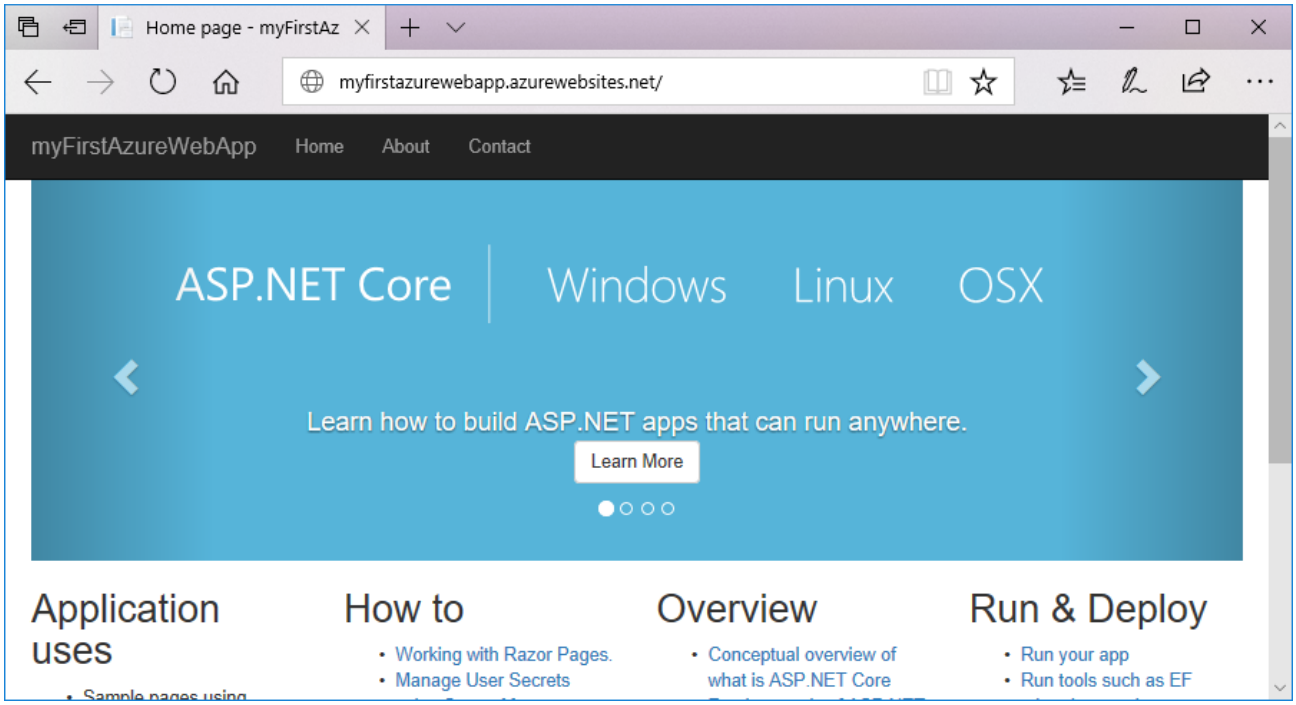
Create an ASP.NET Core web app in Azure

09/05/2018 • 4 minutes to read • Contributors 

In this article

> **Note**
>
> This article deploys an app to App Service on Windows. To deploy to App Service on *Linux*, see **Create a .NET Core web app in App Service on Linux**.

Azure App Service provides a highly scalable, self-patching web hosting service. This quickstart shows how to deploy your first ASP.NET Core web app to Azure App Service. When you're finished, you'll have a resource group that consists of an App Service plan and an App Service app with a deployed web application.



If you don't have an Azure subscription, create a free account before you begin.

Prerequisites

To complete this tutorial, install Visual Studio 2017 with the **ASP.NET and web development** workload.

If you've installed Visual Studio 2017 already:

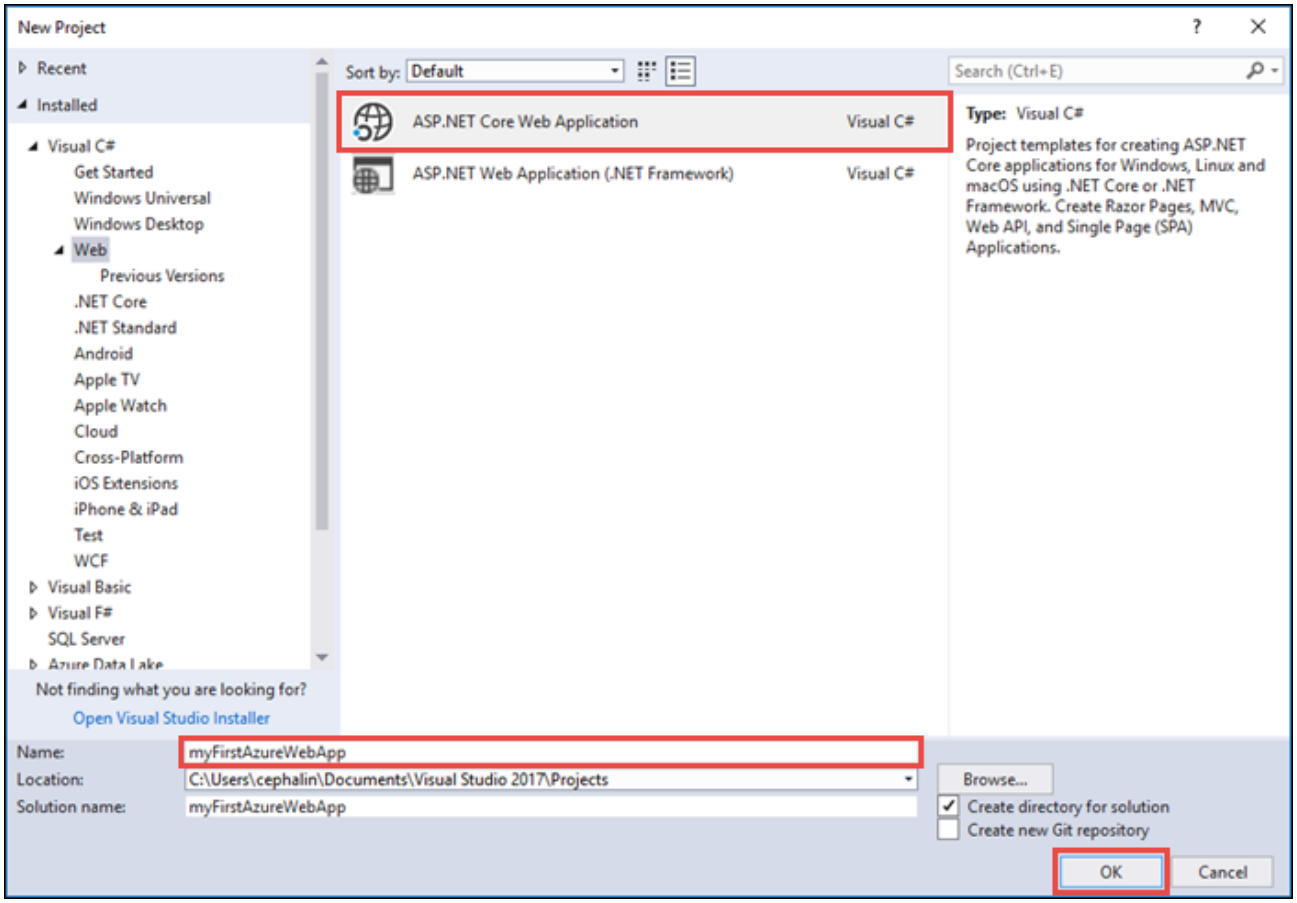
- Install the latest updates in Visual Studio by clicking **Help** > **Check for Updates**.
- Add the workload by clicking **Tools** > **Get Tools and Features**.

