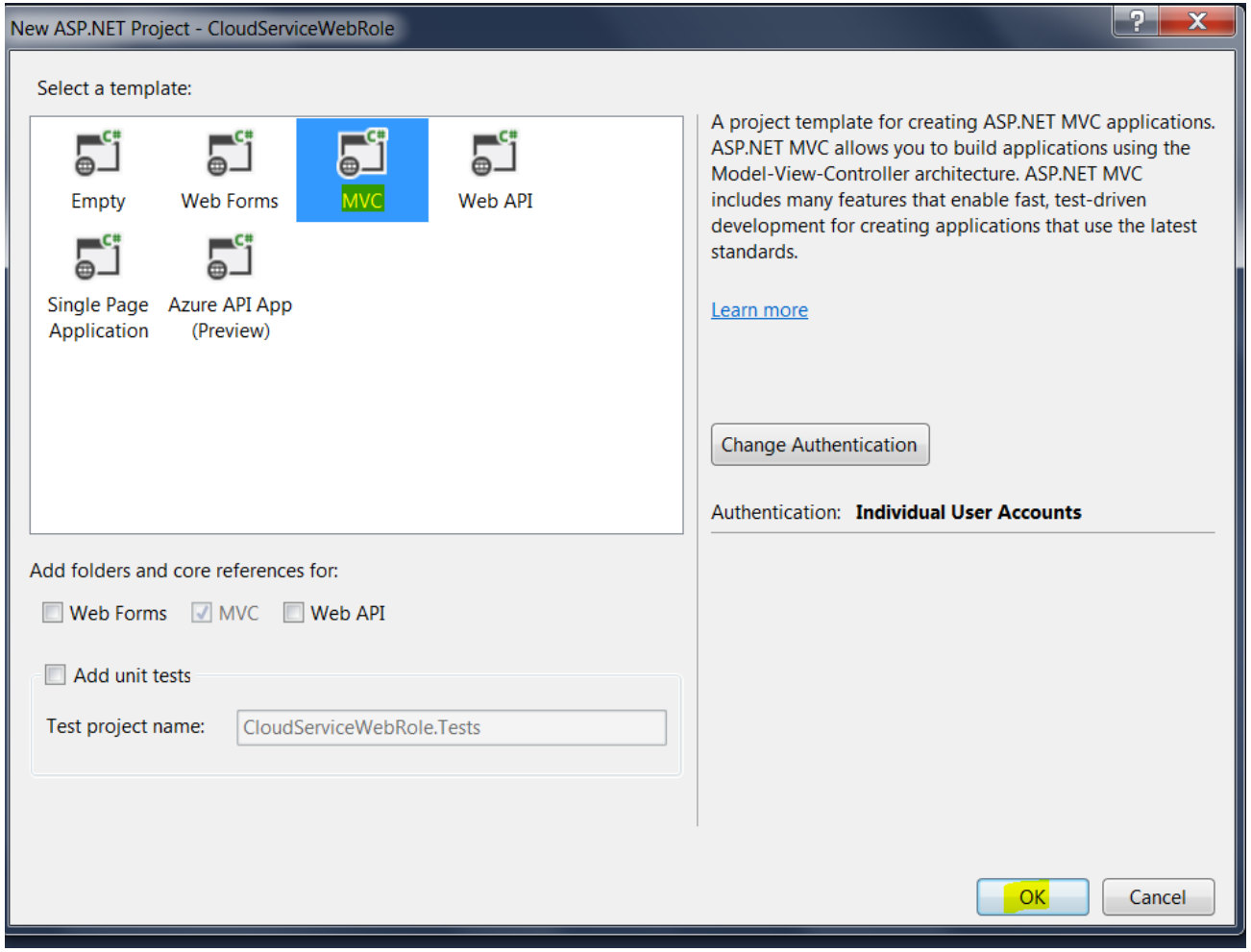


.NET Implementation

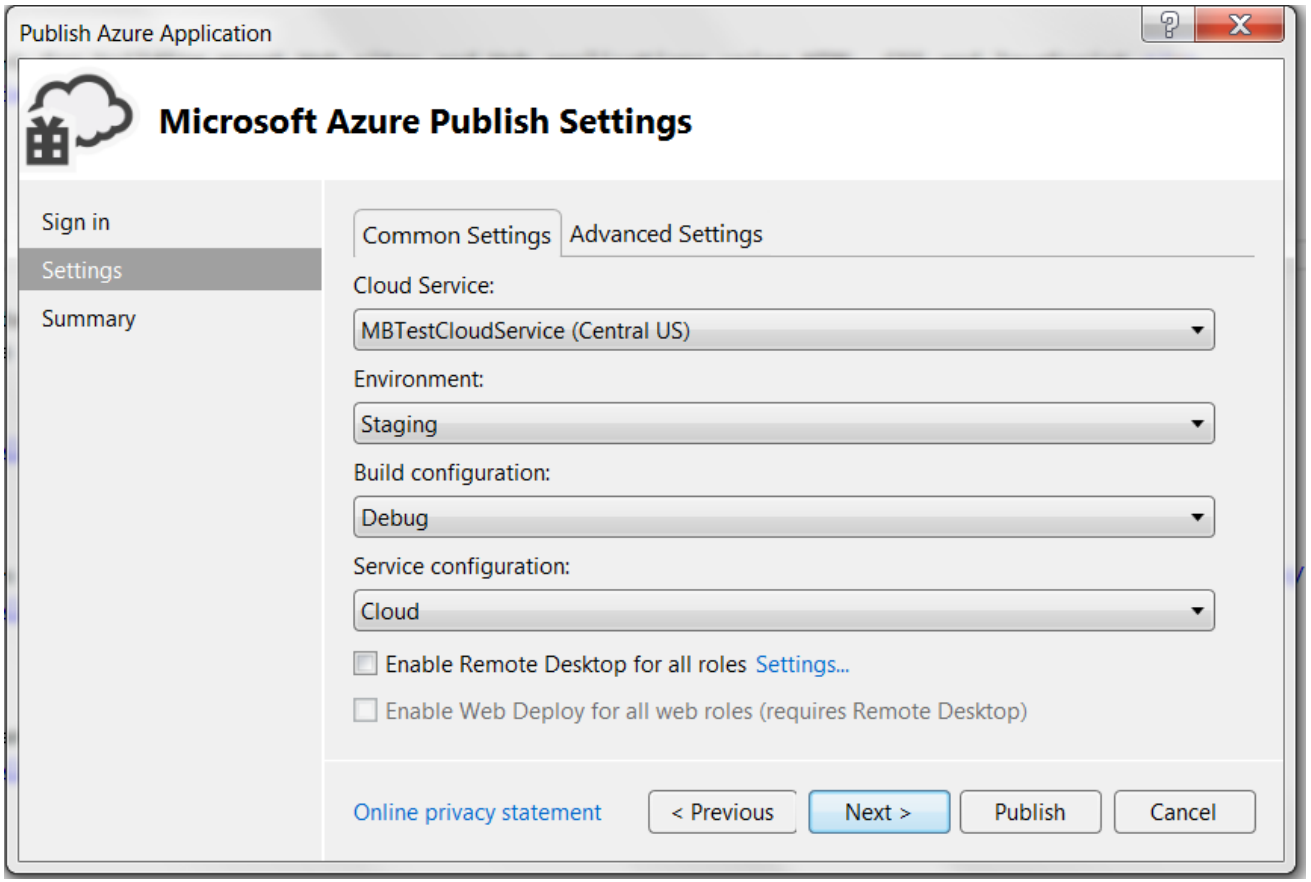
Cloud Service (Web Role)

