

[< Previous](#)

Unit 4 of 8 ▾

[Next >](#)

600 XP

# Exercise - Create a website hosted in Azure

7 minutes

This module requires a sandbox to complete. A **sandbox** gives you access to Azure resources. Your Azure subscription will not be charged. The sandbox may only be used to complete training on Microsoft Learn. Use for any other reason is prohibited, and may result in permanent loss of access to the sandbox.

Due to the impact of the global health pandemic, Azure resources are being prioritized towards health and safety organizations. You may experience some issues when you deploy resources used in the exercises. Please try again or choose a different region. For more information, see Azure blog post - [March 28: Update #2 on Microsoft cloud services continuity](#).

[Activate sandbox](#)

As a technology professional, you likely have expertise in a specific area. Perhaps you're a storage admin or virtualization expert, or maybe you focus on the latest security practices. If you're a student, you may still be exploring what interests you most.

No matter your role, most people get started with the cloud by creating a website. Here you'll deploy a website hosted in an App Service.

Let's review some basic terms and get your first website up and running.

## What is an App Service?

Azure App Service is an HTTP-based service that enables you to build and host many types of

web-based solutions without managing infrastructure. For example, you can host web apps, mobile back ends, and RESTful APIs in several supported programming languages. Applications developed in .NET, .NET Core, Java, Ruby, Node.js, PHP, or Python can run in and scale with ease on both Windows and Linux-based environments.

We aim to create a website in less than the time it takes to eat lunch. Therefore, we're not going to write any code and will instead deploy a predefined application from the Microsoft Azure Marketplace.

## What is the Microsoft Azure Marketplace?

The Microsoft Azure Marketplace is an online store that hosts applications that are certified and optimized to run in Azure. Many types of applications are available, ranging from AI + Machine Learning to Web applications. As you'll see in a couple of minutes, deployments from the store are done via the Azure portal using a wizard-style user interface. This user interface makes evaluating different solutions easy.

We're going to use one of the WordPress application options from the Azure Marketplace for our website.

## Creating resources in Azure

Typically, the first thing we'd do is to create a *resource group* to hold all the things that we need to create. The *resource group* allows us to administer all the services, disks, network interfaces, and other elements that potentially make up our solution as a unit. We can use the Azure portal to create and manage our solution's resource groups. However, keep in mind that you can also manage resources via a command line using the Azure CLI. The Azure CLI is a useful option should you need to automate the process in the future.


In the free Azure sandbox environment, you'll use the pre-created resource group **[sandbox resource group name]**, and you don't need to do this step.

## Choosing a location

The free sandbox allows you to create resources in a subset of the Azure global regions. Select a region from this list when you create resources:

- westus2
- southcentralus
- centralus
- eastus
- westeurope
- southeasia
- japaneast
- brazilsouth
- australiasoutheast
- centralindia

## Create a WordPress website

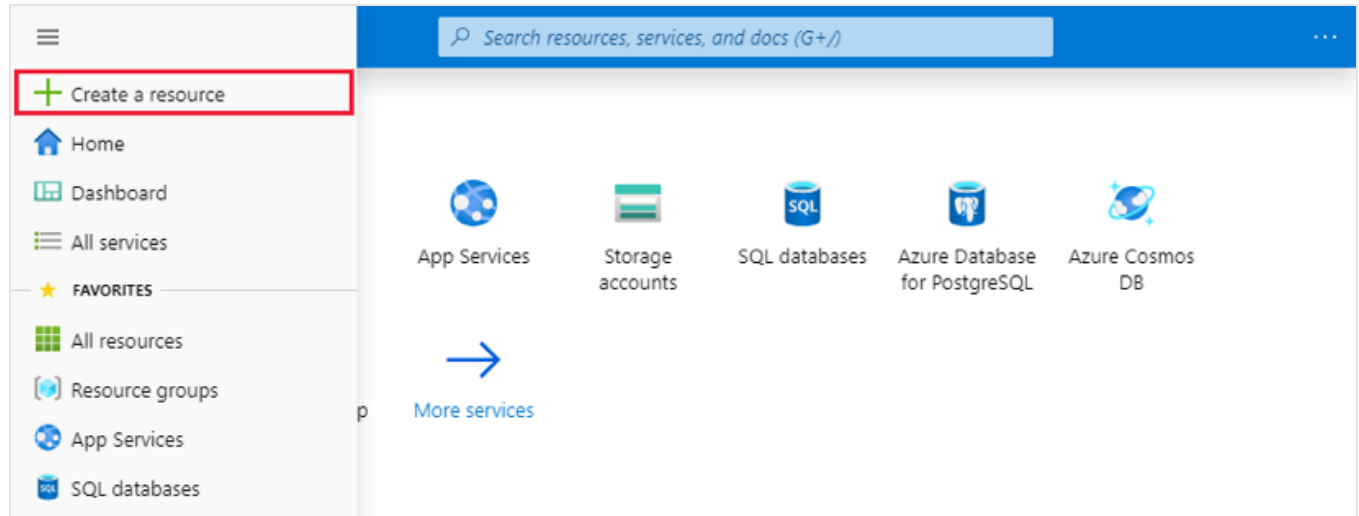
1. If you haven't already, verify that you have activated the sandbox above. Activating the sandbox will allocate the subscription and resource group you will use in this exercise. This step is required for any Microsoft Learn exercises that use a sandbox.
2. Sign in to the [Azure portal](#)  using the same account you activated the sandbox with.