Create an ASP.NET Core web app

In Visual Studio, create a project by selecting **File** > **New** > **Project**.

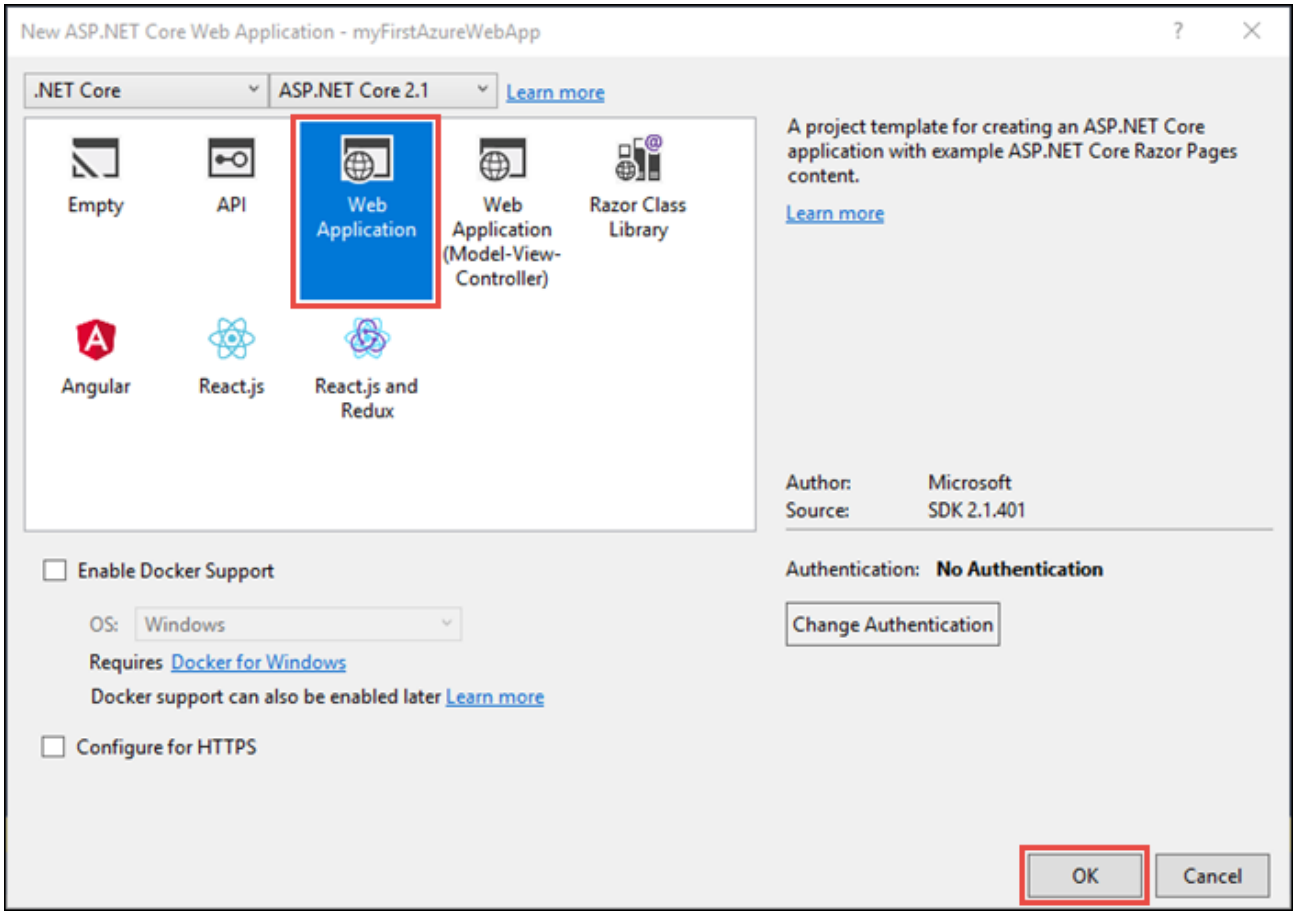
In the **New Project** dialog, select **Visual C#** > **Web** > **ASP.NET Core Web Application**.

Name the application *myFirstAzureWebApp*, and then select **OK**.

