

**Module 1 Lab: Azure Intro & Azure SQL**

Template Version: 2.0

**Estimated Time**

30 minutes

**Objectives**

At the end of this lab, you will be able to:

* Deploy an App Service – Web App and API App
* Deploy your local application to App Services manually using Visual Studio

**Logon Information**

Please use the Azure Pass provided to you for this lab. Your VM credentials are listed for later use:

* VM Username: super
* VM Password: P@ssw0rd123!

Table of Contents

[Module 2 Lab: Azure App Services 3](#_Toc522496050)

[Exercise 1: Deploy the SQL Schema 3](#_Toc522496051)

[Exercise 2: Deploy the Web API to Azure 6](#_Toc522496052)

[Exercise 3: Deploy the Web App to Azure 10](#_Toc522496053)

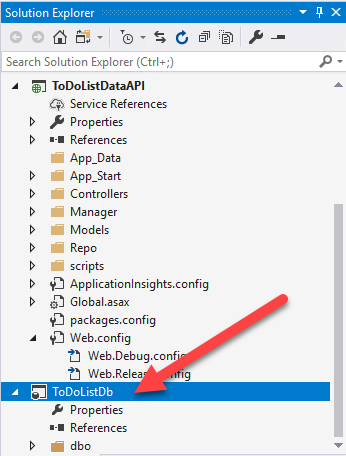
Module 2 Lab: Azure App Services

Exercise 1: Deploy the SQL Schema

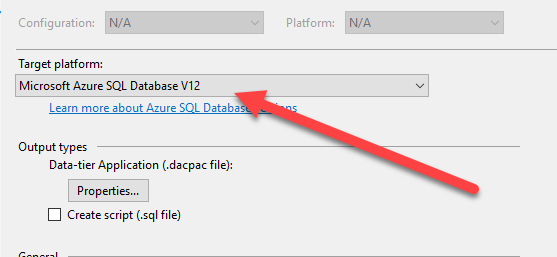
This exercise shows how to use Azure App Services.

Tasks

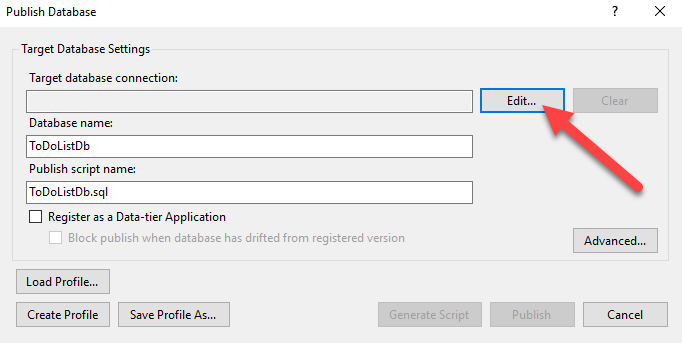
1. **Deploy the Database schema to Azure SQL DB**
2. RDP into your VM, if you are not already connected to the VM from the previous lab. If you need help with this, refer to the Setup instructions in Module 0 or let your instructor know and they can assist. The login information for the VM is:
   1. Username: super
   2. Password: P@ssw0rd123!
3. Open Visual Studio 2017. This is a Community version and will remain active if any email account is signed into it.
4. You will be prompted to Sign in. Login using your Outlook account associated with your Azure Pass.
5. Open the .sln file from C:\labs\day1 in Visual Studio
6. Go to the Solution Explorer, find the ToDoListDb project, right click it, and select Properties.



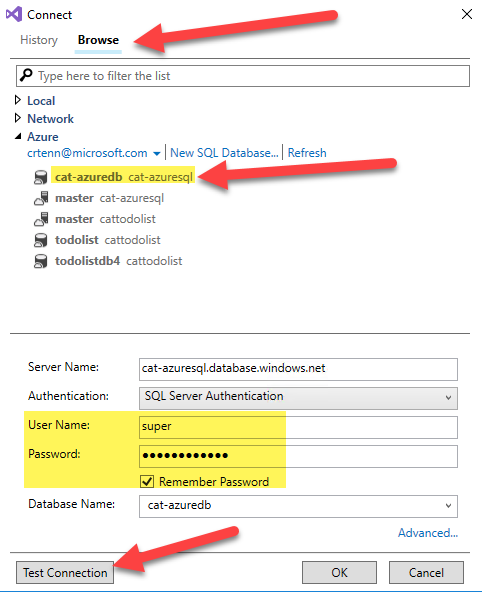
1. Change the Target platform to: **Microsoft Azure SQL Database V12** which is for Azure SQL.