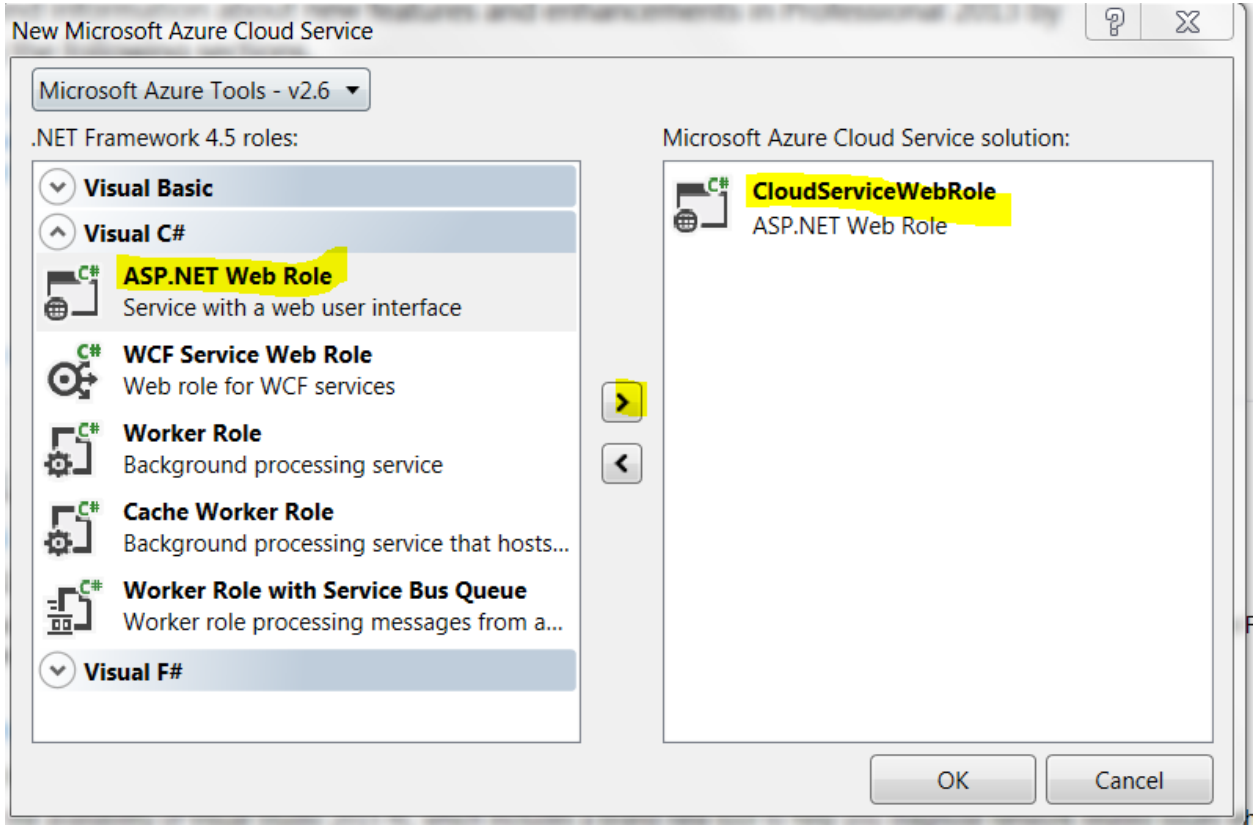
Cloud service



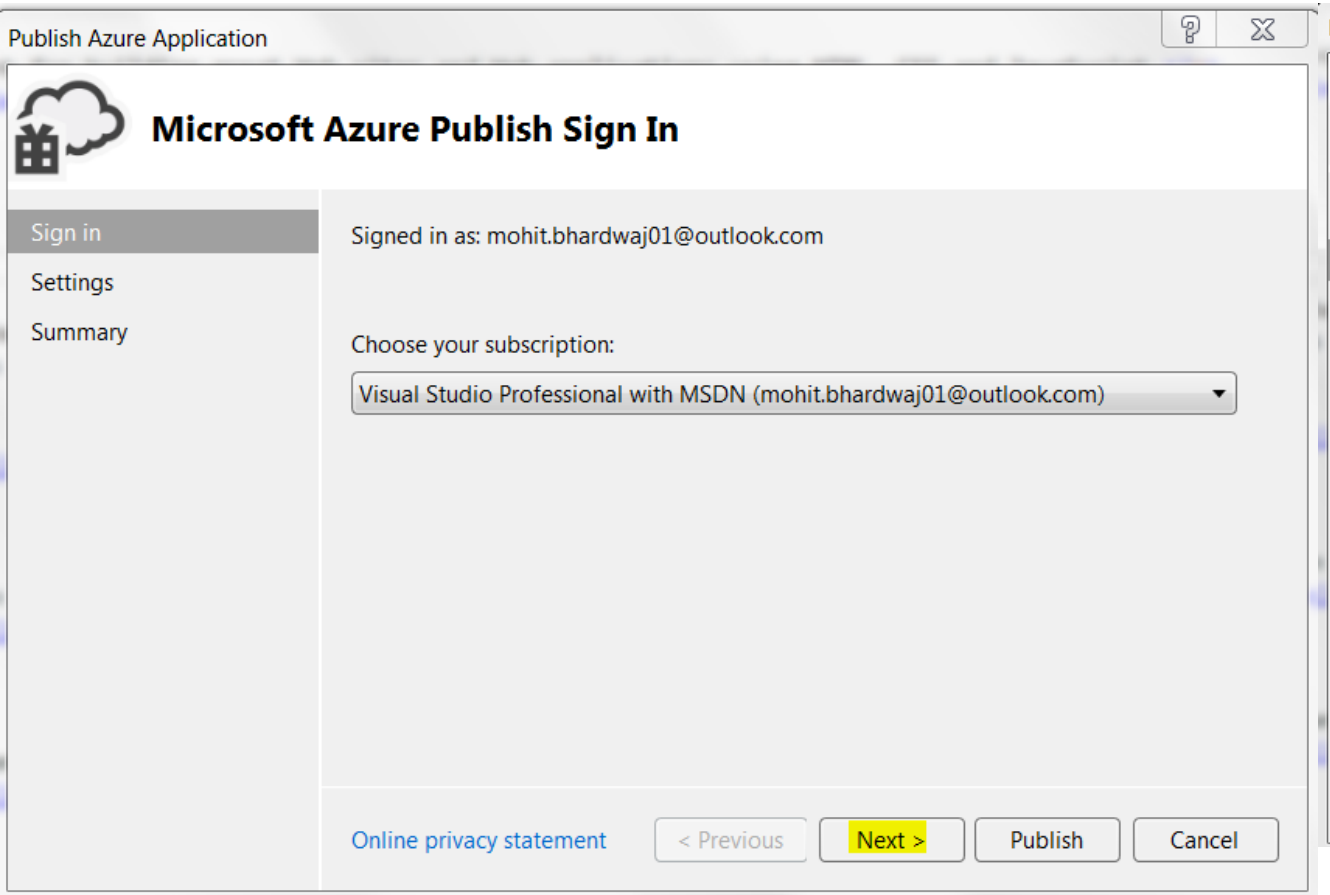


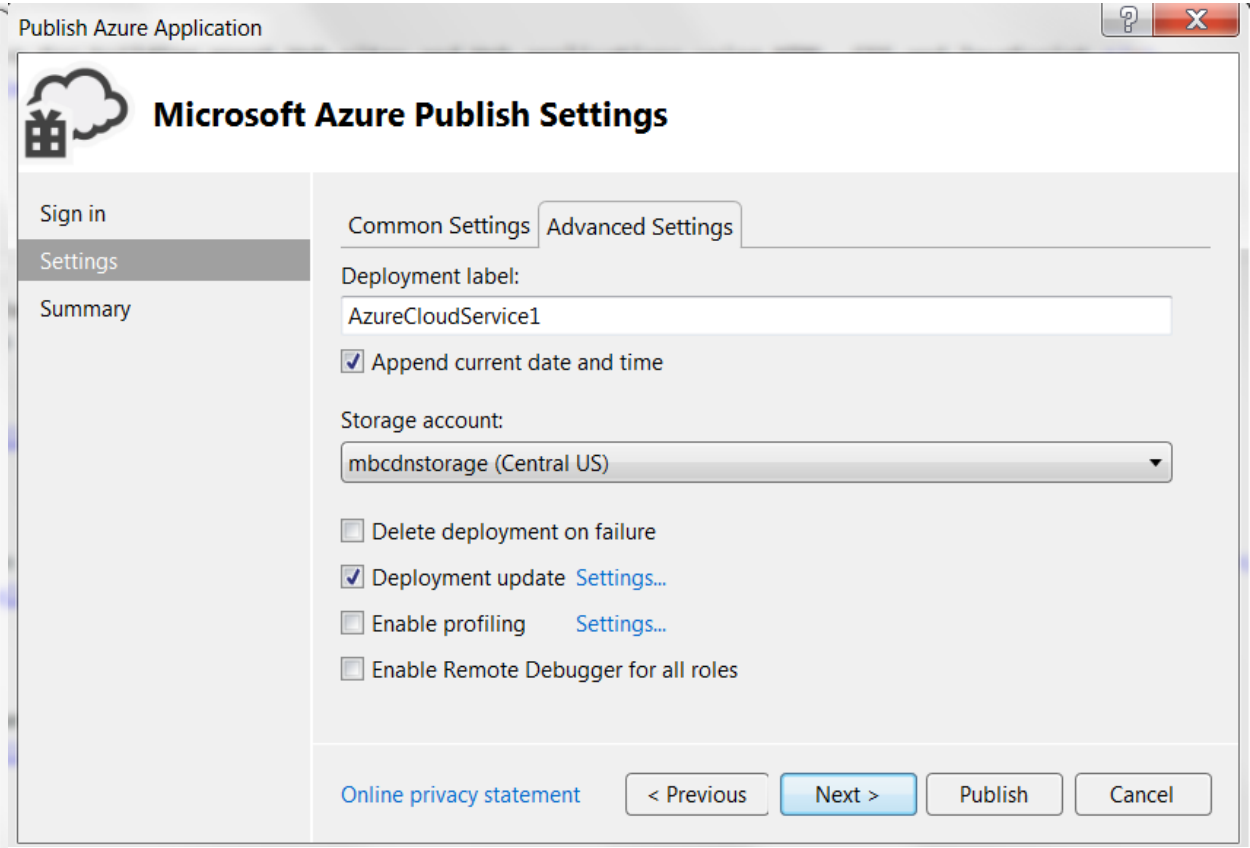
Cloud service

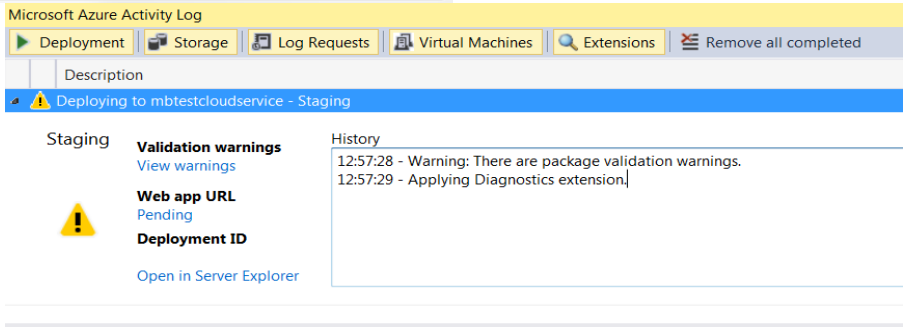




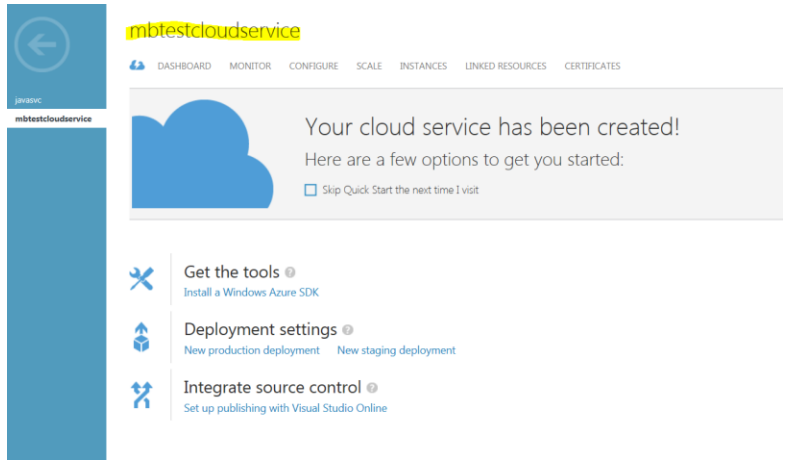
Cloud service

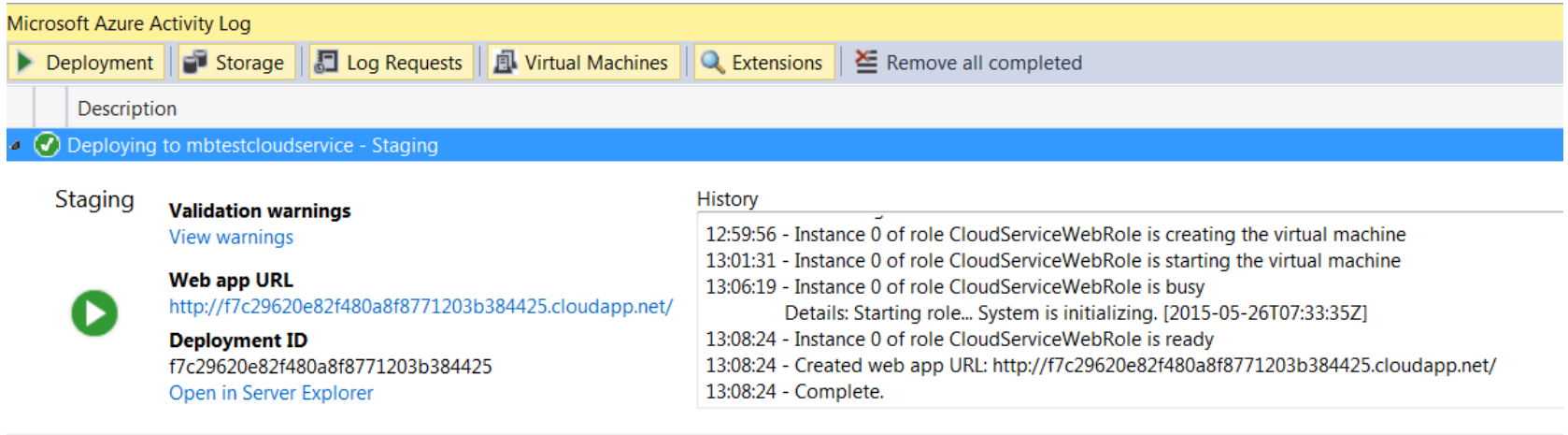




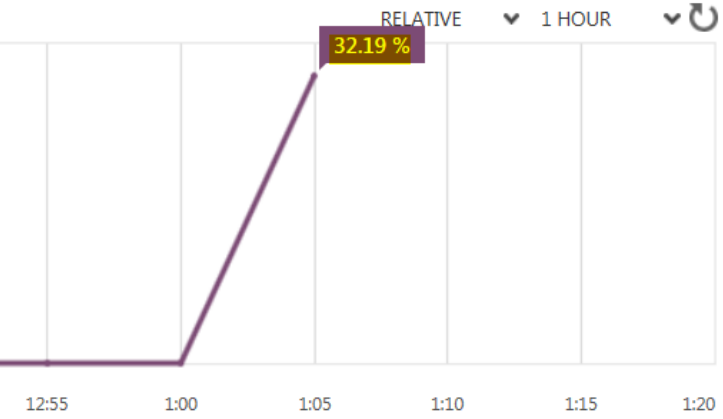


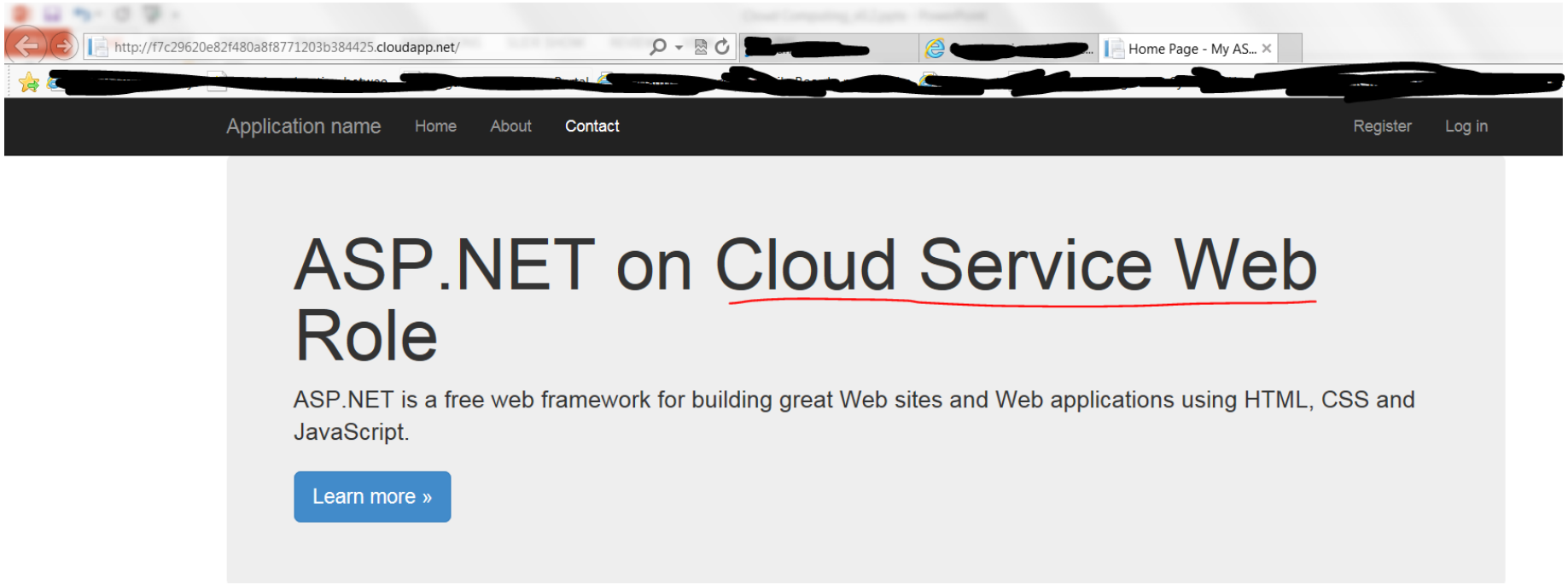