### Note

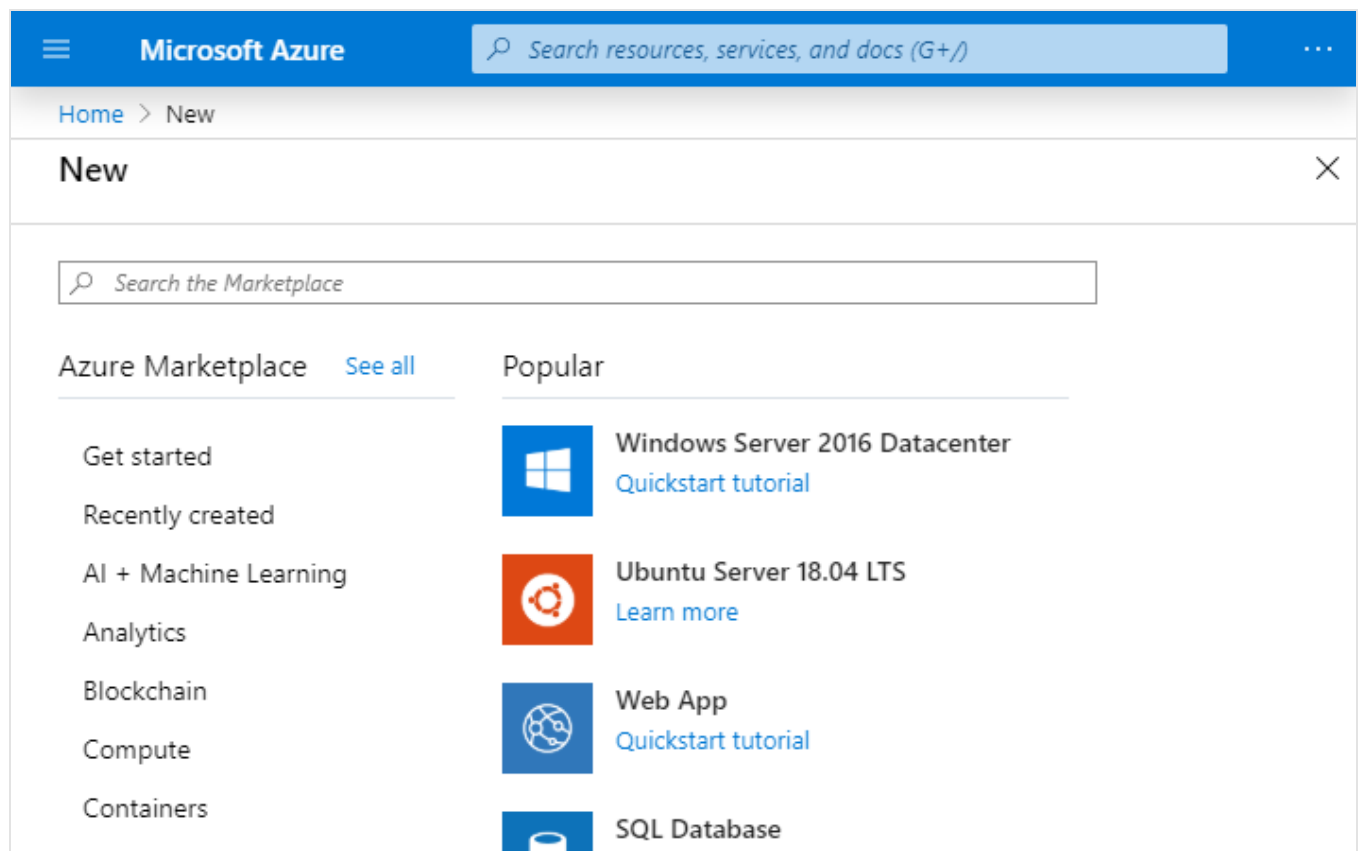
When using the Microsoft Learn Sandbox, it's always good to verify you are in the **Microsoft Learn Sandbox** directory when working within the Azure portal. The directory name is listed either under your email at the top of page or above your account information when you select the user icon. Additionally, make sure you have activated the sandbox at the start of any exercise that uses the sandbox. This will ensure you are allocated an Azure subscription and your own resource group to use for resources created in the exercise.