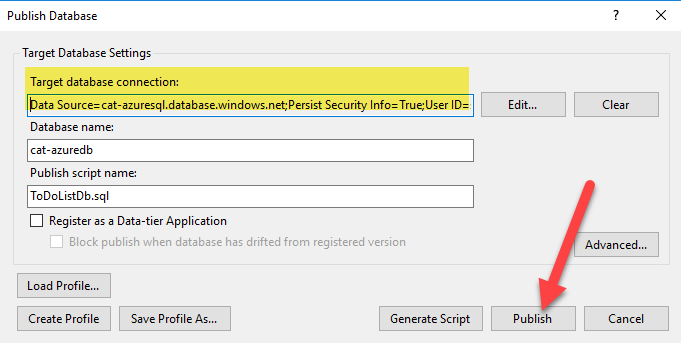
1. Go to the Solution Explorer, find the ToDoListDb project, right click it, and select Publish. Wait a minute for a modal.
2. In the pop-up modal, click Edit.



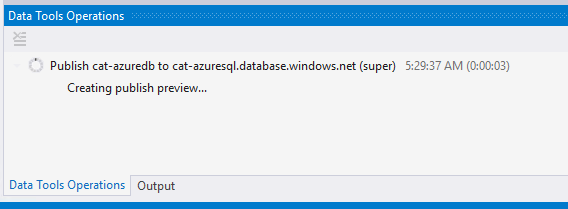
1. Click Browse > Azure > initials-azuredb. The username should be “super”, and the password should be “P@ssw0rd123!”. Click Test Connection. It should succeed. Press OK on the successful test modal pop-up.



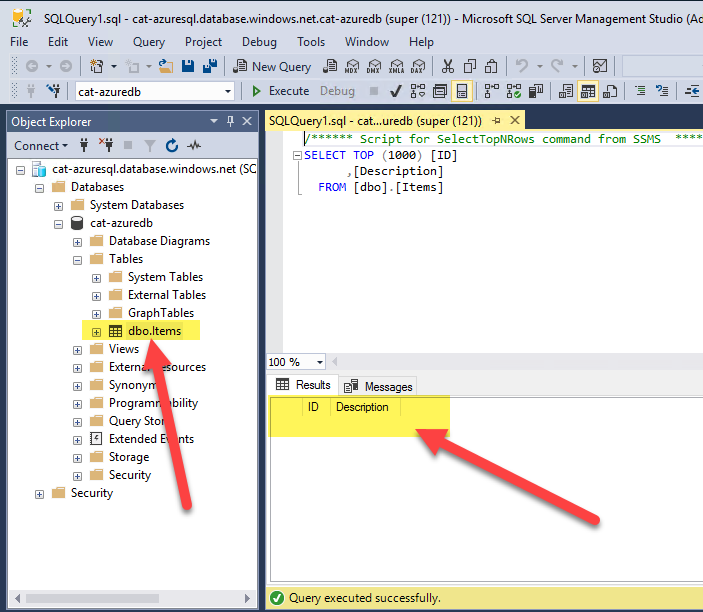
1. Press OK.
2. Your target database connection should now be populated. Press Publish.



1. On the Output window on the bottom of Visual Studio, you should see the Publish starting.



1. Please wait 5 minutes for the Publish to complete.
2. Check your SQL Server Management Studio which should still be connected to your Azure SQL Server. Refresh your connection. You should see that you have a table now called dbo.Items. Run a select to verify the columns.

