# Objective:

The purpose of this document is to demonstrate how to **migrate a simple on premise IIS based website to Azure Web Apps**.

# Prerequisites:

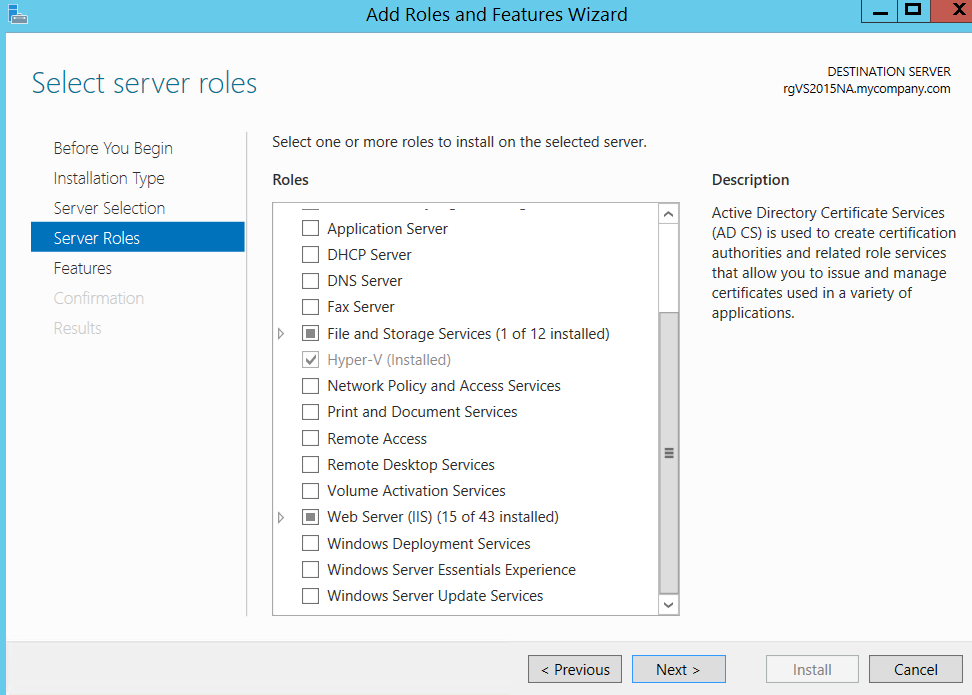
* Visual Studio 2015 with IIS
* Valid Azure Subscription
* This demonstration uses an already built ASP.NET web application named CloudShop. This app is located in the following zip **DemoApp\_CloudShop.zip**..
* This demonstration uses an already built SQL Server database named Adventure Works. You should configure this Database either on premises or in the Cloud. See docs **Demo-MigrateToAzureSqlDB** **or Demo-MigrateToAzureSqlVM** to learn how to setup the database.

# Steps Overview:

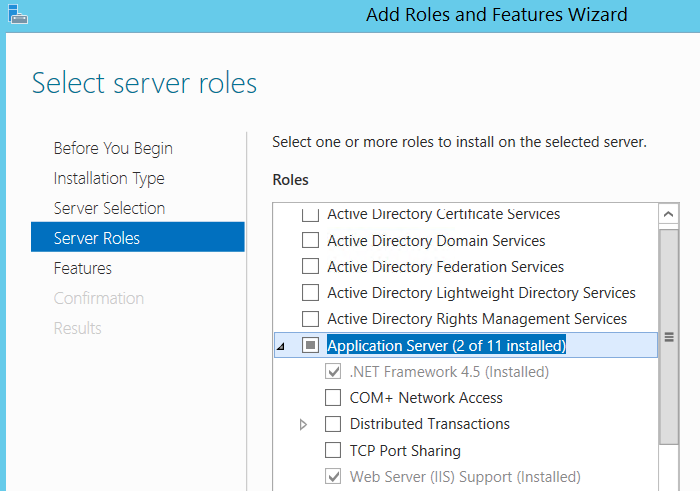
* Setting up On-Premise Website in IIS
* Connecting Visual Studio with Azure
* Create Azure Web App using Visual Studio
* Migrate website to Azure web App