3. Expand the left-hand navigation panel.

4. From the top of the Azure portal navigation list, select **Create a resource**.

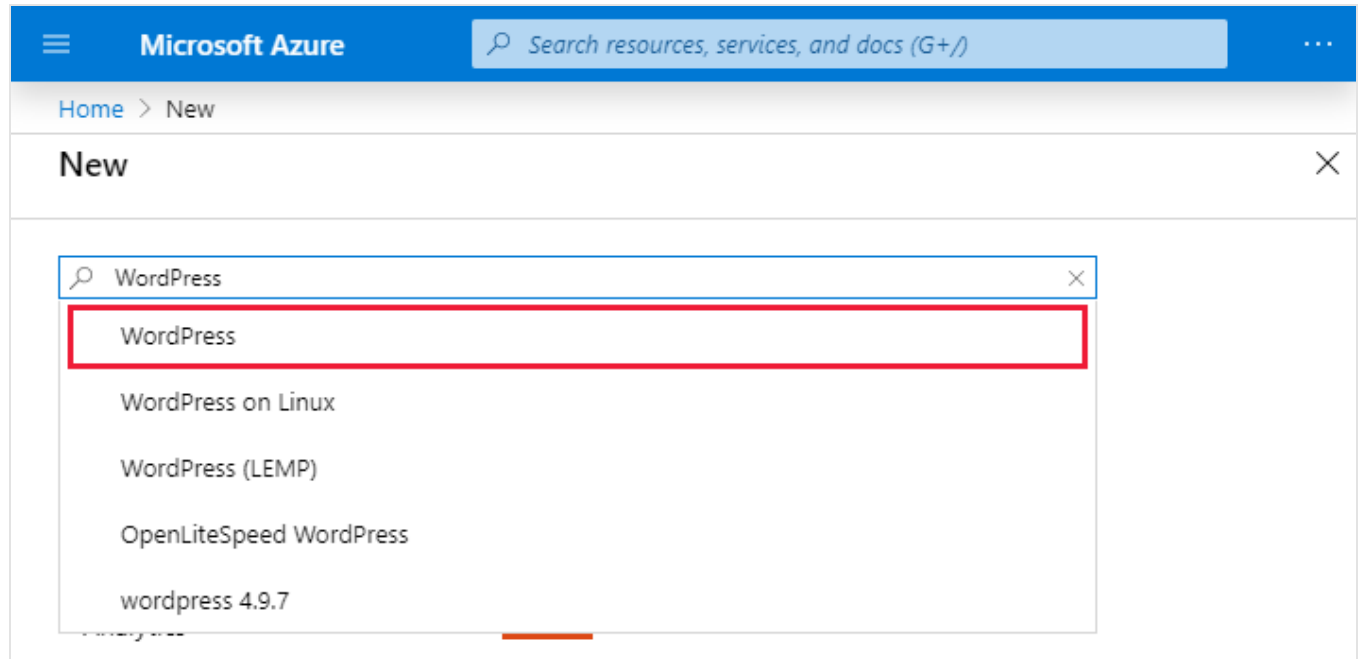


This option takes you to the **Azure Marketplace**.