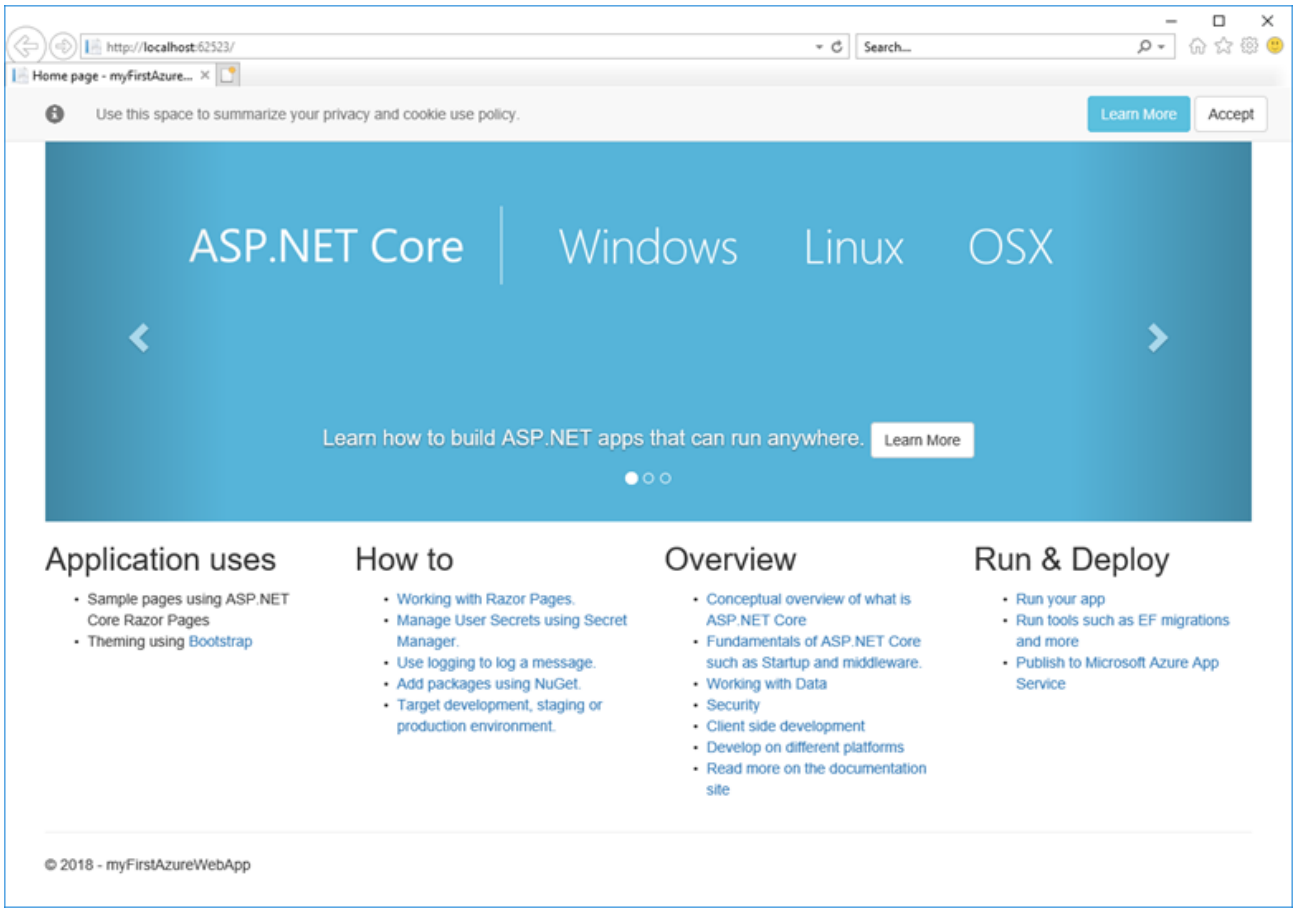


You can deploy any type of ASP.NET Core web app to Azure. For this quickstart, select the **Web Application** template, and make sure authentication is set to **No Authentication** and no other option is selected.

Select **OK**.

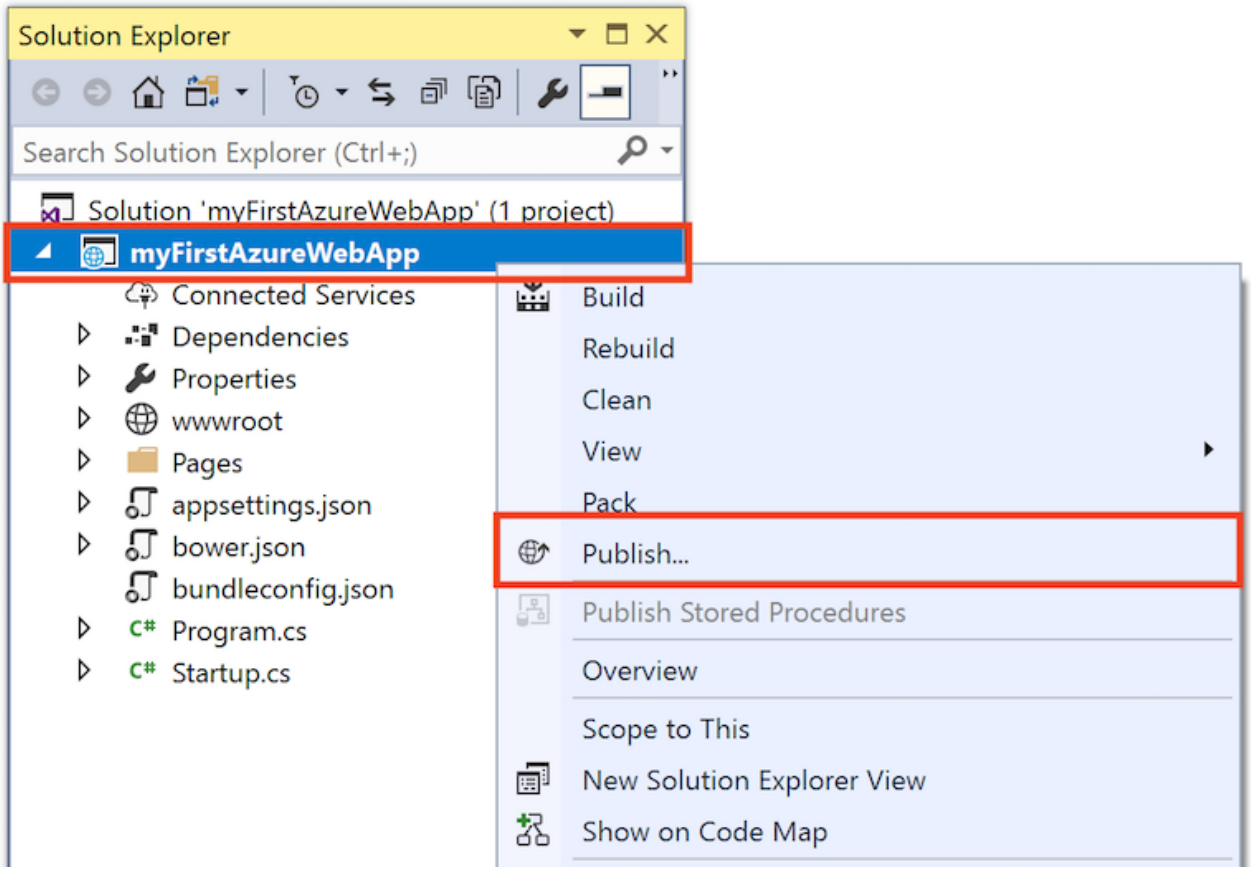


From the menu, select **Debug** > **Start without Debugging** to run the web app locally.