Cloud service





SOURCES CERTIFICATES





Working with Azure SQL database

Azure SQL Database

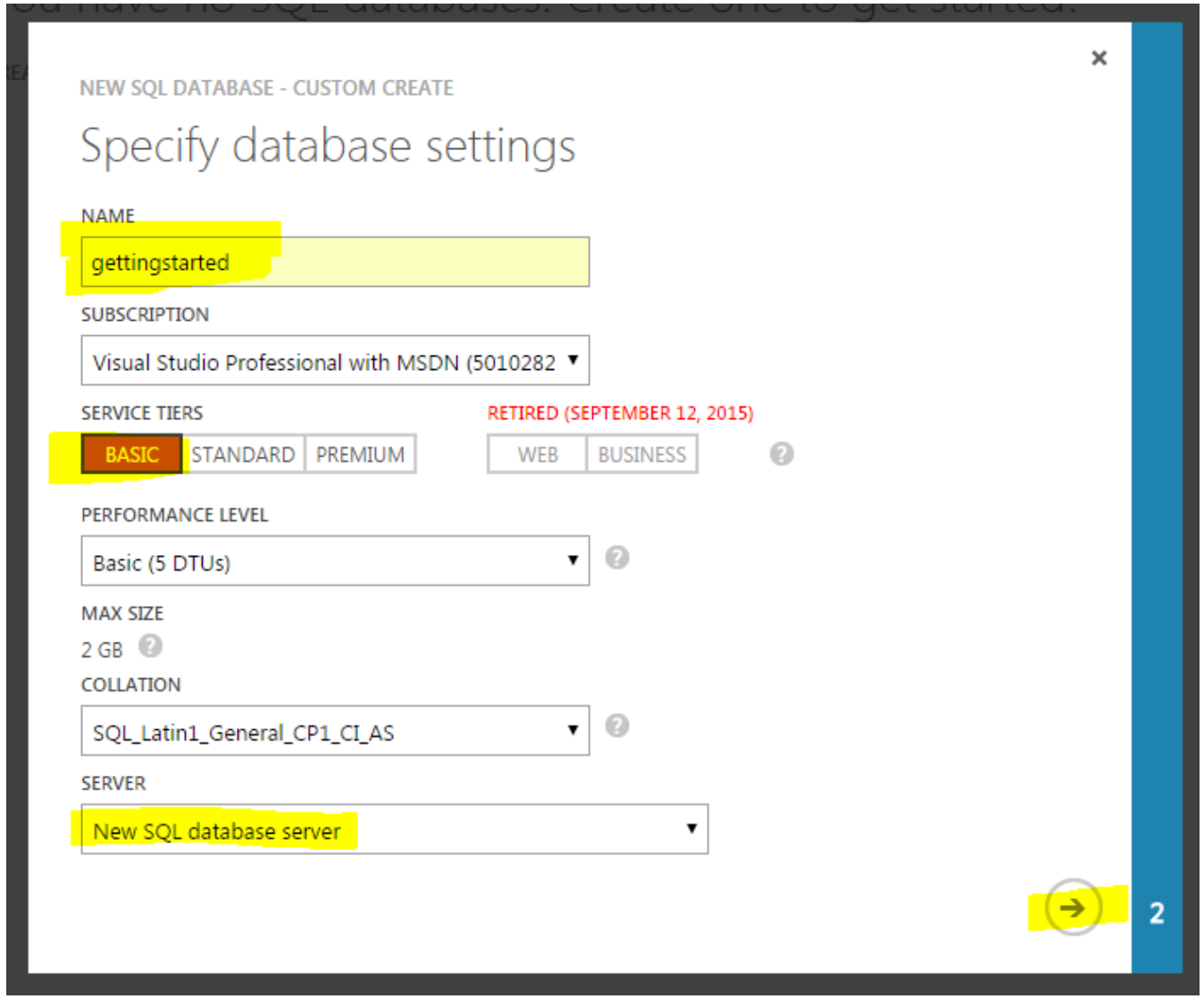
Azure SQL Database is RDBMS for Azure based on SQL Server technology.

Create an Azure SQL Database

Login to Azure Management Portal

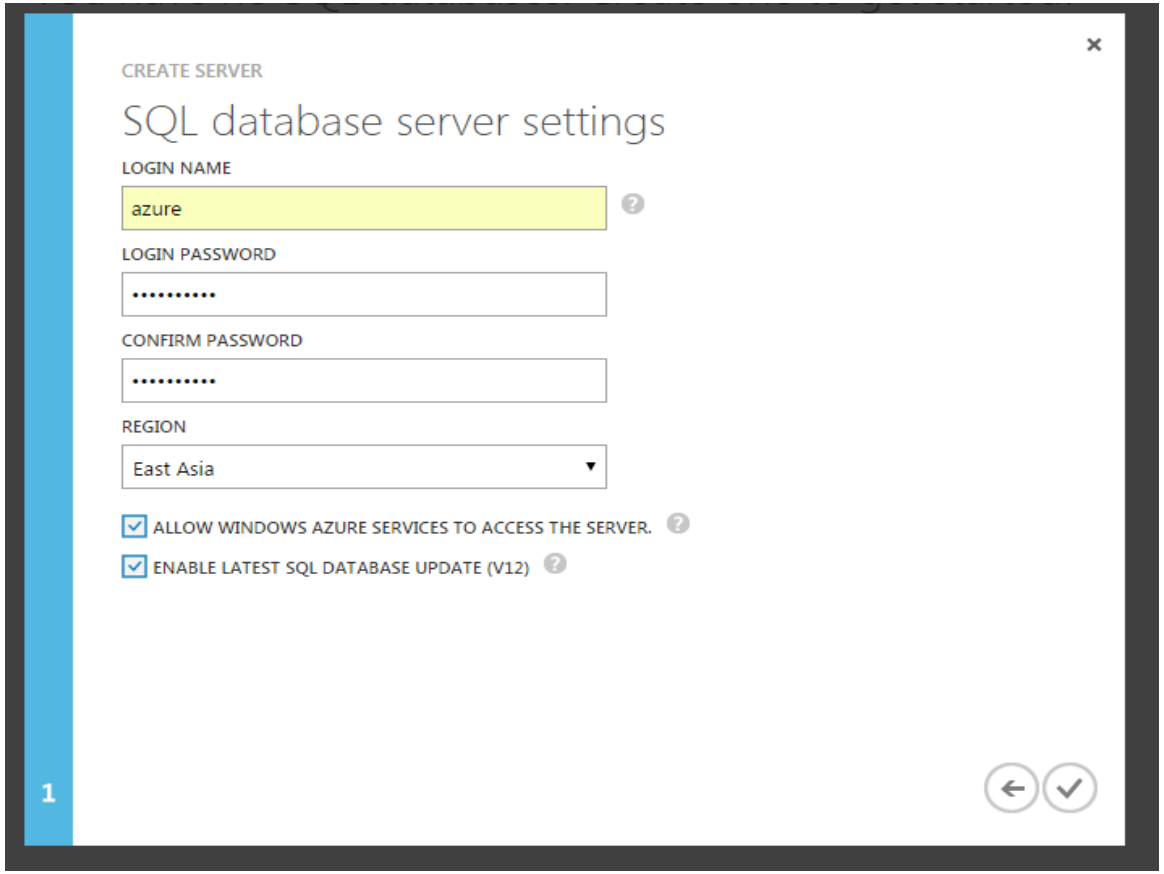
Click **SQL Databases** -> **Create SQL Database**