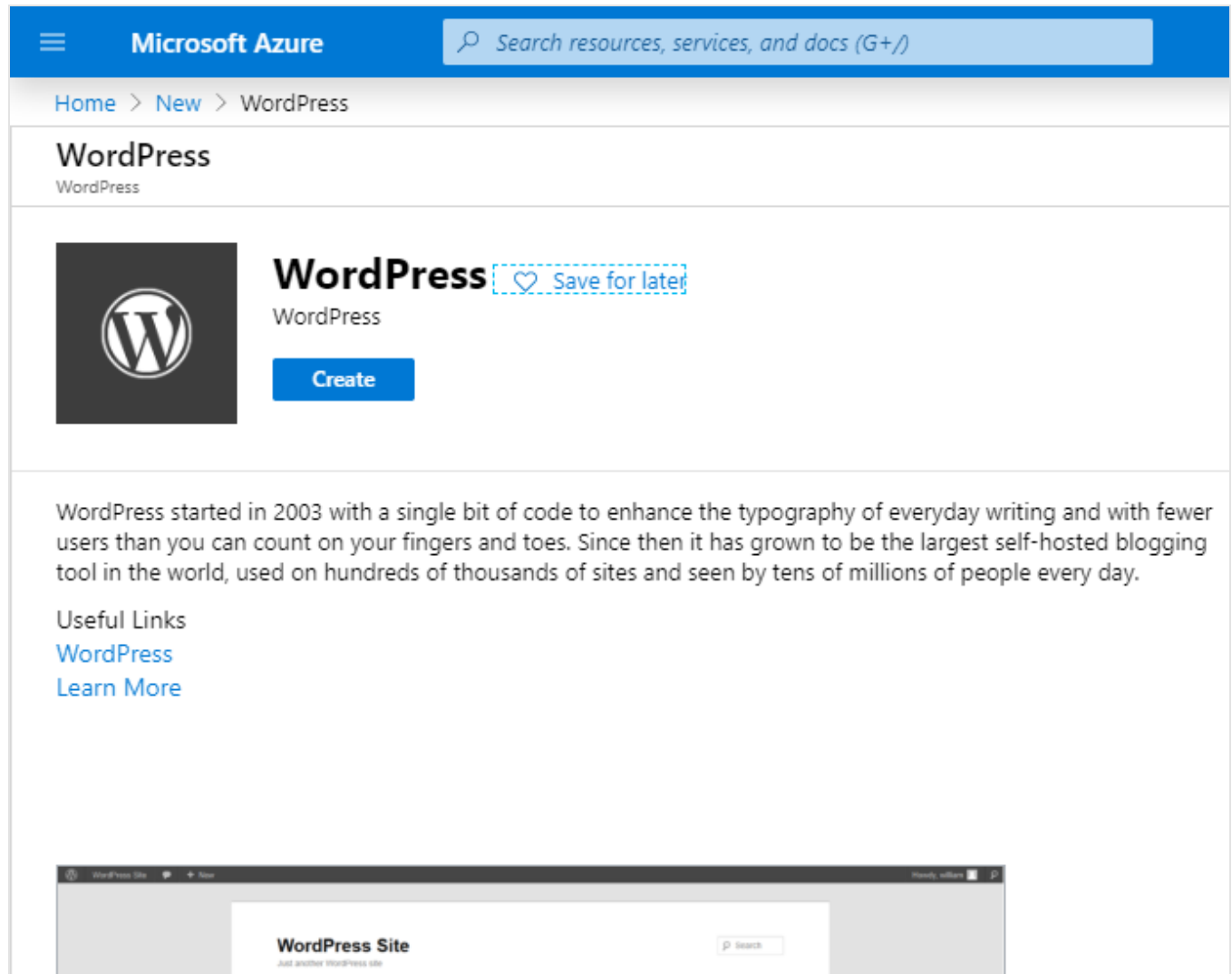


5. The Azure Marketplace has many services, solutions, and resources available for you to use. Since we know that we want to install WordPress, we can do a quick search for it. In the **Search the Marketplace** box above the listed application options, type in

**WordPress.** Select the default *WordPress* option from the list of options available.



6. In the newly presented panel, you'll typically find additional information about the item you're about to install, including the publisher, a brief description of the resource, and links to more information. Make sure to review this information. Select **Create** to begin the process to create a WordPress app.



7. Next, you're presented several options to configure your deployment. Enter the following information:
- App Name:** Choose a unique value for the App name. It will form part of a Fully Qualified Domain Name (FQDN).
  - Subscription:** Make sure the **Concierge Subscription** is selected.
  - Resource Group:** Select the **Use existing** radio button, then select the **[sandbox resource group name]** resource group from the drop-down list.
  - Database Provider:** Select **MySQL in App**.
  - App Service plan/location:** You'll change the App Service plan in the next step.
  - Application Insights:** Leave at the default configuration.

Your configuration should look like this:



The screenshot shows the 'WordPress Create' form in the Azure portal. The form includes the following fields and options:

- App name:** 'BlogFor' (with a green checkmark) and '.azurewebsites.net'.
- Subscription:** 'Concierge Subscription' (dropdown).
- Resource Group:** Radio buttons for 'Create new' and 'Use existing' (selected). The existing group is 'Learn-d451e607-16e0-4534-bb04-98954a4' (dropdown).
- Database Provider:** 'MySQL In App' (dropdown).
- App Service plan/Location:** 'ServicePlan139f3383-9604(Centr...' (dropdown).
- Application Insights:** 'BlogFor' (dropdown).

Below the form is a note box with an information icon and the text: 'MySQL In App runs a local MySQL instance with your app and shares resources from the App Service plan. Note that apps using MySQL In App are not intended for production environments, and they will not scale beyond a single instance.'

At the bottom are a blue 'Create' button and a link for 'Automation options'.

### ! Note

If you still see a section called **Database**, make sure you selected the correct **Database Provider** described in the configuration above.

- Now let's configure the App Service plan to use a specific pricing tier. The App Service plan specifies the compute resources and location for the web app. Select **App Service plan/location**.