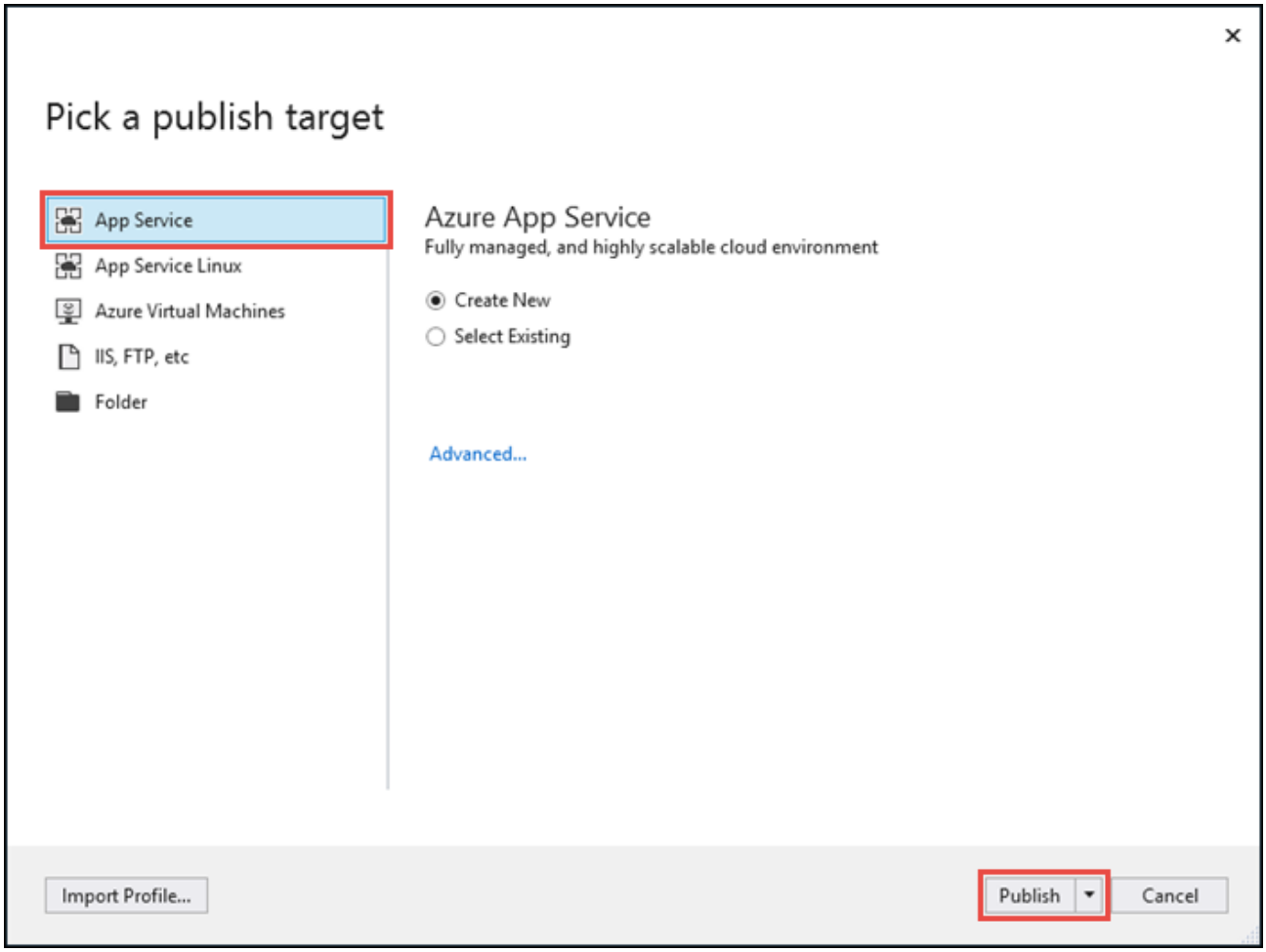


Launch the publish wizard

In the **Solution Explorer**, right-click the **myFirstAzureWebApp** project and select **Publish**.



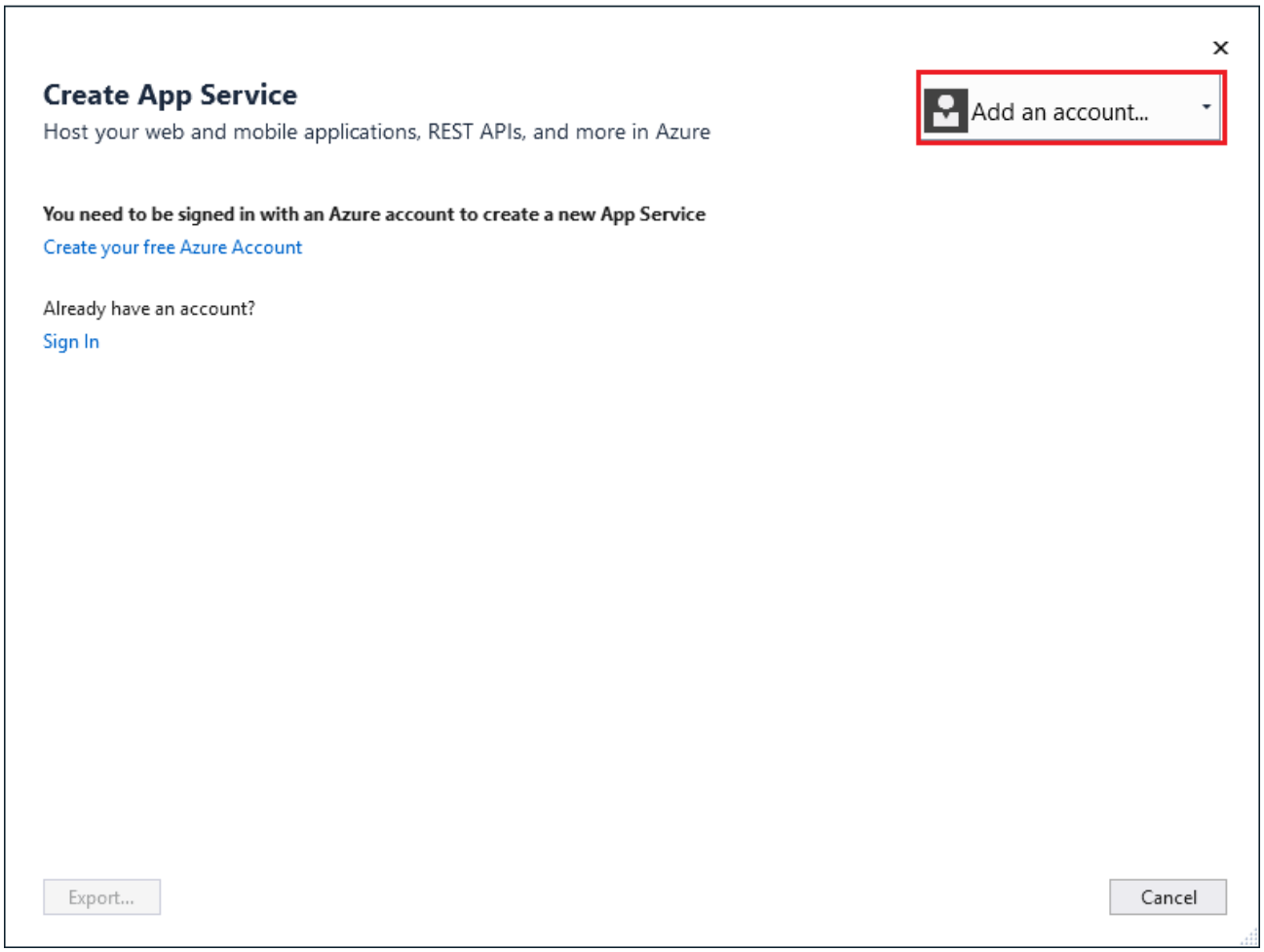
The publish wizard is automatically launched. Select **App Service** > **Publish** to open the **Create App Service** dialog.