Enter database name as “gettingstarted”

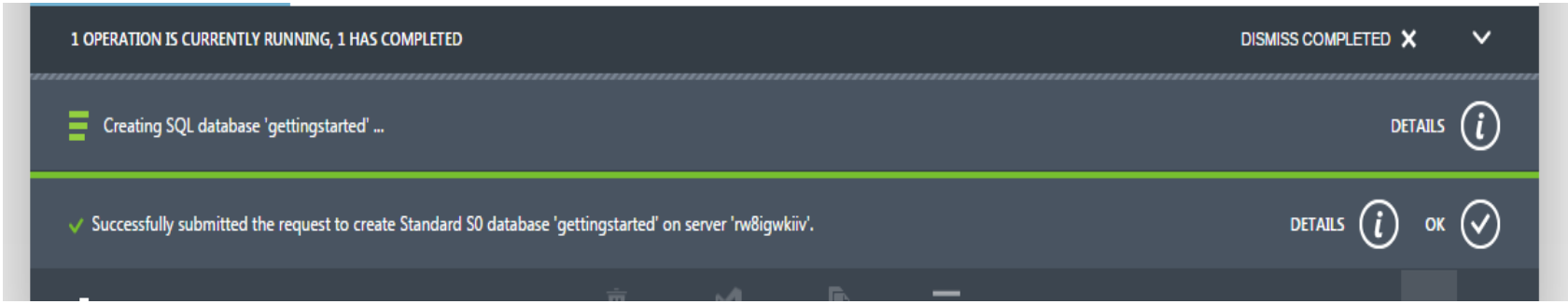


Azure SQL Database

Update the settings for server: Region, Login name and Password and click on completion button.

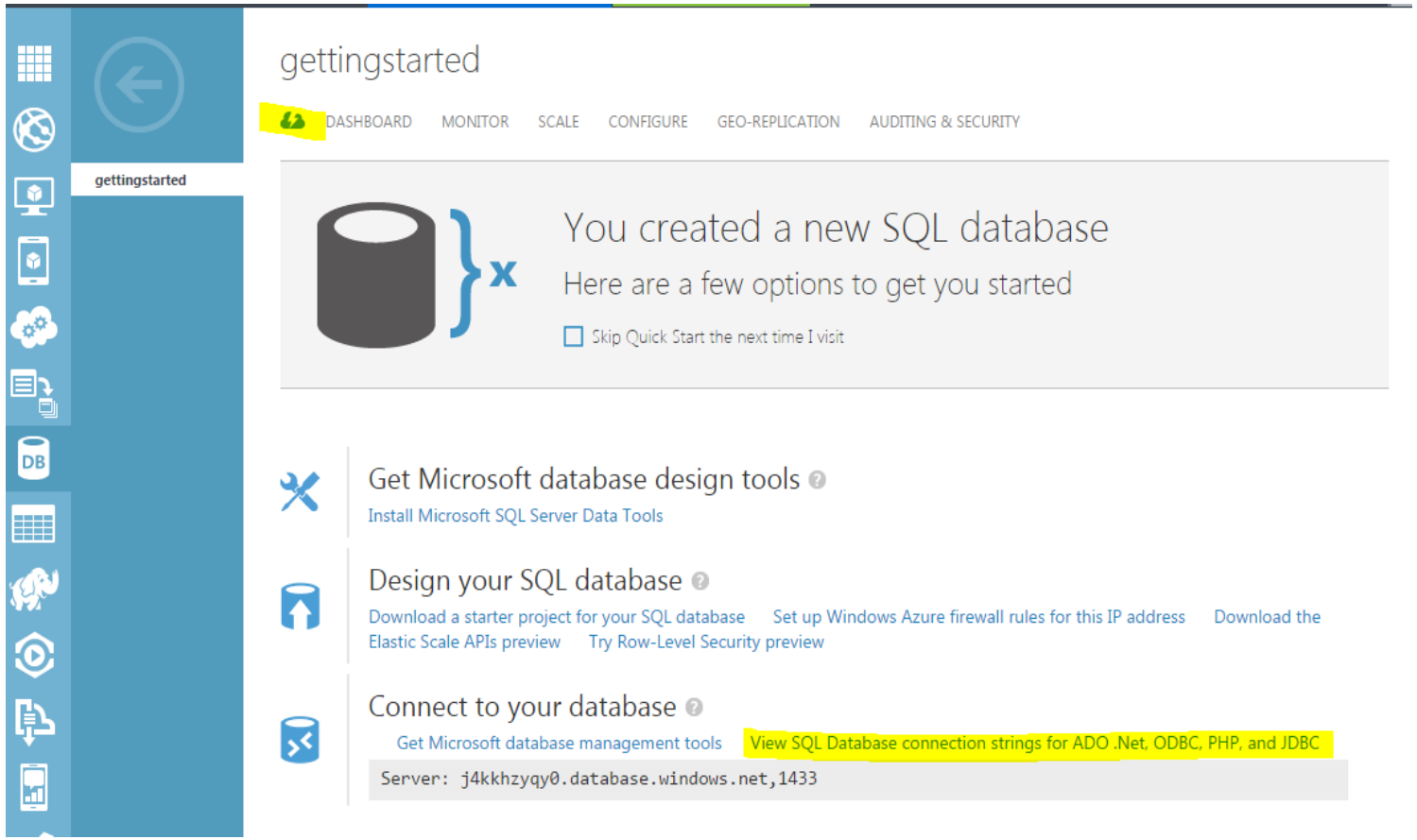


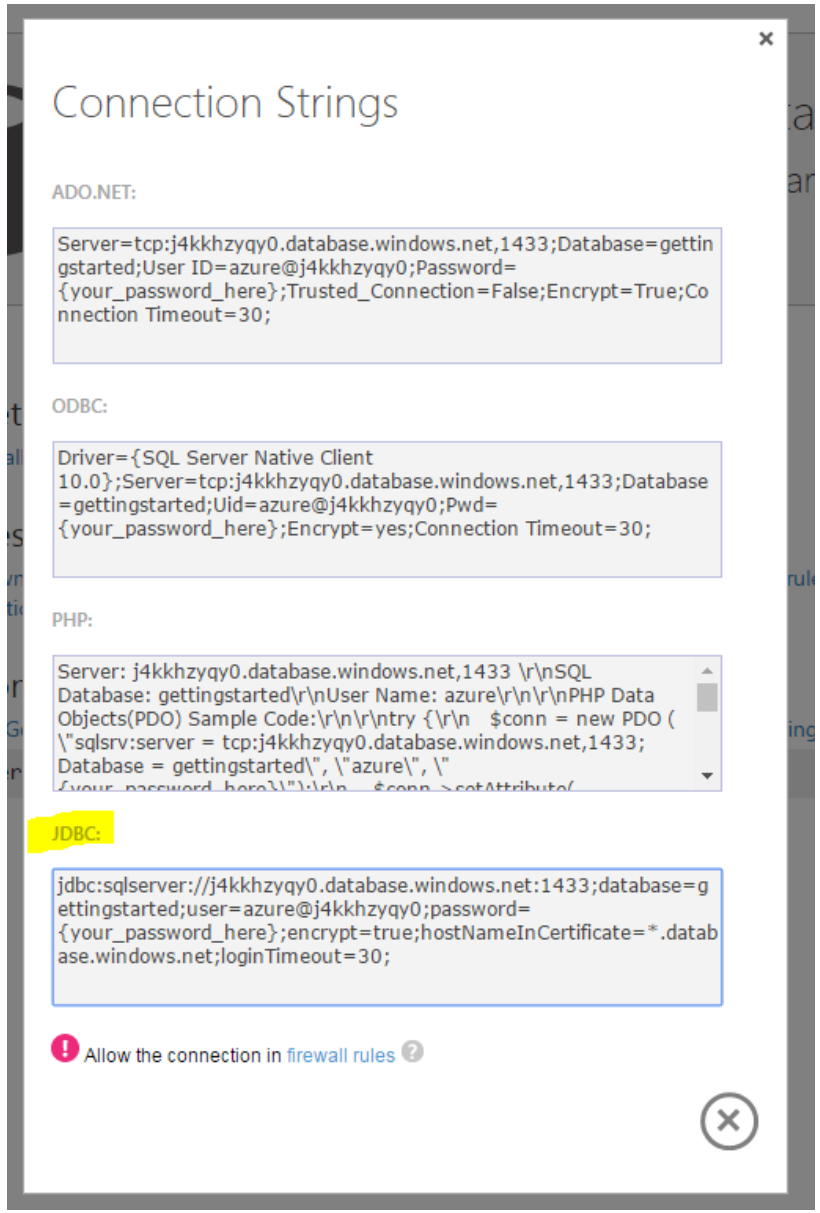
You will see a successful creation notification



Azure SQL Database

- Copy the connection strings (JDBC option)





Use the Azure Db connection String in Java code

TODO activity

a. Create a Java class which creates a table on Azure Cloud

b. Insert some records in table

c. Query the cloud table and print the result set

zure.java

QL ▸ src ▸ (default package) ▸ HelloSQLAzure ▸ main(String[]) : void

```
java.sql.*;
com.microsoft.sqlserver.jdbc.*;

class HelloSQLAzure {
ublic static void main(String[] args)
 {

      // Connection string for your SQL Database server.
      // Change the values assigned to your_server,
      // your_user@your_server,
      // and your_password.
      String connectionString =
          "jdbc:sqlserver://rw8igwkiiv.database.windows.net:1433;" +
          "database=gettingstarted;user=azure@rw8igwkiiv;" +
          "password=Microsoft1;encrypt=true;hostNameInCertificate=japaneast1-a.control.database.windows.net;loginTimeout=30";

      // The types for the following variables are
      // defined in the java.sql library.
      Connection connection = null;  // For making the connection
      Statement statement = null;    // For the SQL statement
      ResultSet resultSet = null;    // For the result set, if applicable

      try
      {
          // Ensure the SQL Server driver class is available.
          Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");

          // Establish the connection
```

Javadoc | Declaration | Search | Console | Call Hierarchy | Servers | History | Azure Activity Log

HelloSQLAzure [Java Application] C:\Program Files\Java\jre1.8.0_25\bin\javaw.exe (04-Jul-2016 2:20:04 PM)

```
Cannot open server 'rw8igwkiiv' requested by the login. Client with IP address '103.56.173.1' is not allowed to access the serv
oft.sqlserver.jdbc.SQLServerException: Cannot open server 'rw8igwkiiv' requested by the login. Client with IP address '103.56.1
: com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:196)
: com.microsoft.sqlserver.jdbc.TDSTokenHandler.onEOF(tdsparser.java:246)
: com.microsoft.sqlserver.jdbc.TDSParser.parse(tdsparser.java:83)
: com.microsoft.sqlserver.jdbc.SQLServerConnection.sendLogon(SQLServerConnection.java:2532)
: com.microsoft.sqlserver.jdbc.SQLServerConnection.logon(SQLServerConnection.java:1929)
: com.microsoft.sqlserver.jdbc.SQLServerConnection.access$000(SQLServerConnection.java:41)
: com.microsoft.sqlserver.jdbc.SQLServerConnection$LogonCommand.doExecute(SQLServerConnection.java:1917)
: com.microsoft.sqlserver.jdbc.TDSCommand.execute(IOBuffer.java:4026)
```

