

Name: AzureAppServiceDeploymentSOP

Date: 2-7-2026

Purpose: Step by step instructions to deploy a web service into Azure using the Azure CLI and the Azure portal

Version: 1.0

Notes: I used my own Azure environment on Azures free tier pricing plan for this project.

Assumptions:

- Your Azure Tenant has multi-factor authentication set up.
- Your Tenant has been given the appropriate permission to deploy an Azure App Service.
- Azure Cloud Shell is installed on your device.
- You have an Azure Resource group already created (See AzureResourceGroupCreationSOP).
- You are using Chrome as your browser.
- This is a simple, manual deployment. No containers and settings are kept simple.
- Your application uses .NET 8.0

A. How to log into Azure from PowerShell

1. Type: "az login --use-device-code
 - I. PowerShell will provide you a code and a browser link.
 2. Press ctrl while you click on the link and a browser window will appear.
 - I. The browser window will show Microsoft's logo and will say "enter code to allow access"
 3. Enter the code provided to you in the previous step.
 - I. If successful you will see a table called Tenant and subscription selection as we as a prompt to select a subscription and tenant.
 4. Enter the Number from the No column associated with the subscription you wish to use.
 - I. If successful you will see a text output for the Tenant you selected and the Subscription
 5. Verify a successful login by typing "az account list --output table"
 6. A table will appear for the account list. Check the state column and make sure it says Enabled.
- B. How to create an Azure App Service Plan from Azures portal

1. Using your browser log into Azure at <https://portal.azure.com/auth/login/>.
 2. Towards the top right-hand side click Create.
 3. Select the subscription you wish to use.
 4. Select the resource group you wish to place the app in. If you don't have a resource group see the repo [AzureResourceGroupCreationSOP](#).
 5. Under Name type the a name for your web service Plan `aspnet-default-weather-api-plan`.
 6. Under operating system select Linux.
 7. Under region select the region that's closest to you. (Reduces latency)
 8. Under pricing plan select the pricing plan you want.
 - I. I selected the free F1 (Shared infrastructure)
 9. Select Review + Create.
 10. Select Create.
- C. How to create an Azure App Service from Azures portal
1. In the search bar type App Services.
 2. Select the App Services from the drop down under the services section.
 3. Towards the top right-hand side click Create.
 4. Select Web App.
 5. Select the subscription you wish to use.
 6. Select the resource group you wish to place the app in. If you don't have a resource group see the repo [AzureResourceGroupCreationSOP](#).
 - I. It is recommended that you use an Azure Resource Group as it enables you to organize, govern and manage your resources.
 - a. Role Based Access (RBAC)
 - b. Cost tracking
 - c. Efficient resource deletion.
 - i. Allows you to delete entire projects and all resources tied to that project at once reducing time and cost.
 7. Under Name type the name for your Web App Service.
 - I. I like to use one similar to the name of my application with a few additional identifiers to help me know what I am looking at later without going into the application. EX: APIs name -> WeatherApiLab App Services name -> `aspnet-api-lab-default-weather-api`.
 8. Under publish select code.
 9. Under runtime stack select .NET 8.
 10. Under OS select Linux.
 11. Select the region closes to you under region.
 12. Under the Linux plan select the App Service Plan we created earlier

13. Under zone redundancy select disabled.
 14. Click Next: Database.
 15. Click Next: Deployment.
 16. Click Next: Networking.
 17. Click Review + Create
- D. Publish, zip and deploy your app.
1. Type: cd [your apps file path].
 2. Type: dotnet publish -c Release -o ./publish
 3. Type: Compress-Archive -path publish* -DestinationPath app.zip -Force
 4. Type: az webapp deploy --resource-group [Your Azure Resource Group name] --name [Your Azure App Service name]
 - I. If successful you will get a Json string and a success message.
- E. Test your deployment.
1. Go to your Azure portal.
 2. Search App Services.
 3. Select App Services.
 4. Select your App Service.
 5. Copy your apps default domain.
 6. Paste it in a new tabs URL.
 - I. If you deployed an app you should see it.
 - II. If you deployed an API or micro service and didn't set a value for the main page go back to your app's program.cs file. Underneath the code block
`if(app.Environment.IsDevelopment){ ... } add app.MapGet("/", () => "application is running...");` then republish and redeploy.