Create a static HTML web app in Azure

[Azure Web Apps](https://docs.microsoft.com/en-us/azure/app-service/app-service-web-overview) provides a highly scalable, self-patching web hosting service. This quickstart shows how to deploy a basic HTML+CSS site to Azure Web Apps. You create the web app using the [Azure CLI](https://docs.microsoft.com/cli/azure/get-started-with-azure-cli), and you use Git to deploy sample HTML content to the web app.

You can follow the steps below using a Mac, Windows, or Linux machine. Once the prerequisites are installed, it takes about five minutes to complete the steps.

Prerequisites

To complete this quickstart:

* [Install Git](https://git-scm.com/)

If you don't have an Azure subscription, create a [free account](https://azure.microsoft.com/free/?ref=microsoft.com&utm_source=microsoft.com&utm_medium=docs&utm_campaign=visualstudio) before you begin.

Download the sample

In a terminal window, run the following command to clone the sample app repository to your local machine.2

bashCopy

git clone https://github.com/Azure-Samples/html-docs-hello-world.git

Change to the directory that contains the sample code.

bashCopy

cd html-docs-hello-world

View the HTML

Navigate to the directory that contains the sample HTML. Open the *index.html* file in your browser.

Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account. Just click the **Copy** to copy the code, paste it into the Cloud Shell, and then press enter to run it. There are two ways to launch the Cloud Shell:

|  |  |
| --- | --- |
| Click **Try It** in the upper right corner of a code block. | Cloud Shell in this article |
| Click the **Cloud Shell** button on the menu in the upper right of the [Azure portal](https://portal.azure.com/). | Cloud Shell in the portal |
|  |  |

Create a deployment user

In the Cloud Shell, create deployment credentials with the [az webapp deployment user set](https://docs.microsoft.com/en-us/cli/azure/webapp/deployment/user?view=azure-cli-latest" \l "az_webapp_deployment_user_set)command. A deployment user is required for FTP and local Git deployment to a web app. The user name and password are account level. *They are different from your Azure subscription credentials.*

In the following example, replace *<username>* and *<password>* (including brackets) with a new user name and password. The user name must be unique. The password must be at least eight characters long, with two of the following three elements: letters, numbers, symbols.

Azure CLICopyTry It

az webapp deployment user set --user-name <username> --password <password>

If you get a 'Conflict'. Details: 409 error, change the username. If you get a 'Bad Request'. Details: 400 error, use a stronger password.

You create this deployment user only once; you can use it for all your Azure deployments.

Note

Record the user name and password. You need them to deploy the web app later.

Create a resource group

In the Cloud Shell, create a resource group with the [az group create](https://docs.microsoft.com/en-us/cli/azure/group?view=azure-cli-latest" \l "az_group_create) command.

A [resource group](https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-overview#terminology) is a logical container into which Azure resources like web apps, databases, and storage accounts are deployed and managed.

The following example creates a resource group named *myResourceGroup* in the *West Europe*location. To see all supported locations for App Service, run the az appservice list-locationscommand.

Azure CLICopyTry It

az group create --name myResourceGroup --location "West Europe"

You generally create your resource group and the resources in a region near you.

Create an Azure App Service plan

In the Cloud Shell, create an App Service plan with the [az appservice plan create](https://docs.microsoft.com/en-us/cli/azure/appservice/plan?view=azure-cli-latest" \l "az_appservice_plan_create) command.

An [App Service plan](https://docs.microsoft.com/en-us/azure/app-service/azure-web-sites-web-hosting-plans-in-depth-overview) 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)

The following example creates an App Service plan named myAppServicePlan in the **Free** pricing tier:

Azure CLICopyTry It

az appservice plan create --name myAppServicePlan --resource-group myResourceGroup --sku FREE

When the App Service plan has been created, the Azure CLI shows information similar to the following example:

JSONCopy

{

"adminSiteName": null,

"appServicePlanName": "myAppServicePlan",

"geoRegion": "West Europe",

"hostingEnvironmentProfile": null,

"id": "/subscriptions/0000-0000/resourceGroups/myResourceGroup/providers/Microsoft.Web/serverfarms/myAppServicePlan",