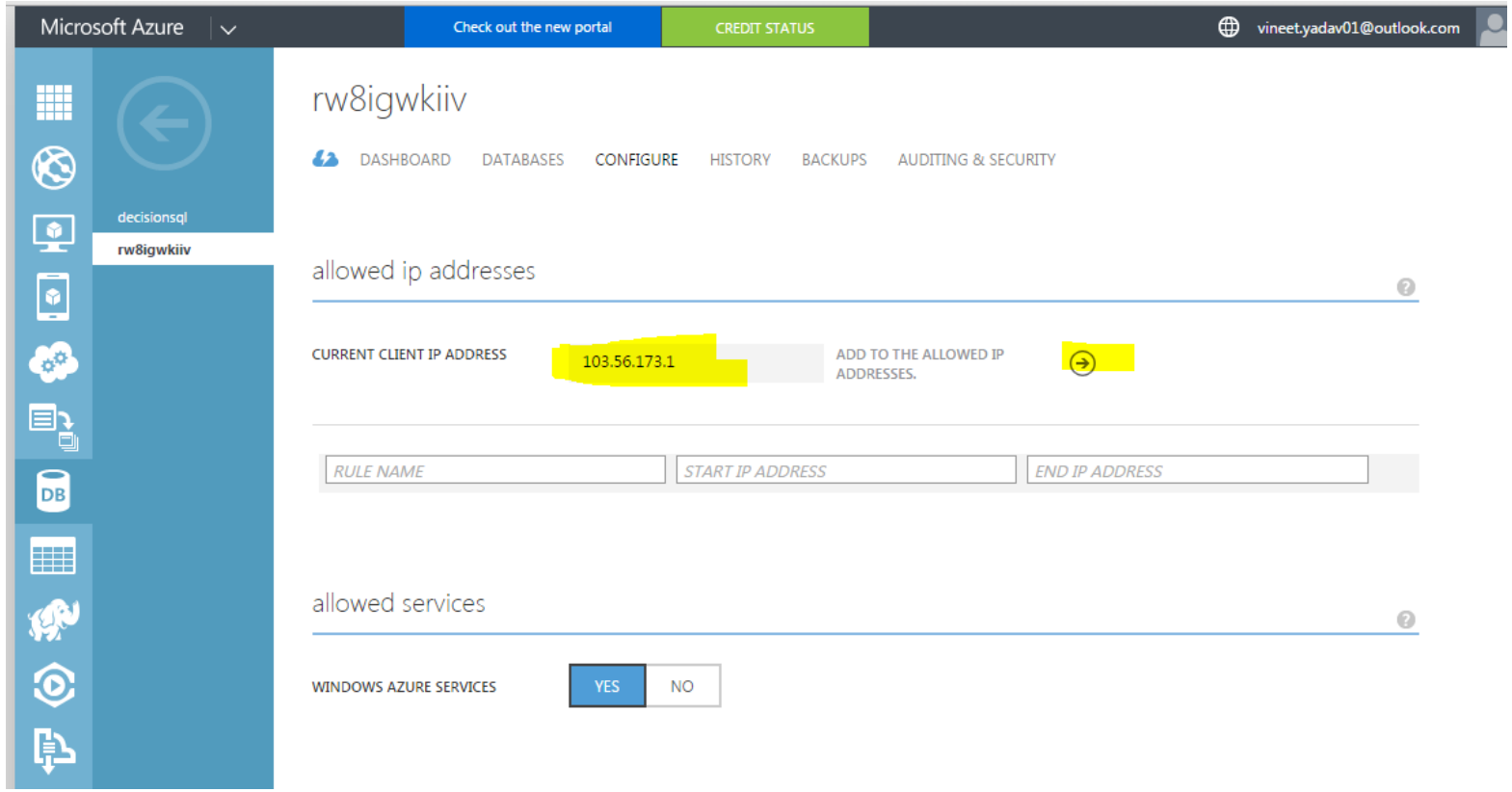
To allow access to range of IP address

Login to Azure Management Portal

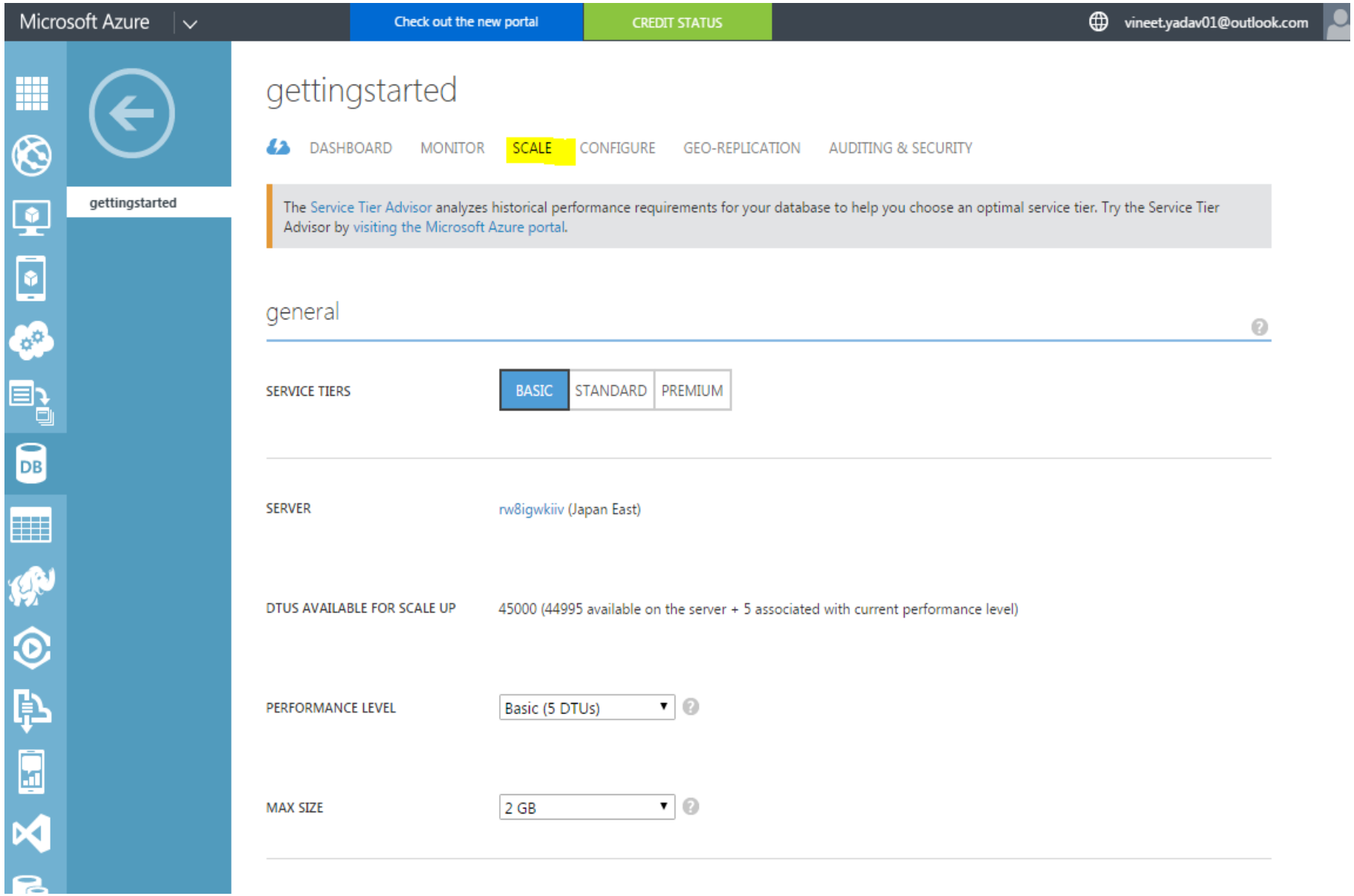
Click **SQL Database** and click **Configure**

Under Allowed IP address, select the **Current Client IP address**

Now re-execute Java code, it works fine!



Scaling options on Azure Cloud for SQL Database



Execute code & Monitor the activity on Cloud

HelloSQLAzure.java

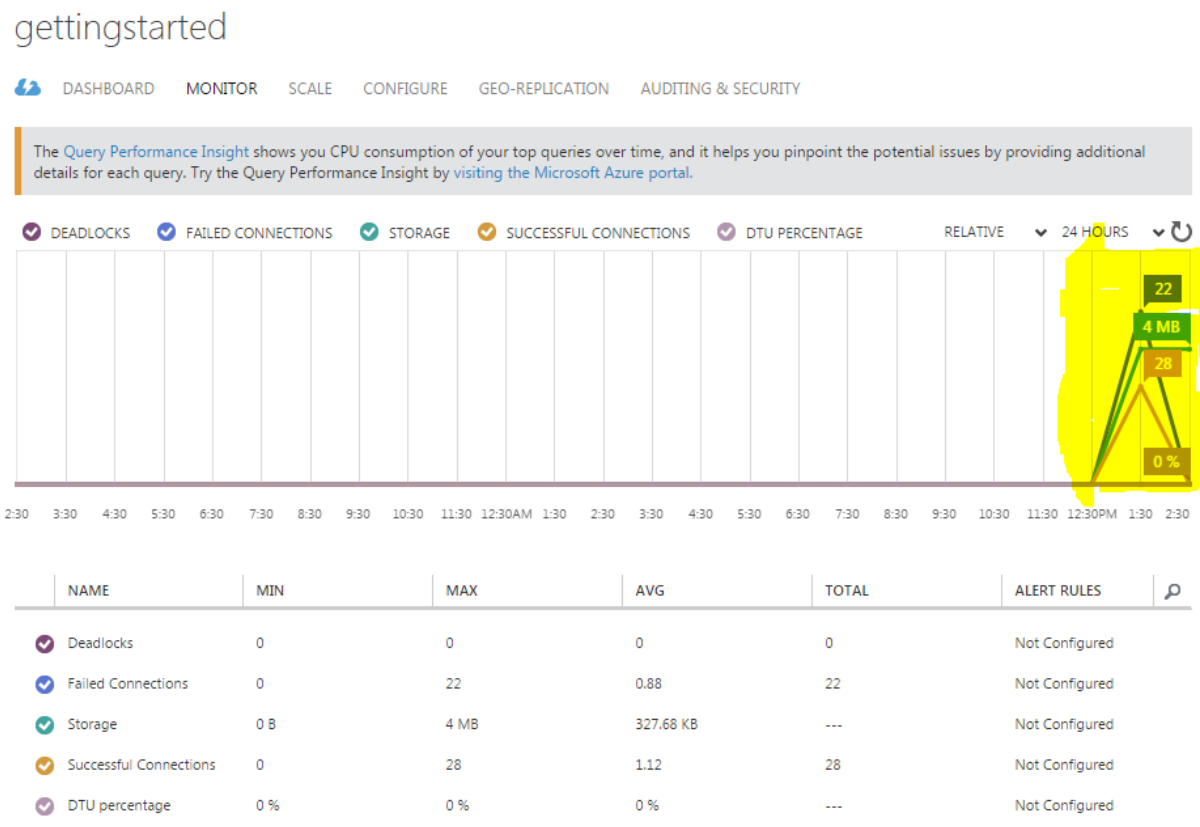
AzureSQL ▸ src ▸ (default package) ▸ HelloSQLAzure ▸ main(String[]) : void

```
            /*String sqlString =
                    "insert into Person values('Bhardwaj','Mohit')";*/

        // Query SQL query on db.
            String sqlString = "Select * from Person";

            // Use the connection to create the SQL statement.
            statement = connection.createStatement();

            // Execute the statement for option 1 & 2
            //statement.executeUpdate(sqlString);

           // Execute the statement for option 3
            ResultSet rs = statement.executeQuery(sqlString);
            while(rs.next()){
                System.out.println(rs.getInt("PersonID")+" "+rs.getString("LastName")+" "+rs.getString("FirstName"));

            }


            // Provide a message when processing is complete.
            System.out.println("Processing complete.");

        }
        // Exception handling
        catch (ClassNotFoundException cnfe)
        {
```