**WordPress**  
Create

\* App name  
BlogFor ✓  
.azurewebsites.net

\* Subscription  
Concierge Subscription

\* Resource Group ⓘ  
☐ Create new ☒ Use existing  
Learn-d451e607-16e0-4534-bb04-98954a4

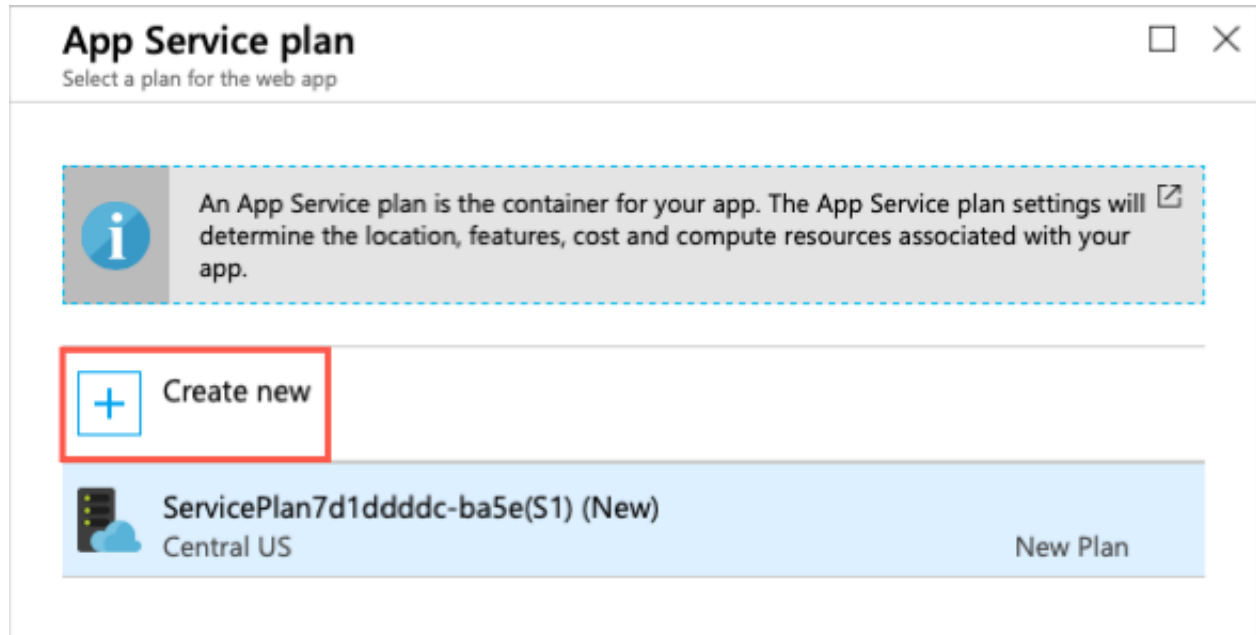
\* Database Provider ⓘ  
MySQL In App

\* App Service plan/Location  
ServicePlan139f3383-9604(Centr... >

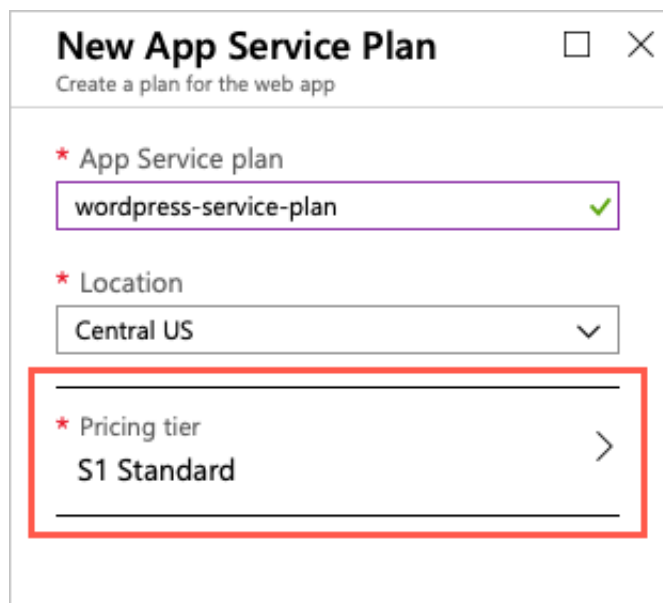
Application Insights  
BlogFor >

9. In the **App Service plan** panel, select **Create new**.





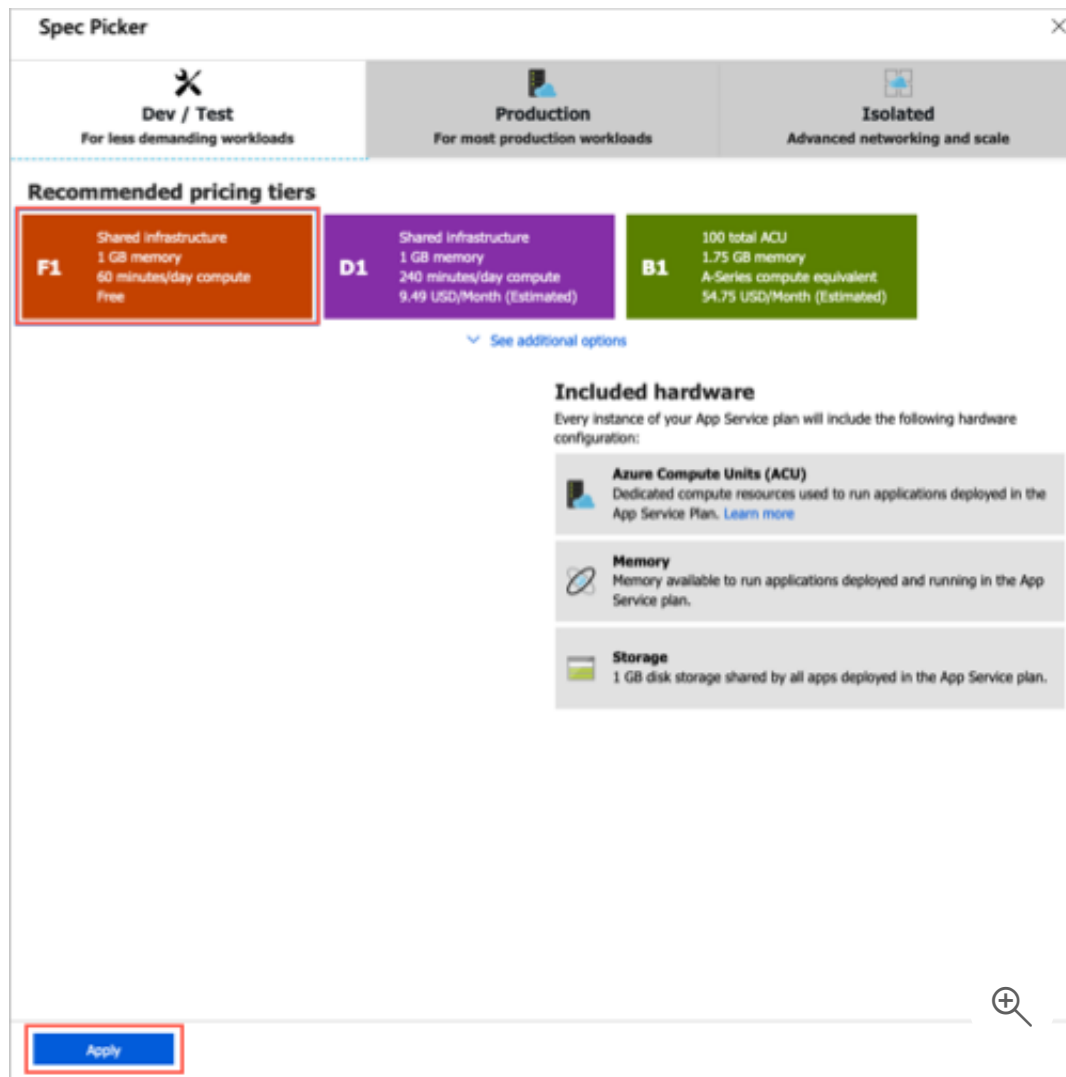
10. In the **New App Service plan** panel, enter a **name** for the new service plan.
11. For **Location**, pick **Central US** to make sure we pick a region that allows the service plan you will pick. Normally, you would pick the region that is closest to your customers while offering the services you need.
12. Select **Pricing tier** to see the performance and feature options of the various types of service plans.



13. The **Spec Picker** allows us to select a new pricing tier for our application. This screen

opens to the *Production* tab, with the S1 pricing tier selected. We'll select a new pricing tier from the *Dev / Test* tab for our website.

Select the **Dev / Test** tab and select the **F1** pricing tier. Then select **Apply**.



14. Back on the **New App Service plan** panel, select **OK** to create the new plan and close the panel.
15. Finally, select the **Create** button to start the deployment of your new site.