"kind": "app",

"location": "West Europe",

"maximumNumberOfWorkers": 1,

"name": "myAppServicePlan",

< JSON data removed for brevity. >

"targetWorkerSizeId": 0,

"type": "Microsoft.Web/serverfarms",

"workerTierName": null

}

Create a web app

In the Cloud Shell, create a [web app](https://docs.microsoft.com/en-us/azure/app-service/app-service-web-overview) in the myAppServicePlan App Service plan with the [az webapp create](https://docs.microsoft.com/en-us/cli/azure/webapp?view=azure-cli-latest" \l "az_webapp_create) command.

In the following example, replace *<app\_name>* with a globally unique app name (valid characters are a-z, 0-9, and -).

Azure CLICopyTry It

az webapp create --name <app\_name> --resource-group myResourceGroup --plan myAppServicePlan --deployment-local-git

When the web app has been created, the Azure CLI shows information similar to the following example:

JSONCopy

Local git is configured with url of 'https://<username>@<app\_name>.scm.azurewebsites.net/<app\_name>.git'

{

"availabilityState": "Normal",

"clientAffinityEnabled": true,

"clientCertEnabled": false,

"cloningInfo": null,

"containerSize": 0,

"dailyMemoryTimeQuota": 0,

"defaultHostName": "<app\_name>.azurewebsites.net",

"deploymentLocalGitUrl": "https://<username>@<app\_name>.scm.azurewebsites.net/<app\_name>.git",

"enabled": true,

< JSON data removed for brevity. >

}

You’ve created an empty web app, with git deployment enabled.

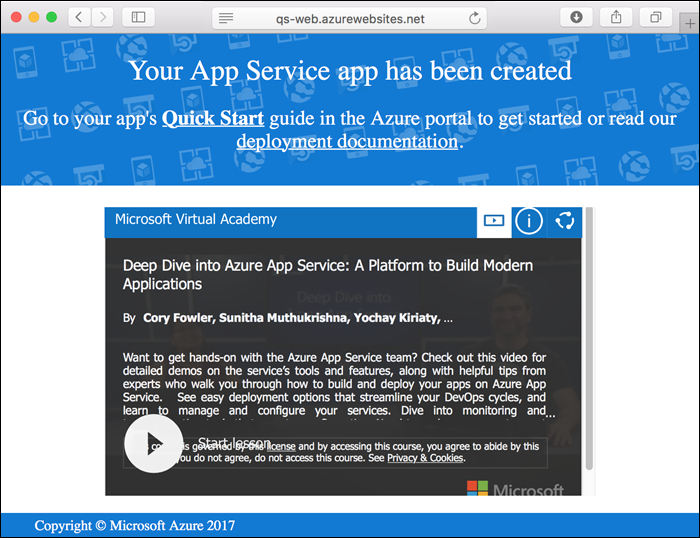
Note

The URL of the Git remote is shown in the deploymentLocalGitUrl property, with the format https://<username>@<app\_name>.scm.azurewebsites.net/<app\_name>.git. Save this URL as you'll need it later.

Browse to the newly created web app.

bashCopy

http://<app\_name>.azurewebsites.net



Push to Azure from Git

In the local terminal window, add an Azure remote to your local Git repository. This Azure remote was created for you in [Create a web app](https://docs.microsoft.com/en-us/azure/app-service/app-service-web-get-started-html#create-a-web-app).

bashCopy

git remote add azure <deploymentLocalGitUrl-from-create-step>

Push to the Azure remote to deploy your app with the following command. When prompted for a password, make sure that you enter the password you created in [Configure a deployment user](https://docs.microsoft.com/en-us/azure/app-service/app-service-web-get-started-html#configure-a-deployment-user), not the password you use to log in to the Azure portal.

bashCopy

git push azure master

This command may take a few minutes to run. While running, it displays information similar to the following example:

bashCopy

Counting objects: 13, done.