Problems @ Javadoc Declaration Search Console Call Hierarchy Servers History Azure Activity Log

<terminated> HelloSQLAzure [Java Application] C:\Program Files\Java\jre1.8.0_25\bin\javaw.exe (04-Jul-2016 2:42:47 PM)

```
1 Balhara Ashish
2 Bhardwaj Mohit
Processing complete.
```

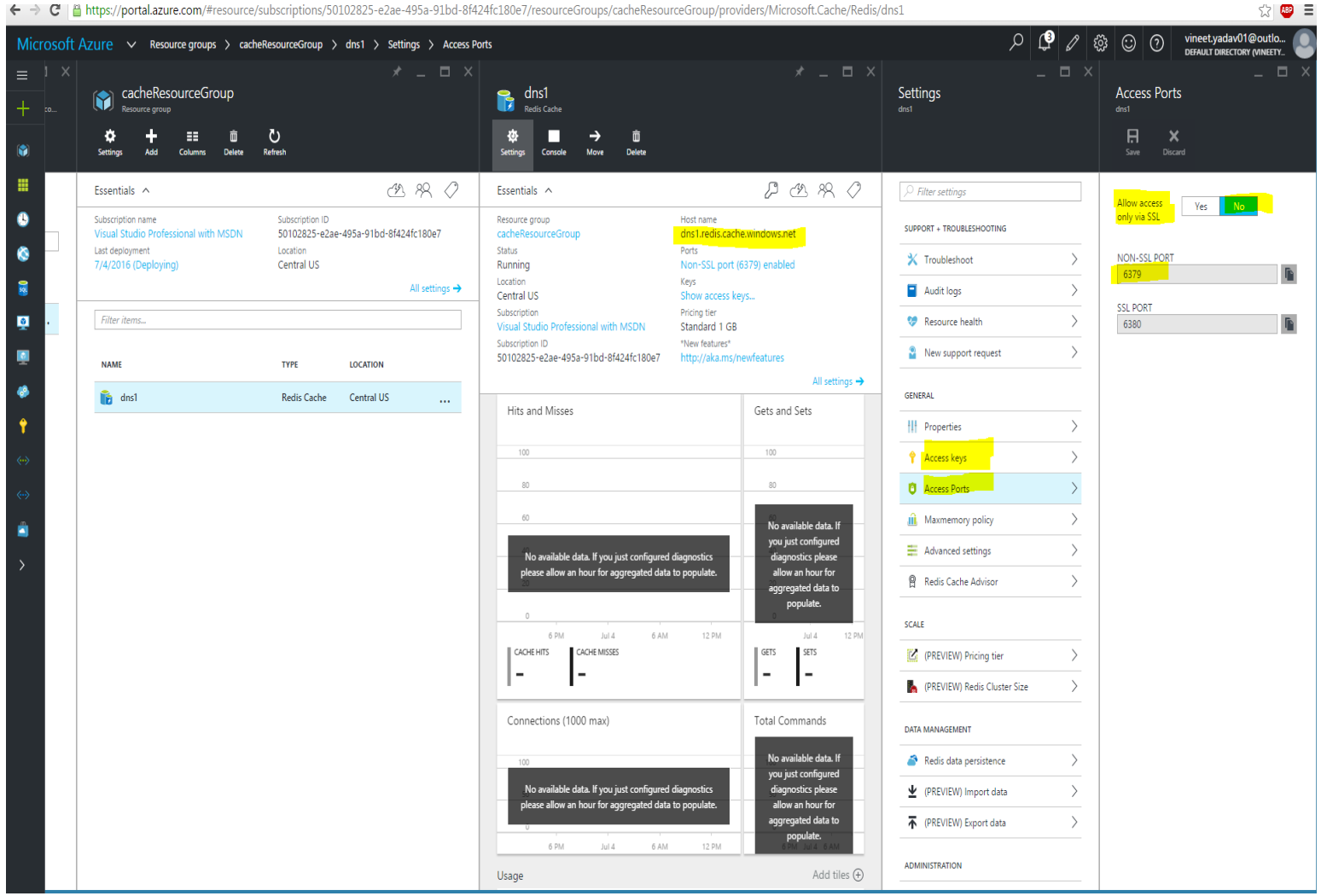


NAME	MIN	MAX	AVG	TOTAL	ALERT RULES
Deadlocks	0	0	0	0	Not Configured
Failed Connections	0	22	0.88	22	Not Configured
Storage	0 B	4 MB	327.68 KB	---	Not Configured
Successful Connections	0	28	1.12	28	Not Configured
DTU percentage	0 %	0 %	0 %	---	Not Configured