#### ! Note

If you encounter an issue creating the resources, verify you've selected the **F1** pricing tier in the new App Service plan. This is a requirement of the sandbox system when creating

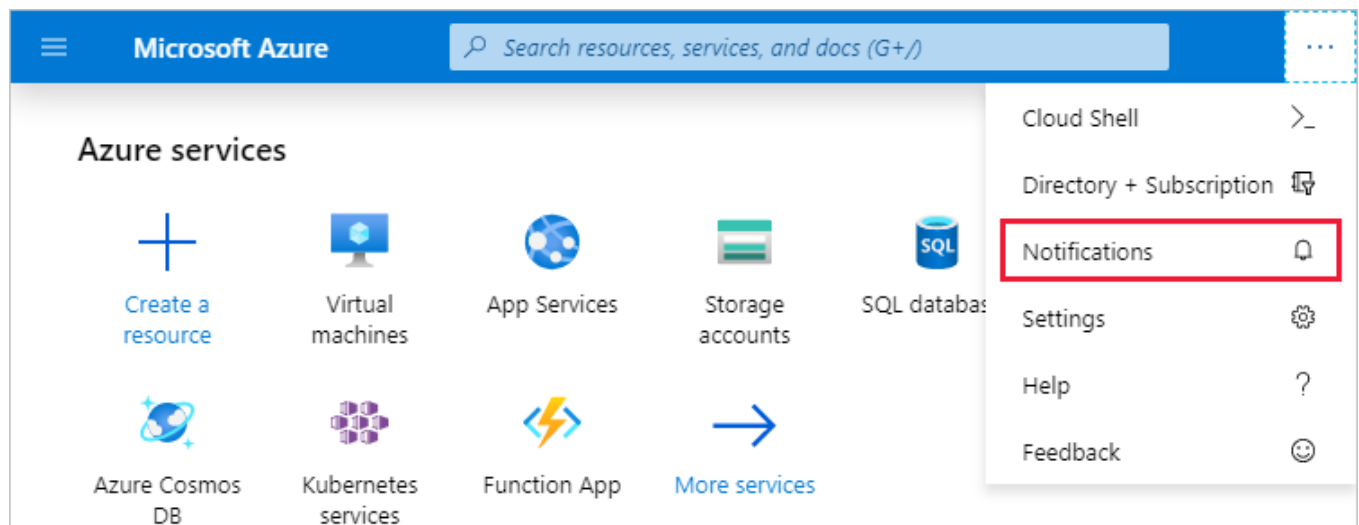
this WordPress site.

## Verify your website is running

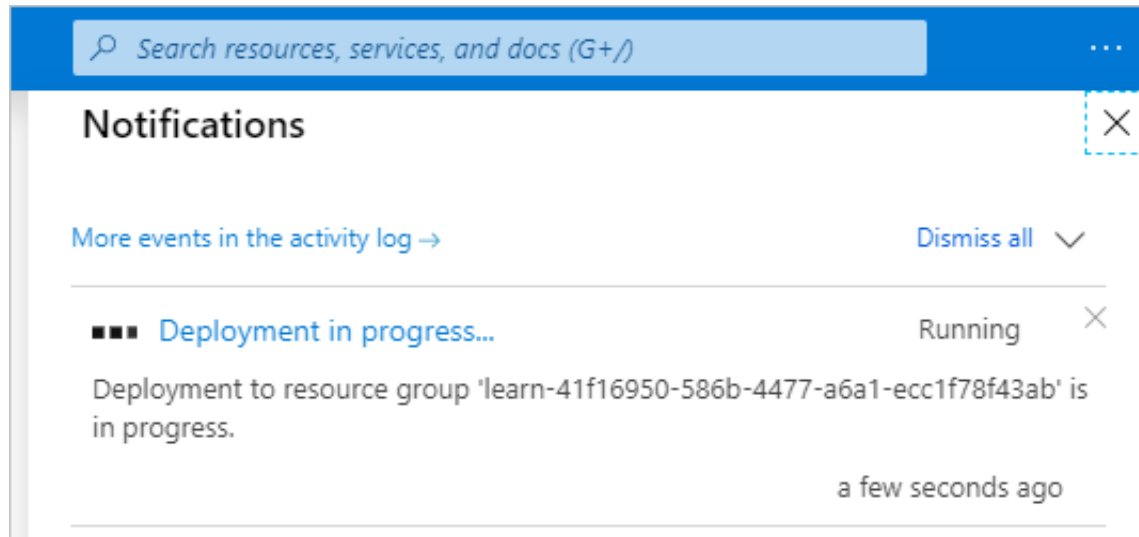
The deployment of the new website can take a few minutes to complete, and you're welcome to explore the portal further on your own.

We can track the progress of the deployment at any time.