# Setting up On-Premise Website in IIS

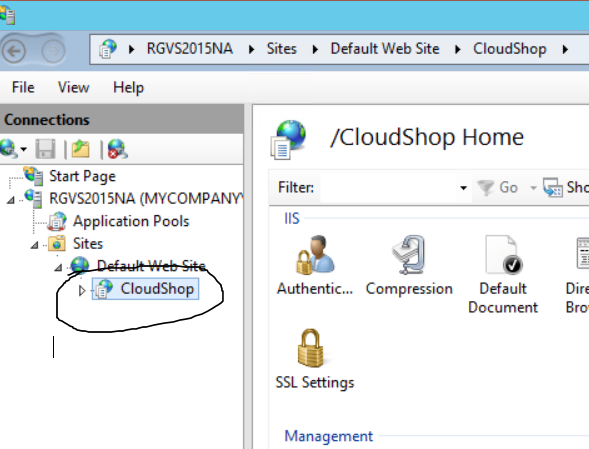
* Make sure that IIS is installed on the local server. On server 2012 R2 you can enable it as role as shown below.



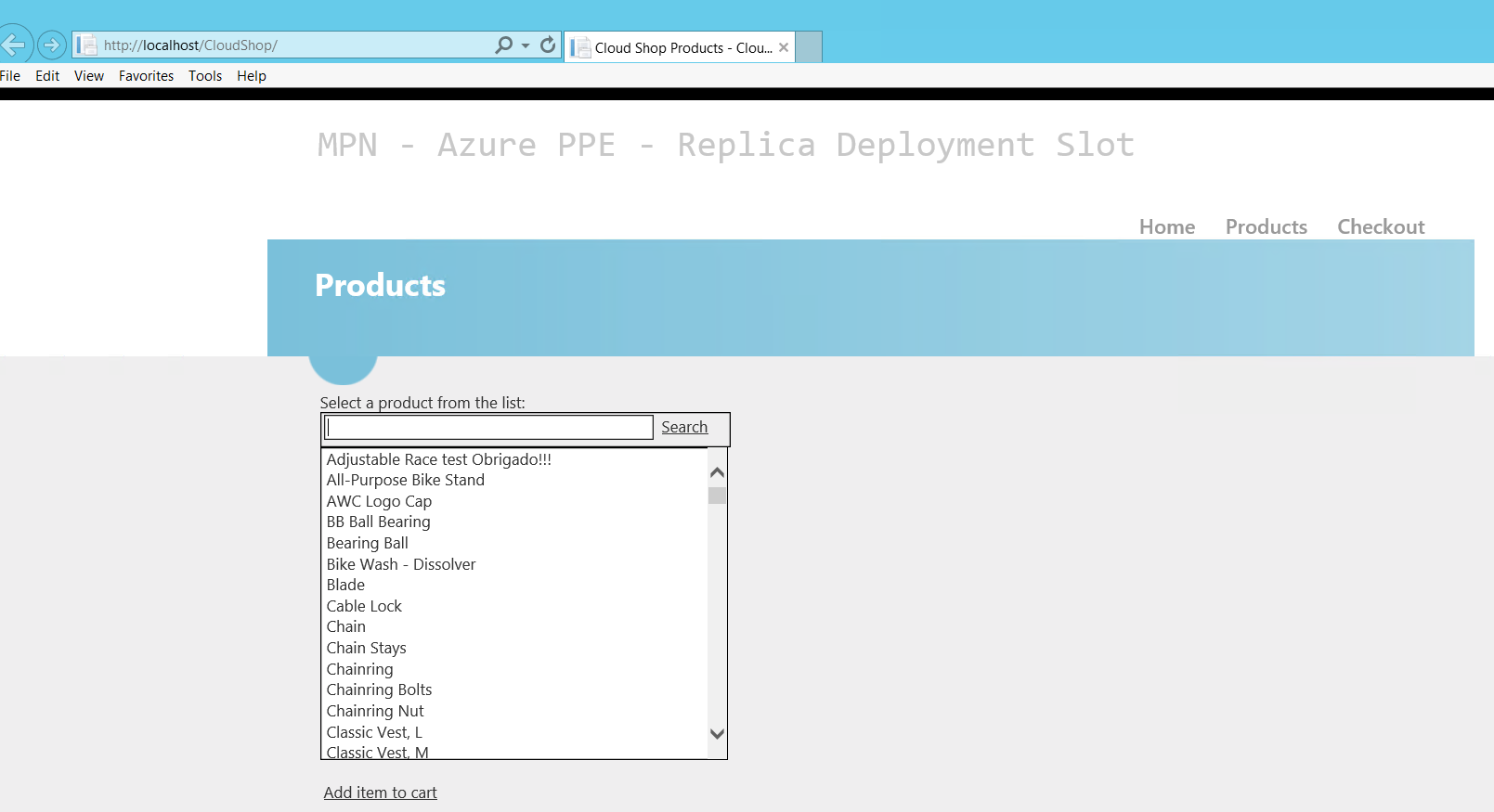
* If this is a fresh machine, installing IIS alone won’t be enough (you’ll face config errors when running). On server 2012 R2, you also need to make sure that you’ve installed “Application Server” role as well as the optional “Web Server (IIS) support” feature under Application server as shown below:



* Also make sure to give permissions to the CloudShop folder to “Everyone” and also make sure that the folder/files are NOT “Read-Only”.
* Extract CloudShop to a local folder.
* Launch VS 2015 in administrative mode and open the Solution file (CloudShopSolution.sln).
* VS 2015 will automatically setup IIS and host CloudSolution to it. You should see something like below.



* Run the application from Visual Studio and you should be able to see the app running on localhost as seen below.



* If you see an ADO.NET error, then that’s probably because the project uses AdventureWorks Databases hosted at sw7heo19qc.database.windows.net and uses connection string mentioned in web.config a sshown below.



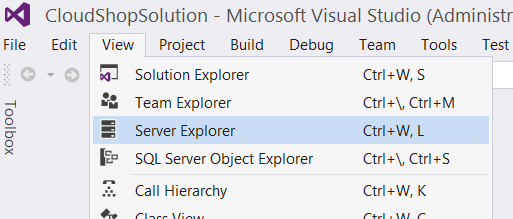
If for some reason, this database is unavailable, please host [AdventureWorks](http://msftdbprodsamples.codeplex.com/releases/view/37304) Database in your own publicly accessible SQL Database and replace the connection string.

* Congratulations! CloudShop is now setup to run on an IIS VM.

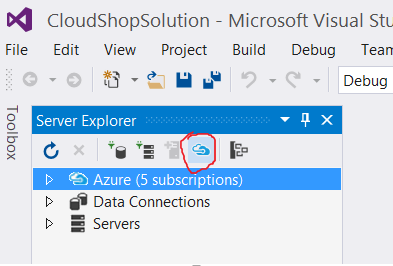
# Connecting Visual Studio with Azure

Before we could migrate on premise CloudShop to Azure Web App, we need to connect Visual Studio to Microsoft Azure. Below steps demonstrate how to do so.

* Open “Server Explorer” in VS 2015 as shown below.



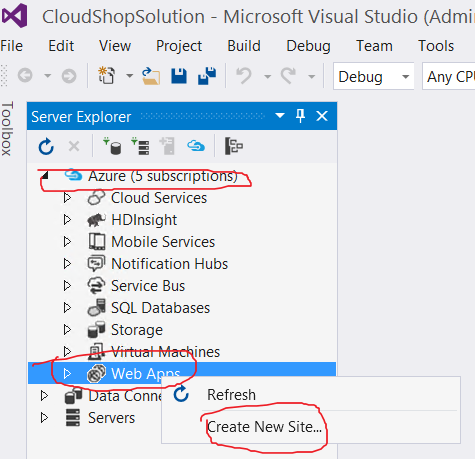
* From Server Explorer, click on icon to connect to Microsoft Azure Subscription as shown below.



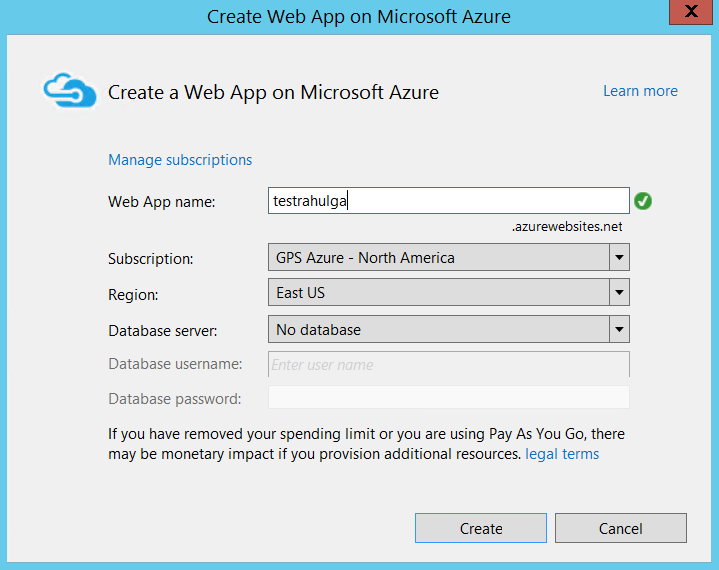
* Follow the wizard and input your credentials for your Azure Subscription.
* Congratulations! Your Visual Studio is now connected to your Azure Subscription.