Sign in to Azure

In the **Create App Service** dialog, click **Add an account**, and sign in to your Azure subscription. If you're already signed in, select the account you want from the dropdown.

> **Note**
>
> If you're already signed in, don't select **Create** yet.



Create a resource group

A resource group is a logical container into which Azure resources like web apps, databases, and storage accounts are deployed and managed. For example, you can choose to delete the entire resource group in one simple step later.

Next to **Resource Group**, select **New**.

Name the resource group **myResourceGroup** and select **OK**.

Create an App Service plan

An App Service plan specifies the location, size, and features of the web server farm that hosts your app. You can save money when hosting multiple apps by configuring the web apps to share a single App Service plan.

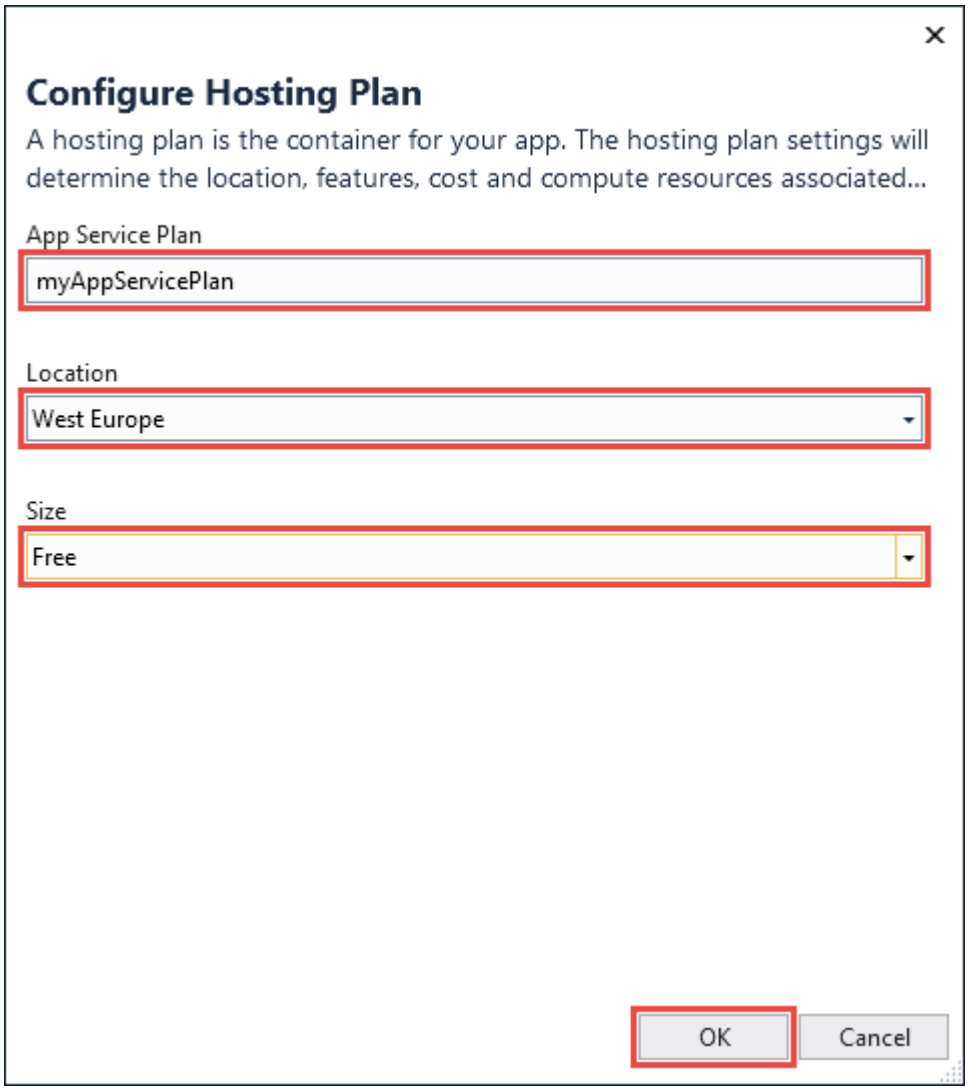
App Service plans define:

- Region (for example: North Europe, East US, or Southeast Asia)

- Instance size (small, medium, or large)
- Scale count (1 to 20 instances)
- SKU (Free, Shared, Basic, Standard, or Premium)

Next to **Hosting Plan**, select **New**.

In the **Configure Hosting Plan** dialog, use the settings in the table following the screenshot.