Working with Redis Cache on Azure

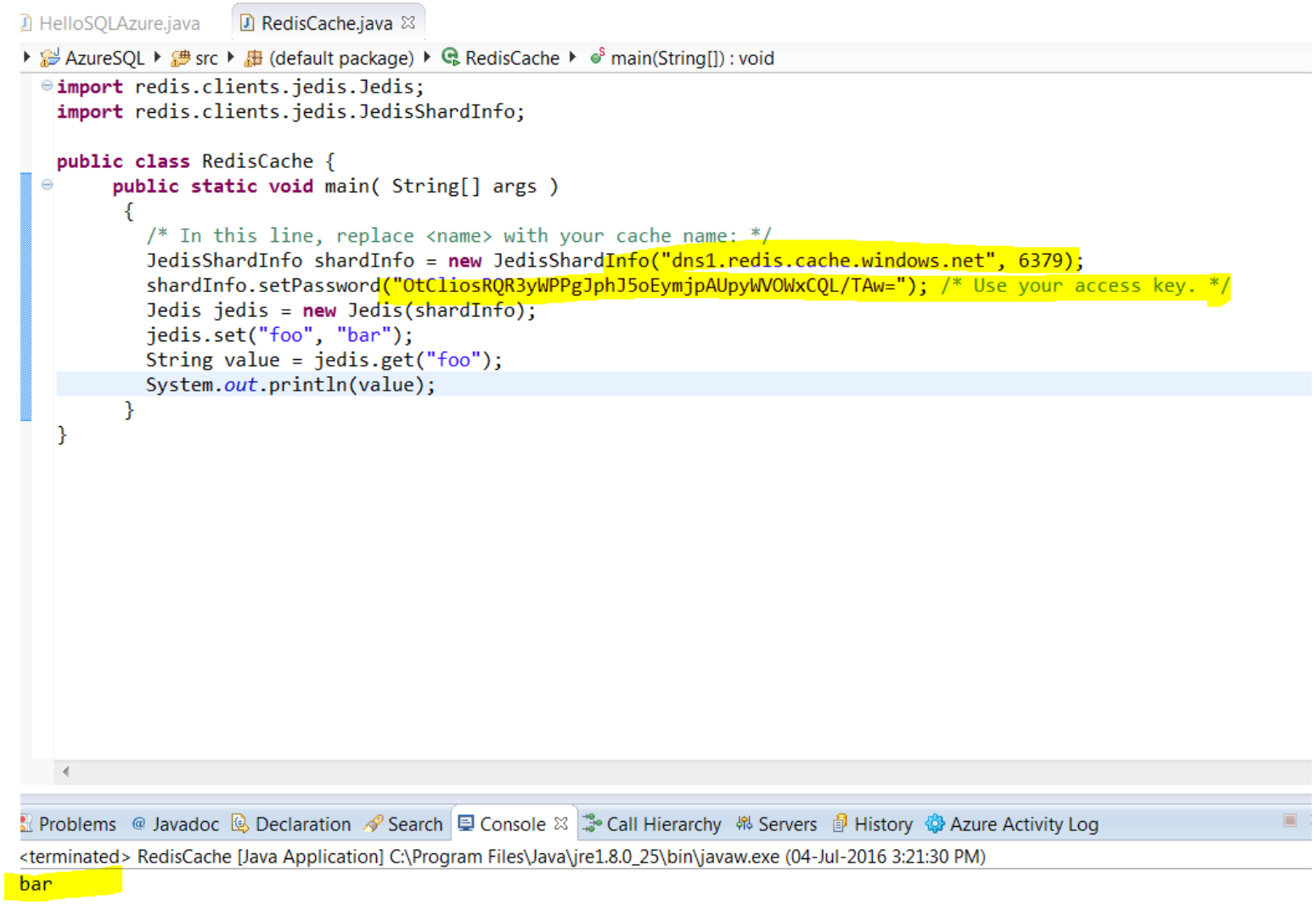
Azure Redis Cache

Create a Redis Cache Service and configure it

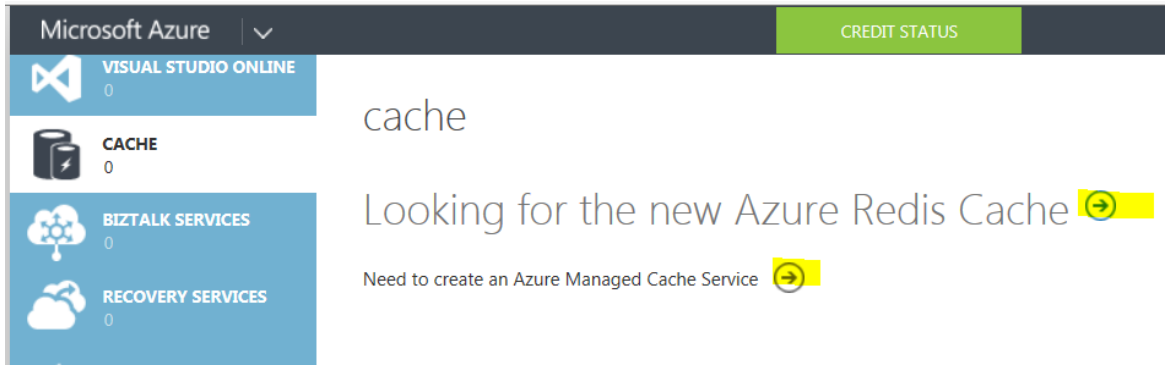


Azure Redis Cache via Java Code

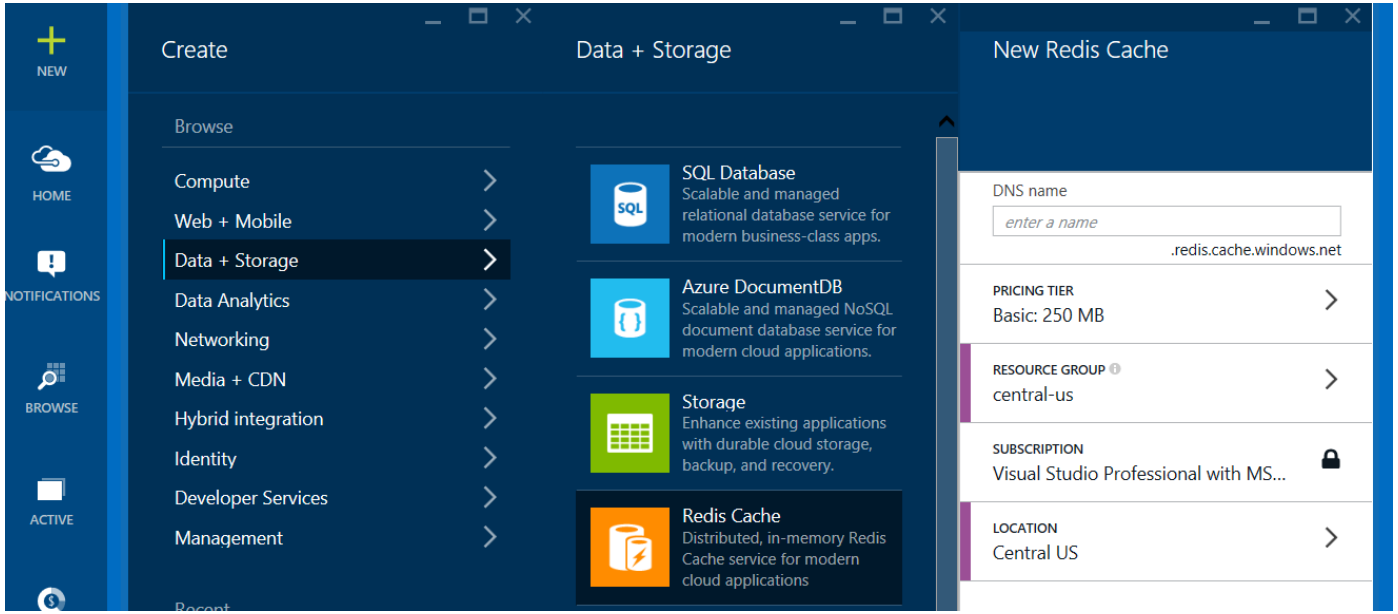
Create a Java class file to access the Azure Redis Cache service and populate it.



Redis Cache



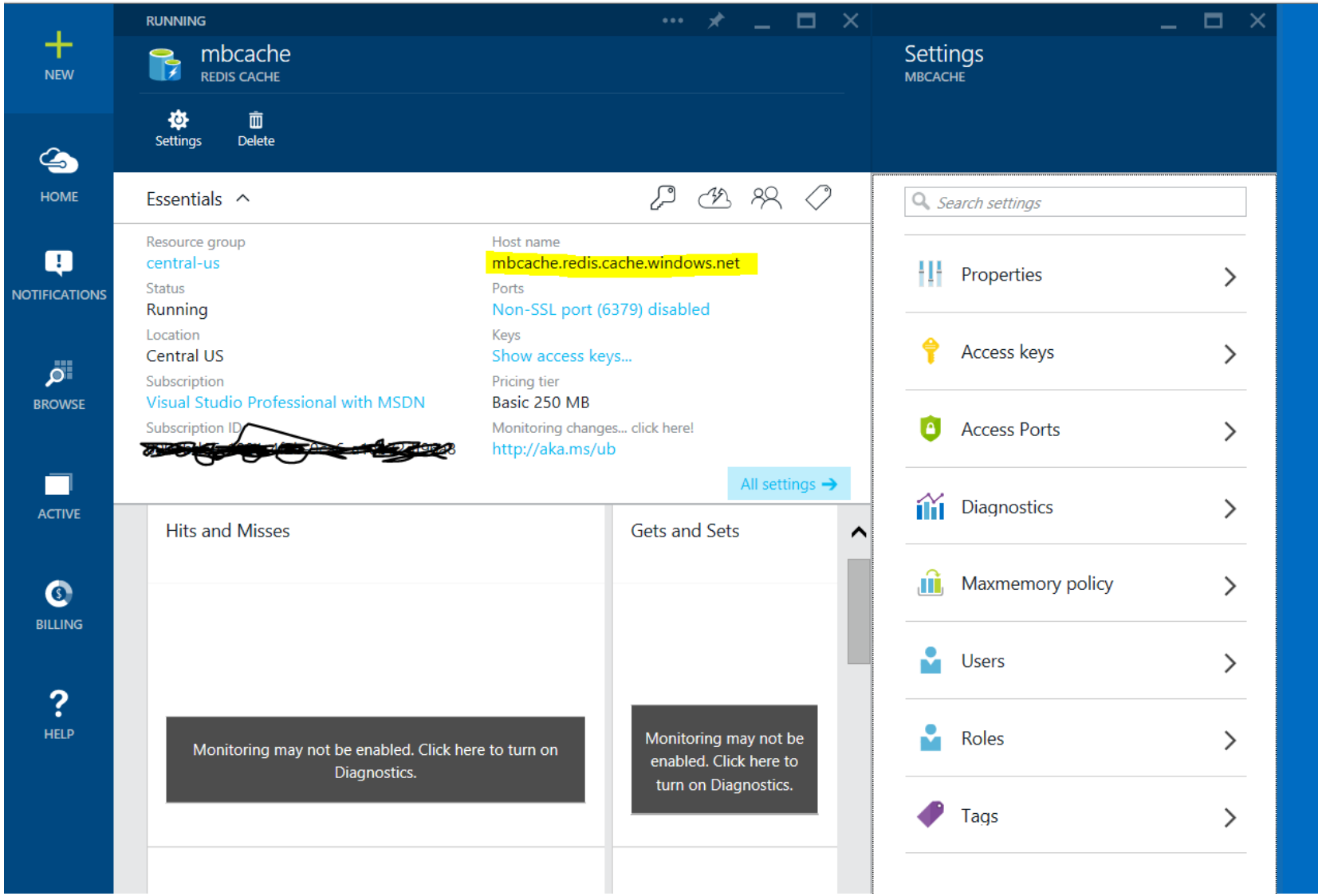
If you have configured Azure PowerShell with a certificate for your account then you can skip this step. For more information about connecting Azure PowerShell with your Azure account, see [How to install and configure Azure PowerShell](#).



C#

csredis	Async (and sync) client for Redis and Sentinel	
Nhiredis	A lightweight wrapper around the C client hiredis.	
redis-sharp		
redisboost	Thread-safe async Redis client. Offers high performance and simple api	
ServiceStack.Redis	This is a fork and improvement of the original C# client written by Miguel De Icaza.	
Sider	Minimalistic client for C#/.NET 4.0	
StackExchange.Redis	This .NET client was developed by Stack Exchange for very high performance needs (replacement to the earlier BookSleeve).	
TeamDev Redis Client	Redis Client is based on redis-sharp for the basic communication functions, but it offers some differences.	

Redis Cache


NEW
HOME
NOTIFICATIONS
BROWSE
ACTIVE
BILLING
HELP
RUNNING
mbcache
REDIS CACHE
Settings
Delete
Essentials
Resource group
central-us
Status
Running
Location
Central US
Subscription
Visual Studio Professional with MSDN
Subscription ID
Host name
mbcache.redis.cache.windows.net
Ports
Non-SSL port (6379) disabled
Keys
Show access keys...
Pricing tier
Basic 250 MB
Monitoring changes... click here!
http://aka.ms/ub
All settings
Hits and Misses
Gets and Sets
Monitoring may not be enabled. Click here to turn on Diagnostics.
Monitoring may not be enabled. Click here to turn on Diagnostics.
Settings
MBCACHE
Search settings
Properties
Access keys
Access Ports
Diagnostics
Maxmemory policy
Users
Roles
Tags

Redis Cache

```
PM> Install-Package StackExchange.Redis
Installing 'StackExchange.Redis 1.0.450'.
Successfully installed 'StackExchange.Redis 1.0.450'.
Adding 'StackExchange.Redis 1.0.450' to AzureSample1.
Successfully added 'StackExchange.Redis 1.0.450' to AzureSample1.
```

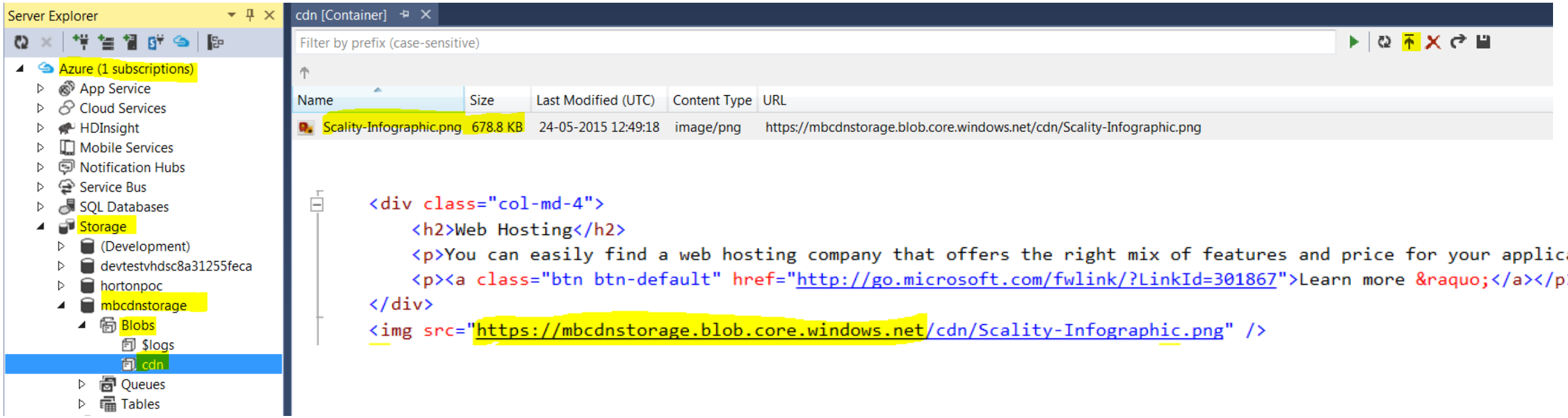
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using StackExchange.Redis;


namespace AzureSample1.Controllers
{
    public class HomeController : Controller
    {
        public ActionResult Index()
        {
            ConnectionMultiplexer connection = ConnectionMultiplexer.Connect("mbcache.redis.cache.windows.net,ssl=true,password=[redacted]");
            IDatabase cache = connection.GetDatabase();
            cache.StringSet("key1", "value");
            cache.StringSet("key2", 25);