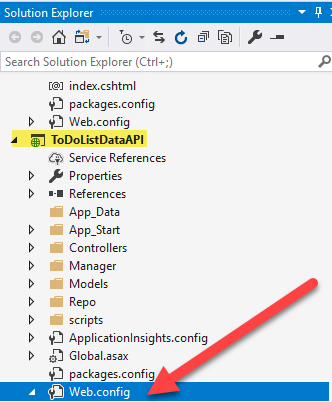


Exercise 1 has been completed

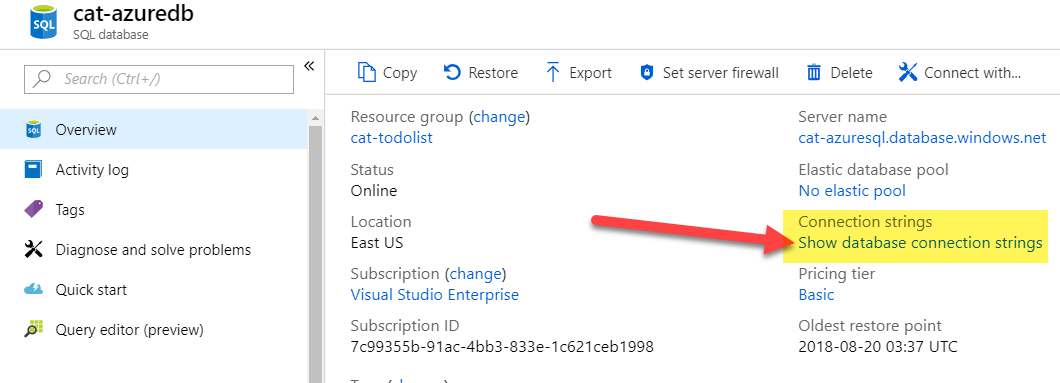
Exercise 2: Deploy the Web API to Azure

Tasks

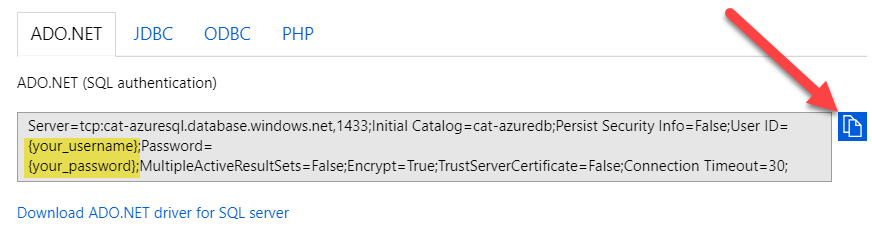
1. Connect the API App to the Azure SQL DB in Azure
2. Go to the Solution Explorer > ToDoListDataAPI > Web.config



1. Under the ConnectionStrings section, comment out line 9 and comment in line 15.
2. Go to the Azure Portal and find your SQL Database resource. Click **Show database connection strings.**



1. Copy the ADO.NET connection string.



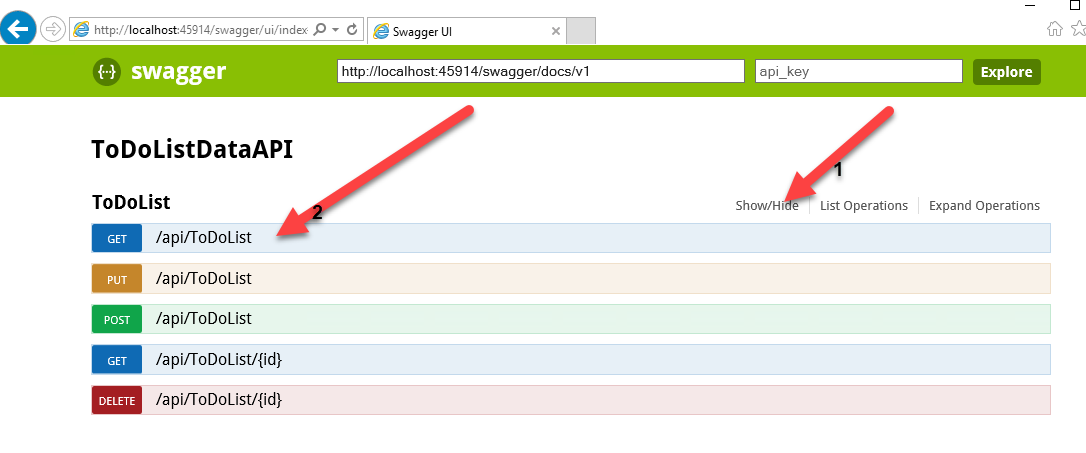
1. Notice that the username and password in the screenshot above need to be filled in. Paste the Connection String into your API layer web.config onto line 15.
2. Fill in the username and password with: super and P@ssw0rd123!

Web.config before:

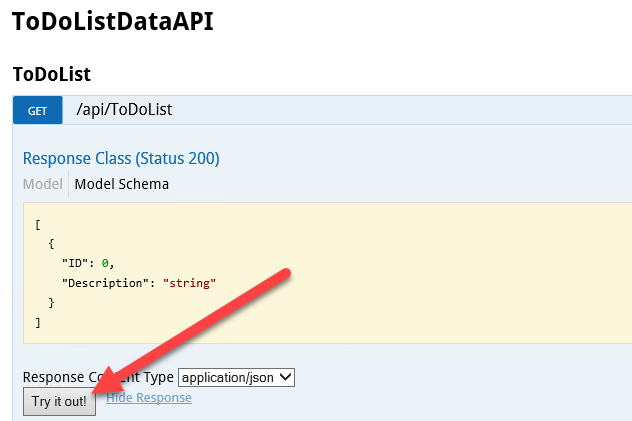


Web.config after:

1. Set the ToDoListDataAPI project as the Startup project.
2. Hit F5.
3. Add /swagger to the URL so it is: <http://localhost:45914/swagger/>
4. Click Show/Hide. Click GET.