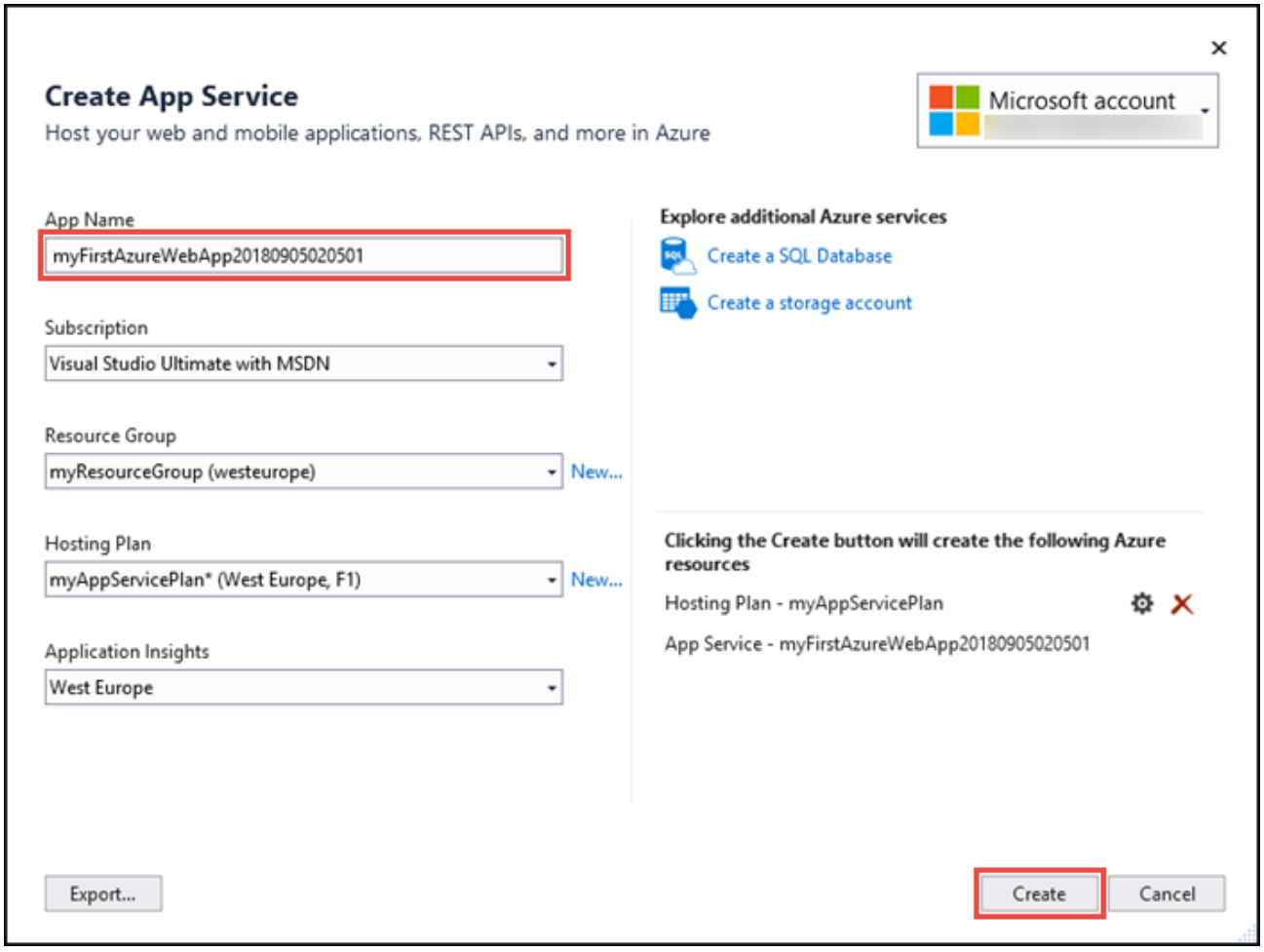
Setting	Suggested Value	Description
App Service Plan	myAppServicePlan	Name of the App Service plan.
Location	West Europe	The datacenter where the web app is hosted.
Size	Free	Pricing tier determines hosting features.

Select **OK**.

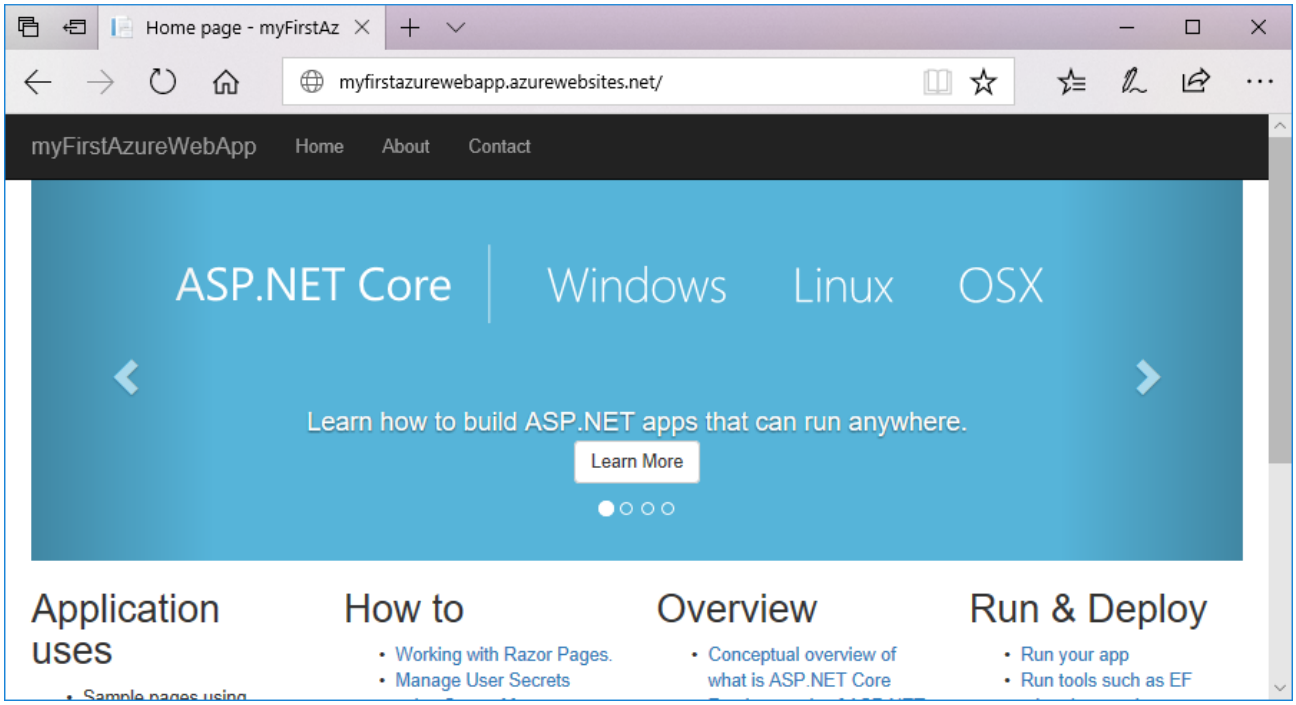
Create and publish the web app

In **App Name**, type a unique app name (valid characters are `a-z`, `0-9`, and `-`), or accept the automatically generated unique name. The URL of the web app is `http://<app_name>.azurewebsites.net`, where `<app_name>` is your app name.