# Create Azure Web App using Visual Studio

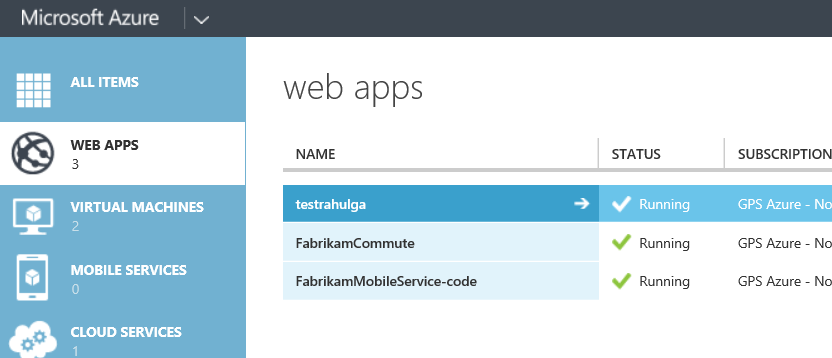
* By now, you must be able to see all resources under your Azure Subscriptions. Navigate to Azure Web Apps, right click and choose the option to “Create New Site…” as shown below:



* On the “Create We App…” dialog, key in the details like a valid, available DNS name for your web app, subscription, region etc. and click “Create” as shown below. Note that I have not selected any database at present. This is because a database is not needed by CloudShop application. We are using in memory session to store items and not any persistent data store.

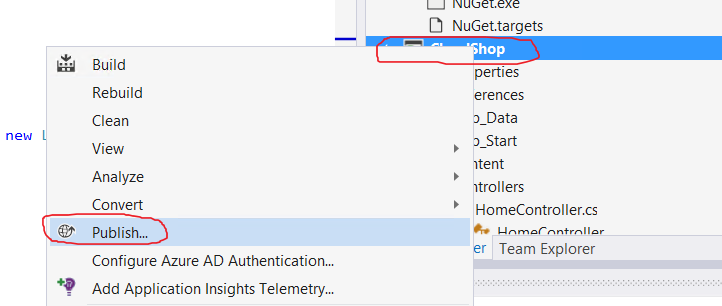


* Verifying From Portal: Once Web App is created, right click your web app and choose “Open in Management Portal”. You should be able to see the same web app that you just created from VS, on Azure Portal too as shown below:

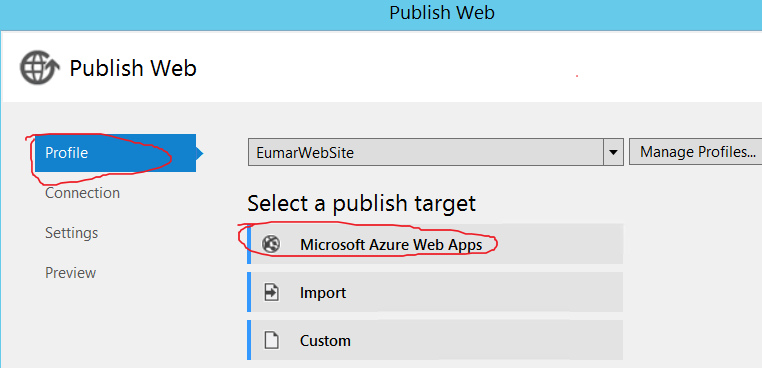


# Migrate website to Azure web App

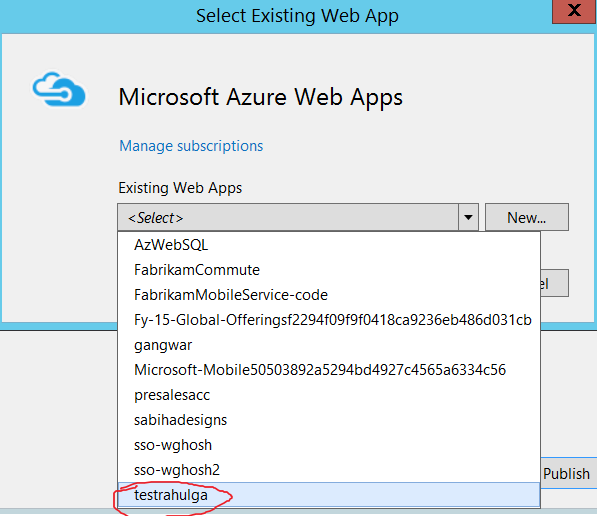
* From Solution Explorer, right click your project and select "Publish" as shown below:



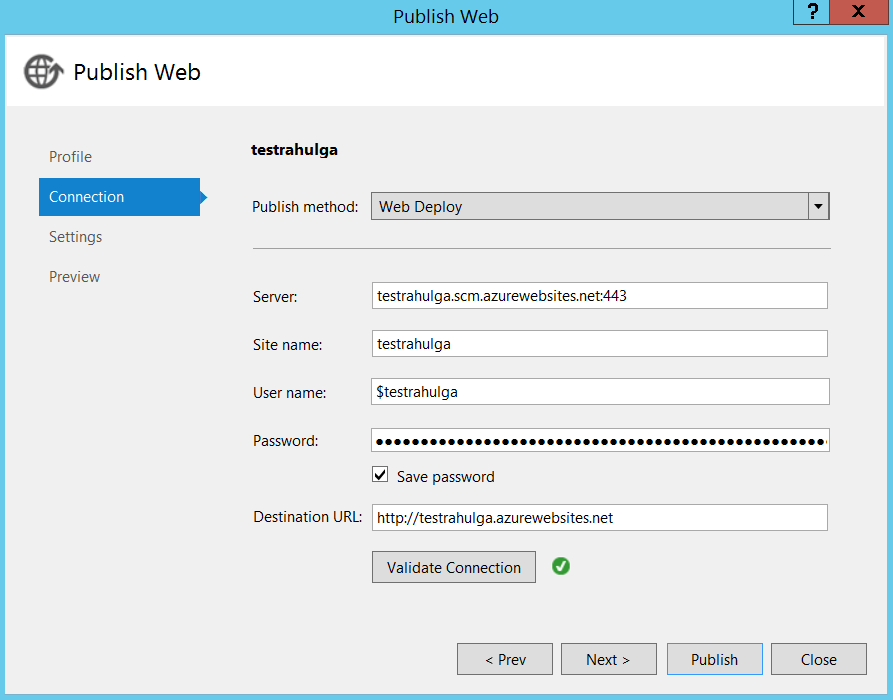
* On “Publish Web” dialog, select “Profile” from left menu, choose “Microsoft Azure Web Apps” from options as shown below.



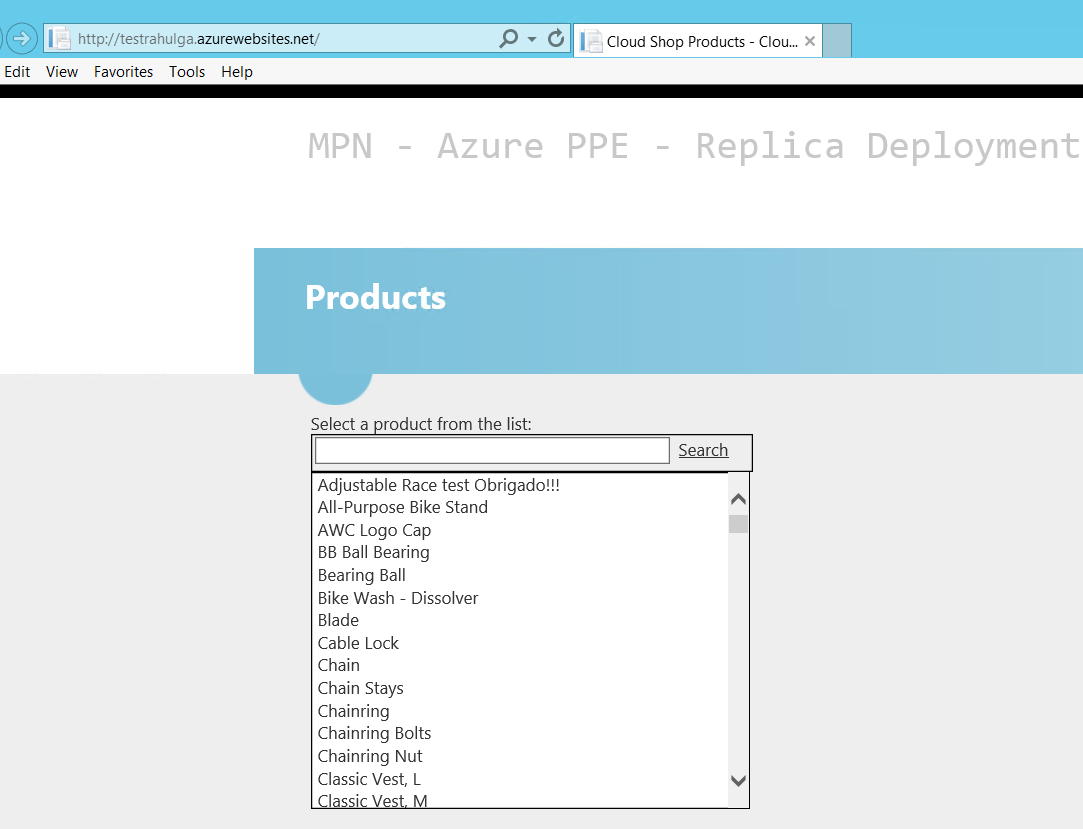
* On the subsequent dialog, select the web app that you created earlier ash shown below.  
  Note that we could have created web app at this scree also instead of doing that as separate step but then the clarity of steps would have lost.