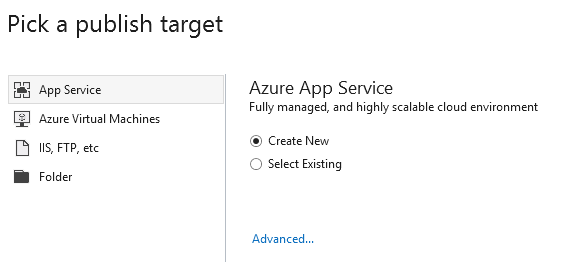
1. Click Try It out.



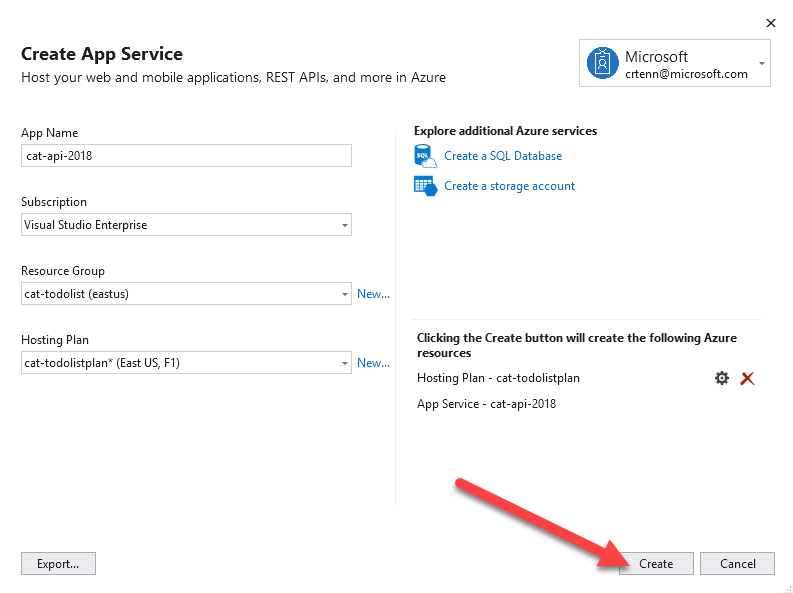
1. You should get a 200 Response Code if you are connected to the SQL DB successfully.



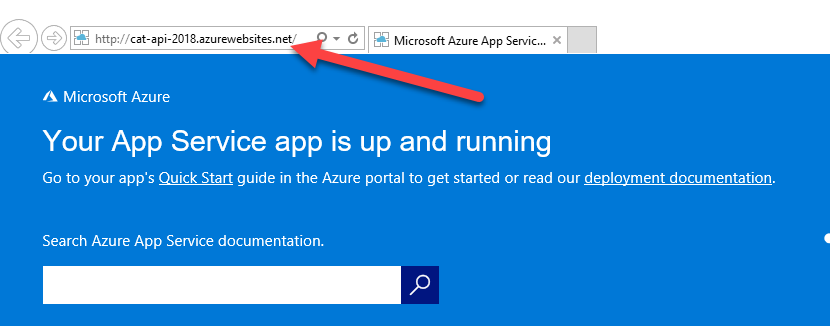
1. Deploy the Web App to Azure
2. In the Solution Explorer, find the ToDoListDataAPI project, right click, and select Publish.
3. Select “Create New” under Azure App Service



1. Click the Publish button on the bottom.
2. Set the following:
   1. Set the App Name: initials-api-2018
   2. Subscription: Azure Pass
   3. Resource Group: initials-todolist (eastus)
   4. Hosting Plan:
      1. Click New.
      2. App Service Plan: initials-todolistplan
      3. Location: East US
      4. Size: Free
   5. Your screen should look as follows, but with your initials:



* 1. Click Create.
  2. Please wait 2 minutes, or until the browser pops up on its own with your App Service. Once it does, please copy the URL out of the browser.