Select **Create** to start creating the Azure resources.



Once the wizard completes, it publishes the ASP.NET Core web app to Azure, and then launches the app in the default browser.



The app name specified in the create and publish step is used as the URL prefix in the format `http://<app_name>.azurewebsites.net`.

Congratulations, your ASP.NET Core web app is running live in Azure App Service.

Update the app and redeploy

From the **Solution Explorer**, open *Pages/Index.cshtml*.

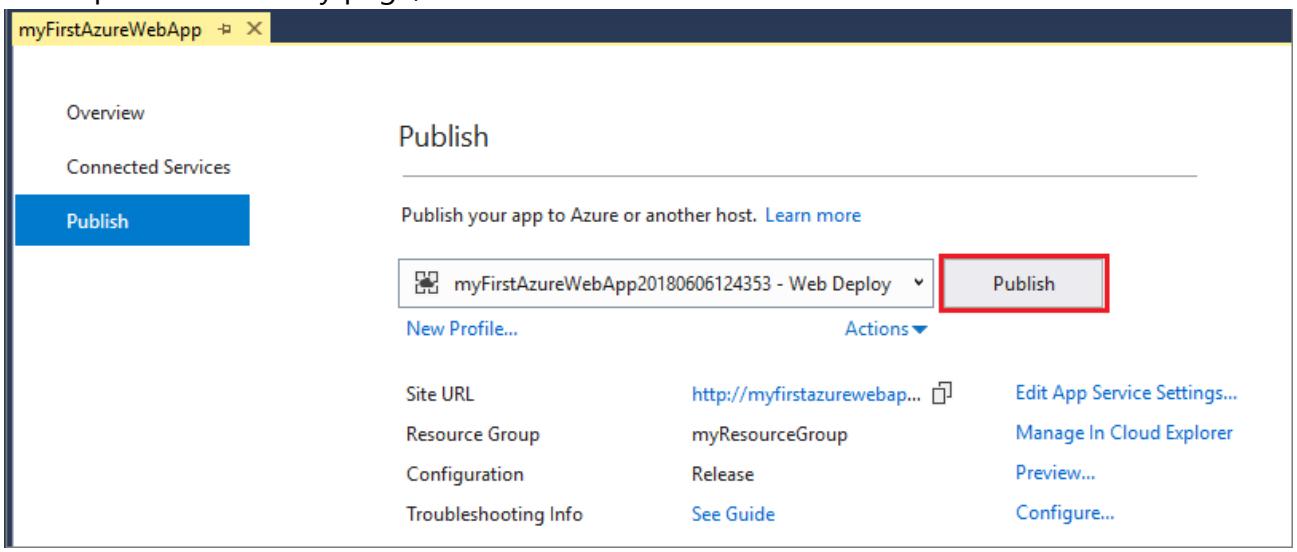
Replace the two `<div>` tags with the following code:

HTML Copy

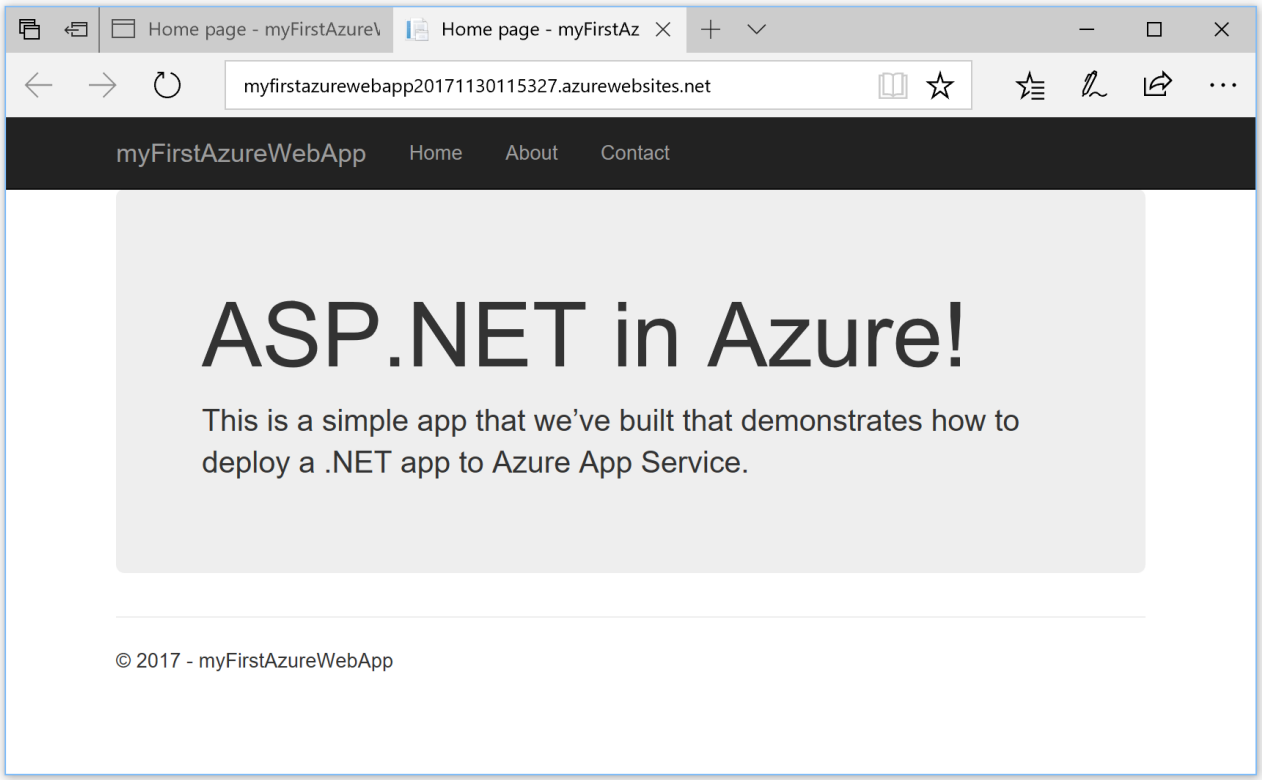
```
<div class="jumbotron">
    <h1>ASP.NET in Azure!</h1>
    <p class="lead">This is a simple app that we've built that demonstrates how
to deploy a .NET app to Azure App Service.</p>
</div>
```

To redeploy to Azure, right-click the **myFirstAzureWebApp** project in **Solution Explorer** and select **Publish**.