* You must finally see all the settings filled by VS on the publish dialog as shown below. You can also, optionally, validate connection at this stage. Once done, click on “Publish”.

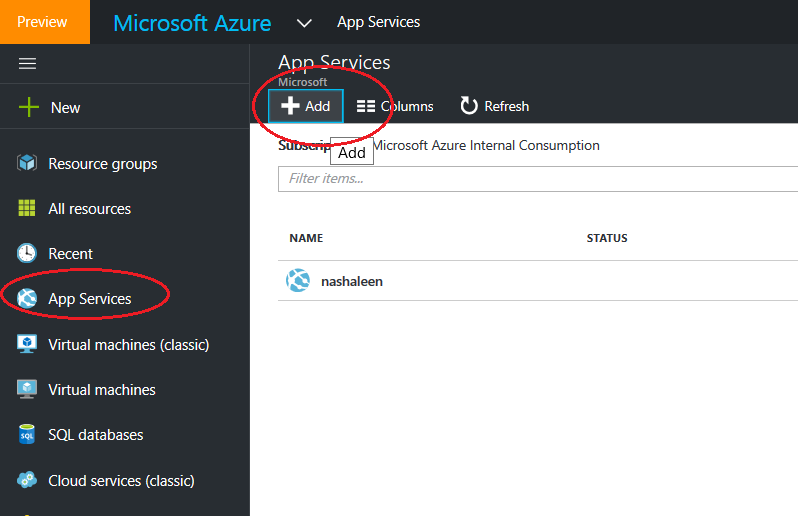


Congratulations! You have successfully published your IIS website to Azure Web App. You can now navigate to it publicly as shown below. Actually VS will do that you OOB.

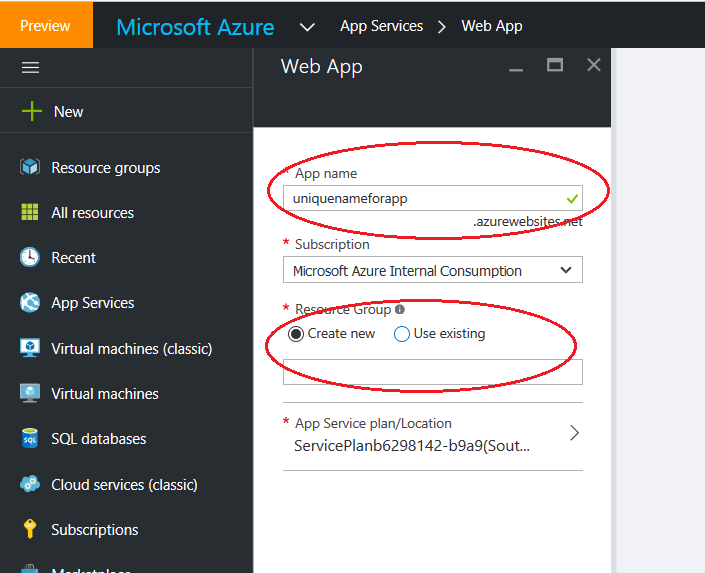


Please Note: For Web Apps in the ARM/New Portal:

* Login to <http://portal.azure.com>
* Use the same credentials as you use for <http://manage.windowsazure.com>
* Click on “More Services” or if you see “App Services” click on it. In the search window, type App Services. Once it comes, click on “+Add”



Next add the following:



Rest of the options remain same.

Lab Completed!