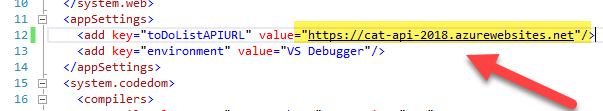


Exercise 2 has been completed

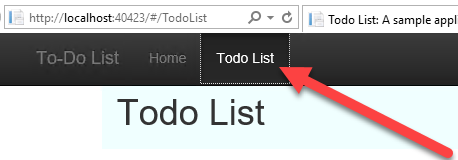
Exercise 3: Deploy the Web App to Azure

Tasks

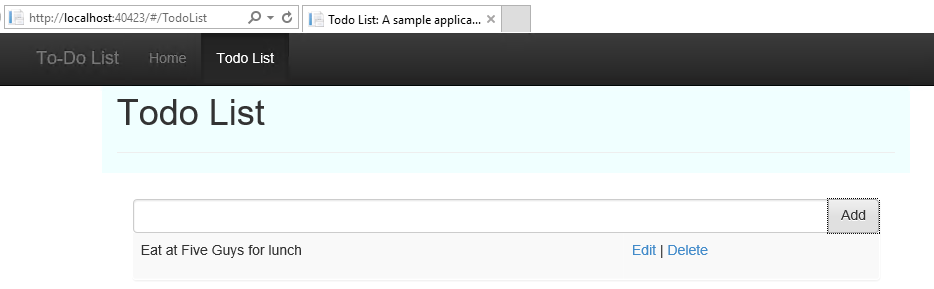
1. Connect the Web App to the API App in Azure
2. In Solution Explorer, under the ToDoListAngular project, go to the web.config file.
3. On line 12, please change the **value** to the URL you copied from your browser. However, please change **http** to **https**. And remove the **slash** at the end. Please double check that you have done this correctly.



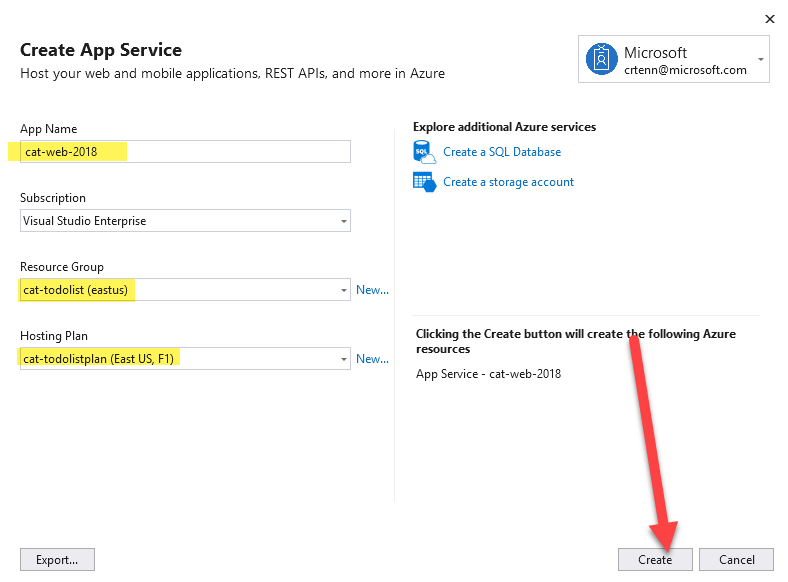
1. Now we need to test our connectionSet the Startup project to the ToDoListAngular project.
2. Press F5.
3. Click Todo List.



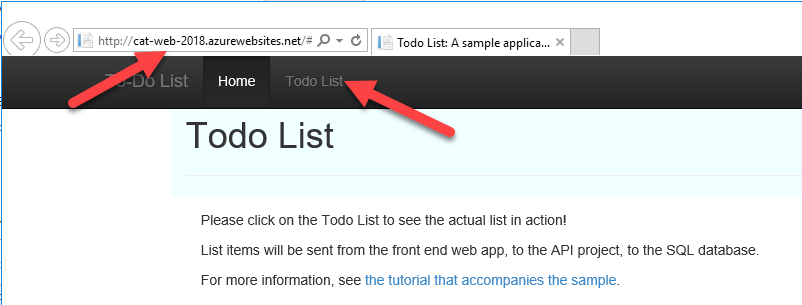
1. Try adding entries, editing, and deleting them. Refresh the page and make sure that they persist and check the database table for the entries.



1. Deploy the Web App to Azure
2. Right click on the ToDoListAngular project and select Publish.
3. Select Create New under App Service.
4. Hit Publish
5. Fill out the Create form as follows:
   1. Set the App Name: initials-web-2018
   2. Subscription: Azure Pass
   3. Resource Group: initials-todolist (eastus)
   4. Hosting Plan: initials-todolistplan (already created from previous exercise)



1. Click Create. Wait for 2 mins / until a browser pops up with your Web App front end. Notice the URL is now a cloud URL. Click Todo List again and test out the app. It should have retained any saved data from your local session since it is connected to the same SQL Database in the cloud.



Exercise 3 has been completed. Congrats you just deployed your first full web app into Azure!