In the publish summary page, select **Publish**.



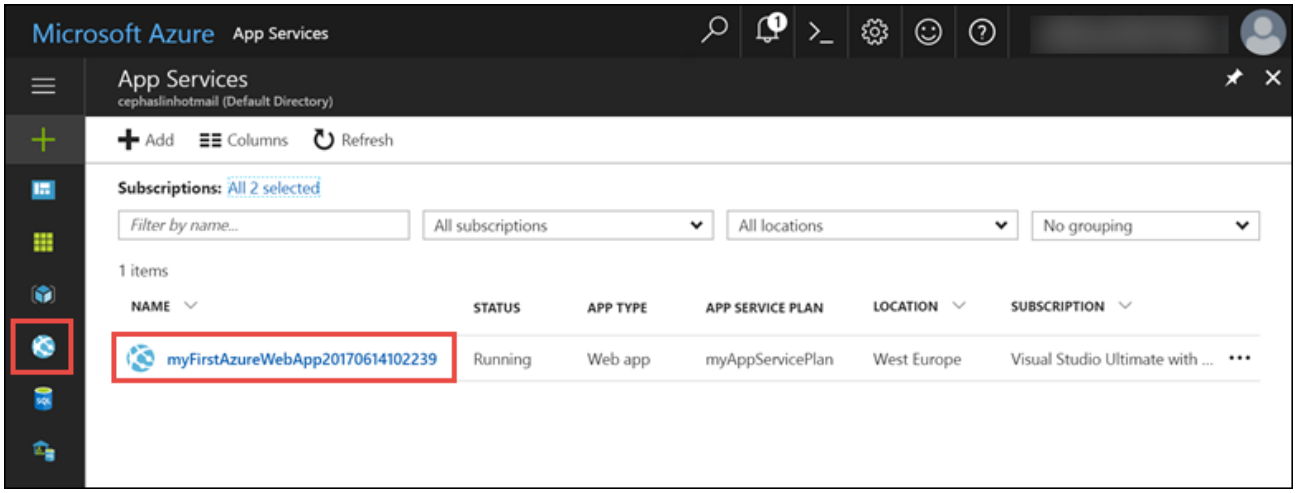
When publishing completes, Visual Studio launches a browser to the URL of the web app.



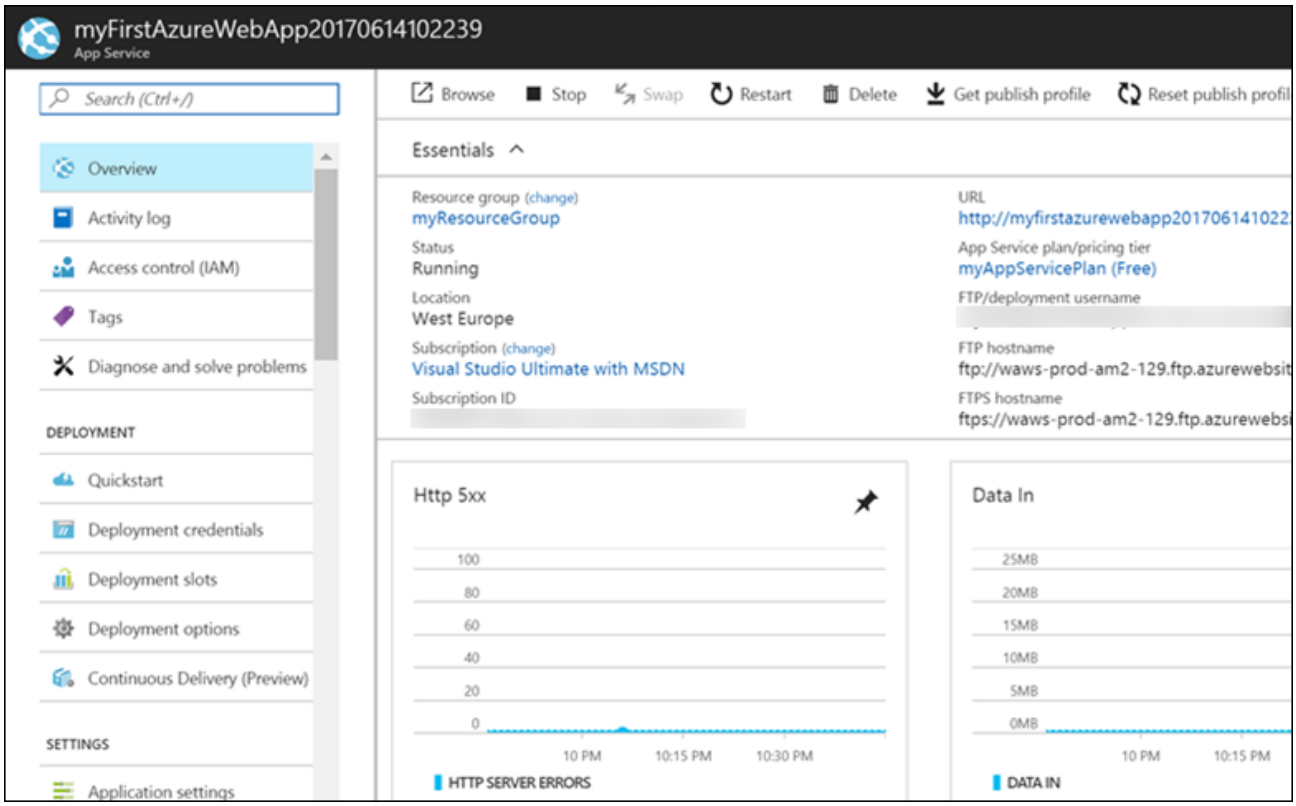
Manage the Azure app

Go to the Azure portal to manage the web app.

From the left menu, select **App Services**, and then select the name of your Azure app.



You see your web app's Overview page. Here, you can perform basic management tasks like browse, stop, start, restart, and delete.



The left menu provides different pages for configuring your app.

Clean up resources

In the preceding steps, you created Azure resources in a resource group. If you don't expect to need these resources in the future, you can delete them by deleting the resource group.

From the left menu in the Azure portal, select **Resource groups** and then select **myResourceGroup**.

On the resource group page, make sure that the listed resources are the ones you want to delete.

Select **Delete**, type **myResourceGroup** in the text box, and then select **Delete**.

Next steps

ASP.NET Core with SQL Database