Delta compression using up to 4 threads.

Compressing objects: 100% (11/11), done.

Writing objects: 100% (13/13), 2.07 KiB | 0 bytes/s, done.

Total 13 (delta 2), reused 0 (delta 0)

remote: Updating branch 'master'.

remote: Updating submodules.

remote: Preparing deployment for commit id 'cc39b1e4cb'.

remote: Generating deployment script.

remote: Generating deployment script for Web Site

remote: Generated deployment script files

remote: Running deployment command...

remote: Handling Basic Web Site deployment.

remote: KuduSync.NET from: 'D:\home\site\repository' to: 'D:\home\site\wwwroot'

remote: Deleting file: 'hostingstart.html'

remote: Copying file: '.gitignore'

remote: Copying file: 'LICENSE'

remote: Copying file: 'README.md'

remote: Finished successfully.

remote: Running post deployment command(s)...

remote: Deployment successful.

To https://<app\_name>.scm.azurewebsites.net/<app\_name>.git

\* [new branch] master -> master

Browse to the app

In a browser, go to the Azure web app URL: http://<app\_name>.azurewebsites.net.

The page is running as an Azure App Service web app.



**Congratulations!** You've deployed your first HTML app to App Service.

Update and redeploy the app

Open the *index.html* file in a text editor, and make a change to the markup. For example, change the H1 heading from "Azure App Service - Sample Static HTML Site" to just "Azure App Service`.

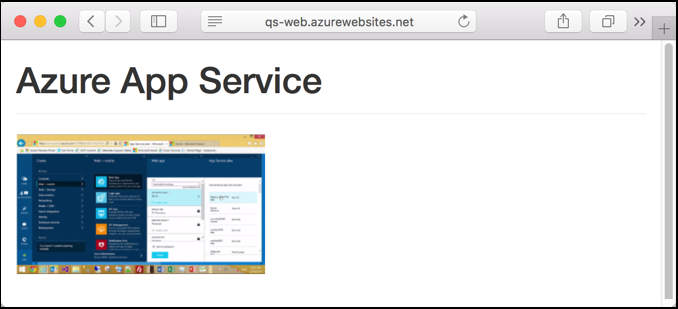
In the local terminal window, commit your changes in Git, and then push the code changes to Azure.

bashCopy

git commit -am "updated HTML"

git push azure master

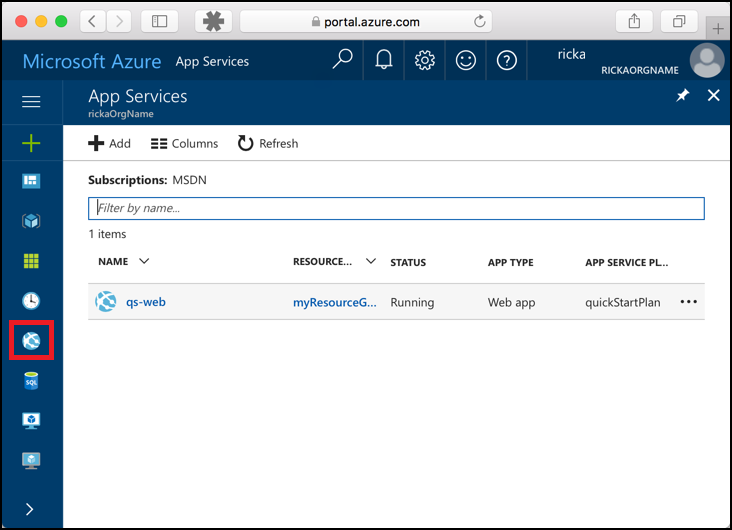
Once deployment has completed, refresh your browser to see the changes.



Manage your new Azure web app

Go to the [Azure portal](https://portal.azure.com/) to manage the web app you created.

From the left menu, click **App Services**, and then click the name of your Azure web 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, delete the resource group by running the following command in the Cloud Shell:+

Azure CLICopyTry It

az group delete --name myResourceGroup

This command may take a minute to run.

Next steps