            // Simple get of data types from the cache
            string key1 = cache.StringGet("key1");
            int key2 = (int)cache.StringGet("key2");
            return View();
        }
```

.NET Implementation

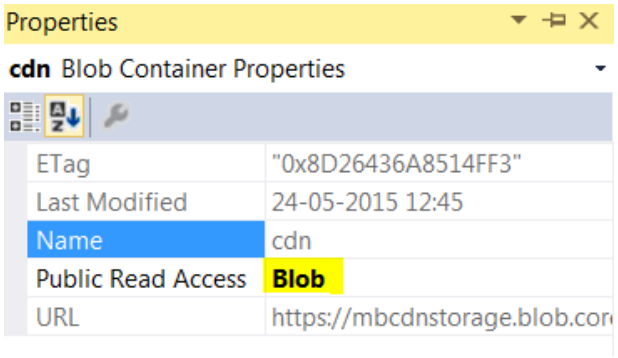
Content Delivery Network

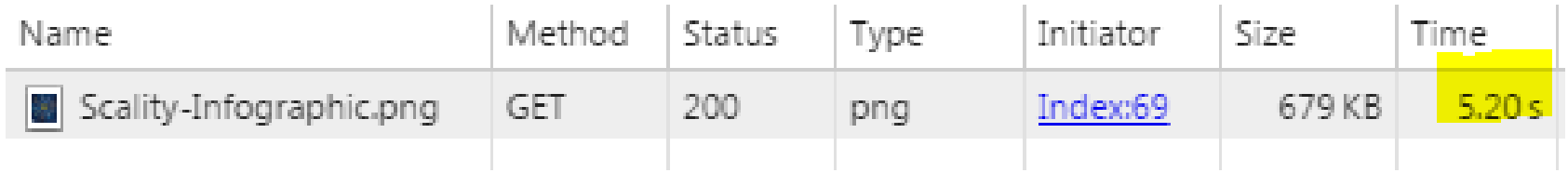
CDN

http://azure.microsoft.com/en-in/documentation/articles/cdn-serve-content-from-cdn-in-your-web-application/



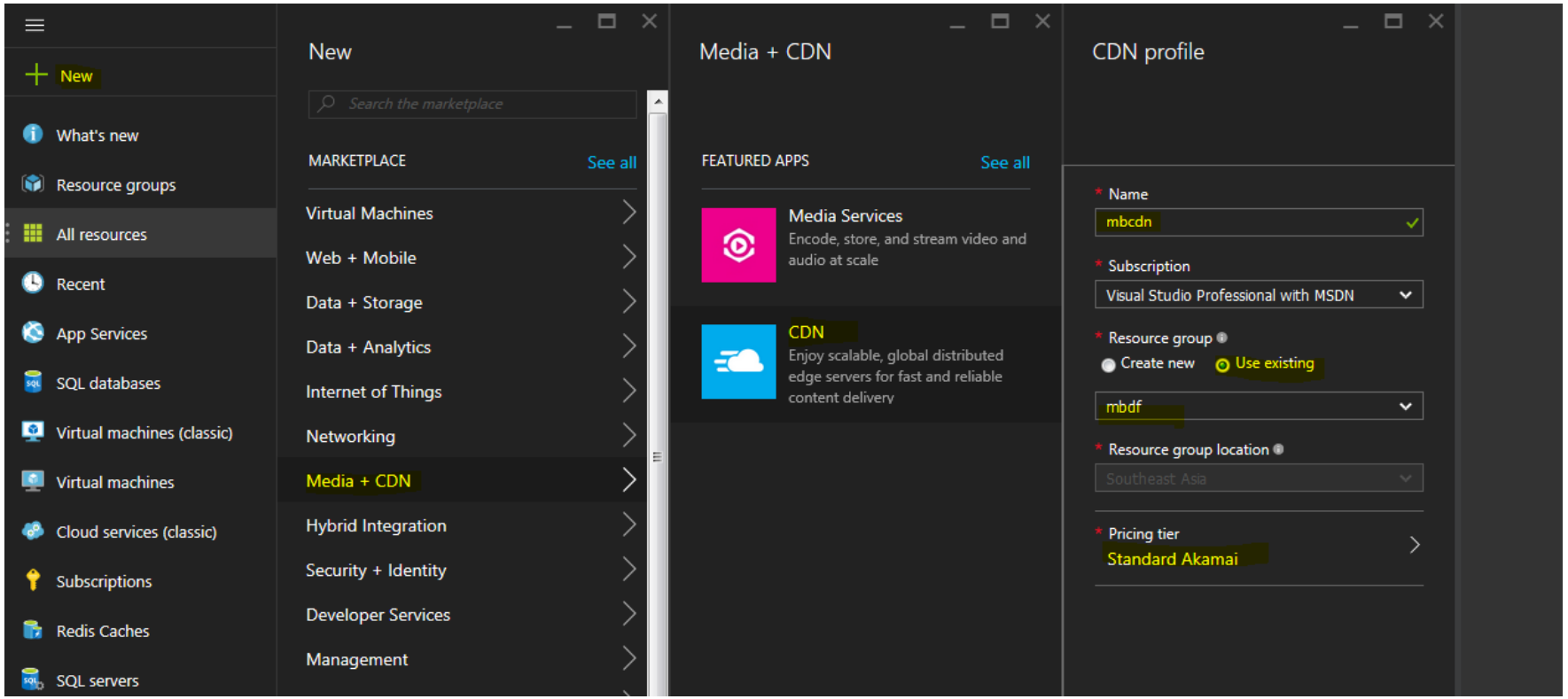

```
<div class="col-md-4">
    <h2>Web Hosting</h2>
    <p>You can easily find a web hosting company that offers the right mix of features and price for your applications.</p>
    <p><a class="btn btn-default" href="http://go.microsoft.com/fwlink/?LinkId=301867">Learn more &raquo;</a></p>
</div>
<img src="https://mbcdnstorage.blob.core.windows.net/cdn/Scality-Infographic.png" />
```

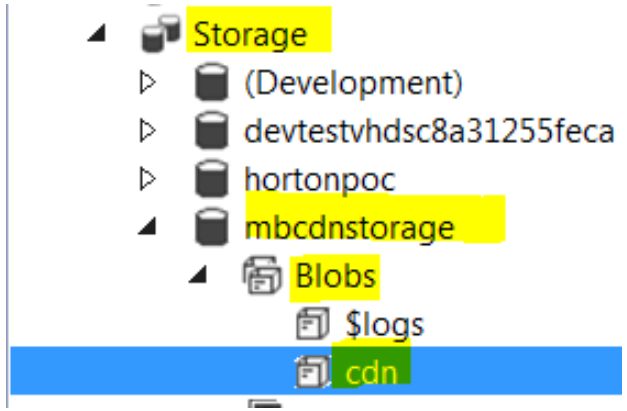




CDN

http://azure.microsoft.com/en-in/documentation/articles/cdn-serve-content-from-cdn-in-your-web-application/






```
<img src="https://az763451.vo.msecnd.net/cdn/Scality-Infographic.png"
```

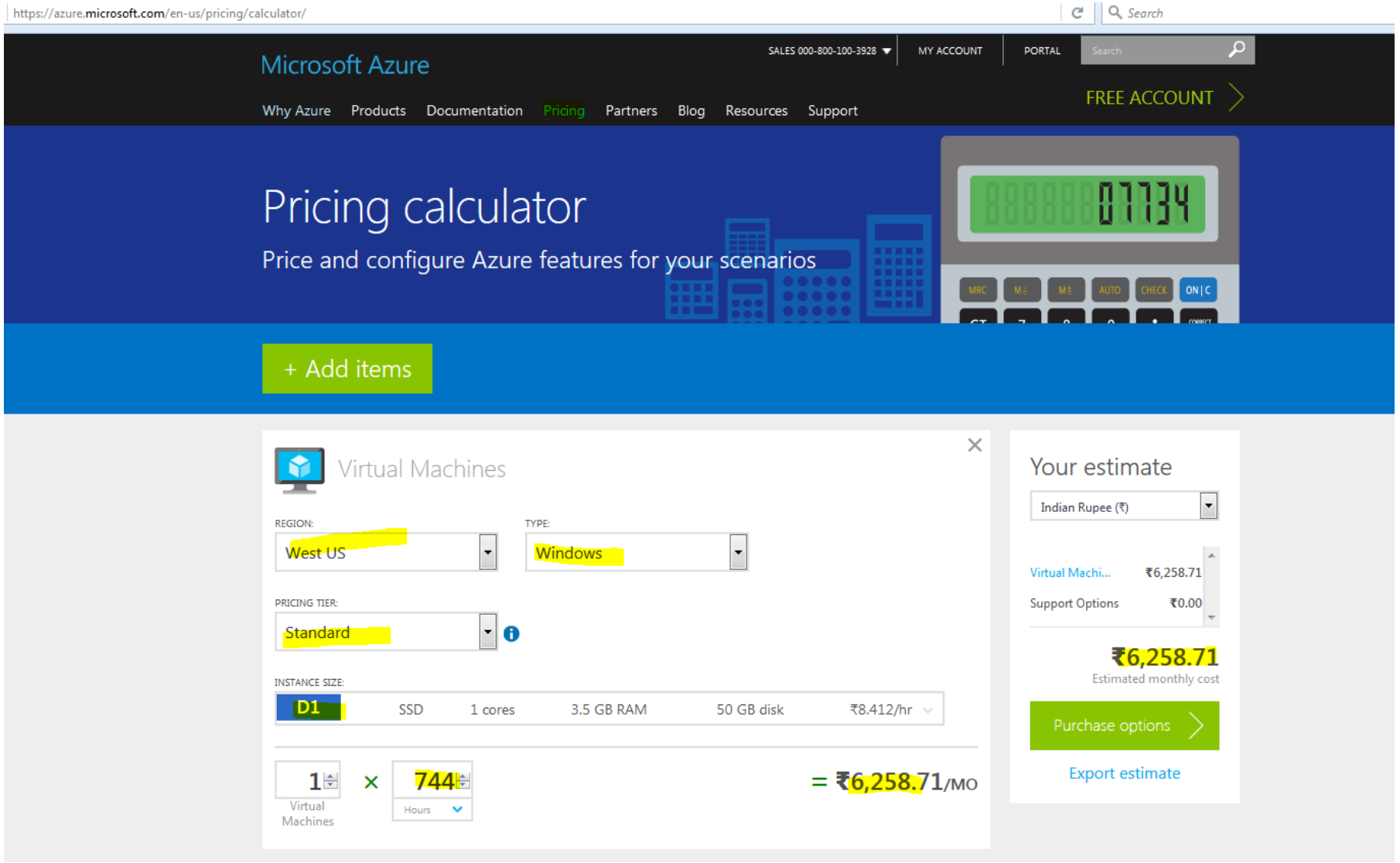
Name	Method	Status	Type	Initiator	Size	Time	Timeline
Scality-Infographic.png	GET	200	png	Index:70	679 KB	2.84 s	

How to calculate costing on Azure

Azure Calculator

Azure Calculator

https://azure.microsoft.com/en-us/pricing/calculator/




Thank you!