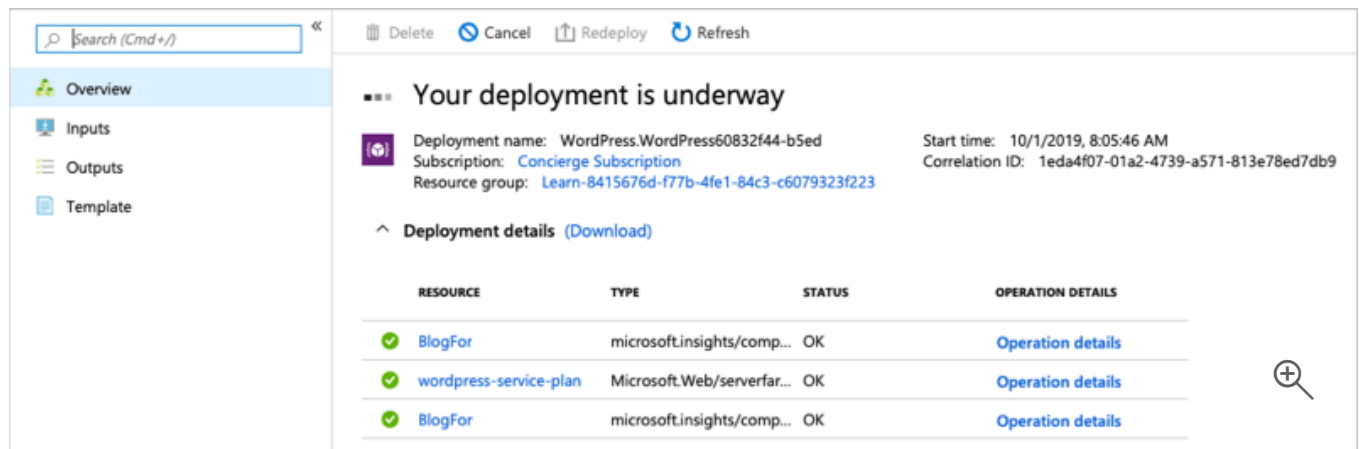
1. Select the **notification bell** icon at the top of the portal. If your browser window width is smaller, it may be shown when you click on the ellipsis (...) icon at the top right.



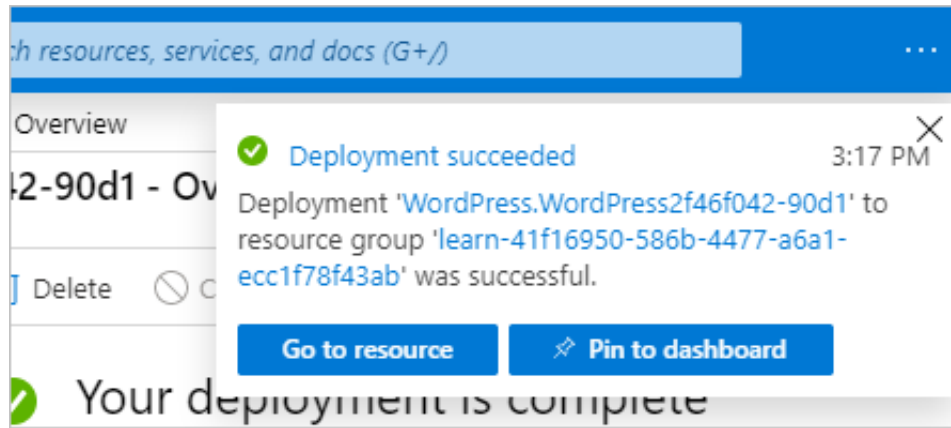
2. Select **Deployment in progress...** to see the details about all the resources that are created.



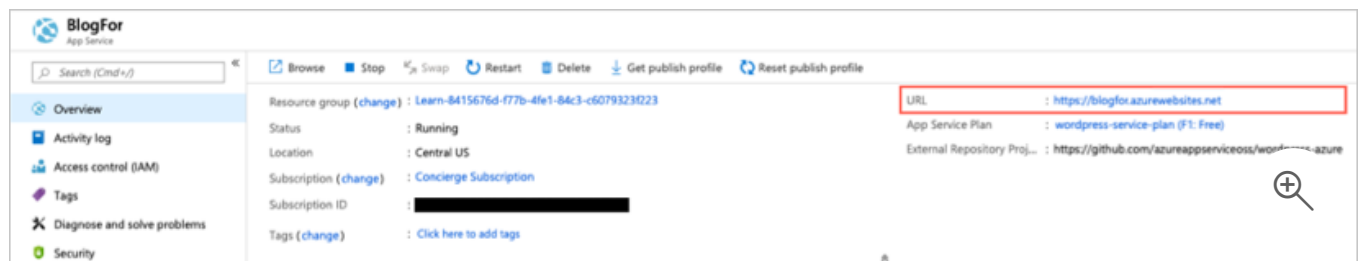
Notice how resources are listed as they're created and the status changes to a green check as each component in the deployment completes.



- Once the deployment status message change to **Your deployment is complete**, you'll notice the status in the notification dialogue changes to **Deployment succeeded**. Select **Go to resource** to navigate to the App Service overview.



4. Find the URL in the Overview section.



Copy the **URL** information. Open a new tab in your browser and use the information to browse to your new WordPress site. You can now configure your WordPress website and add content.

