

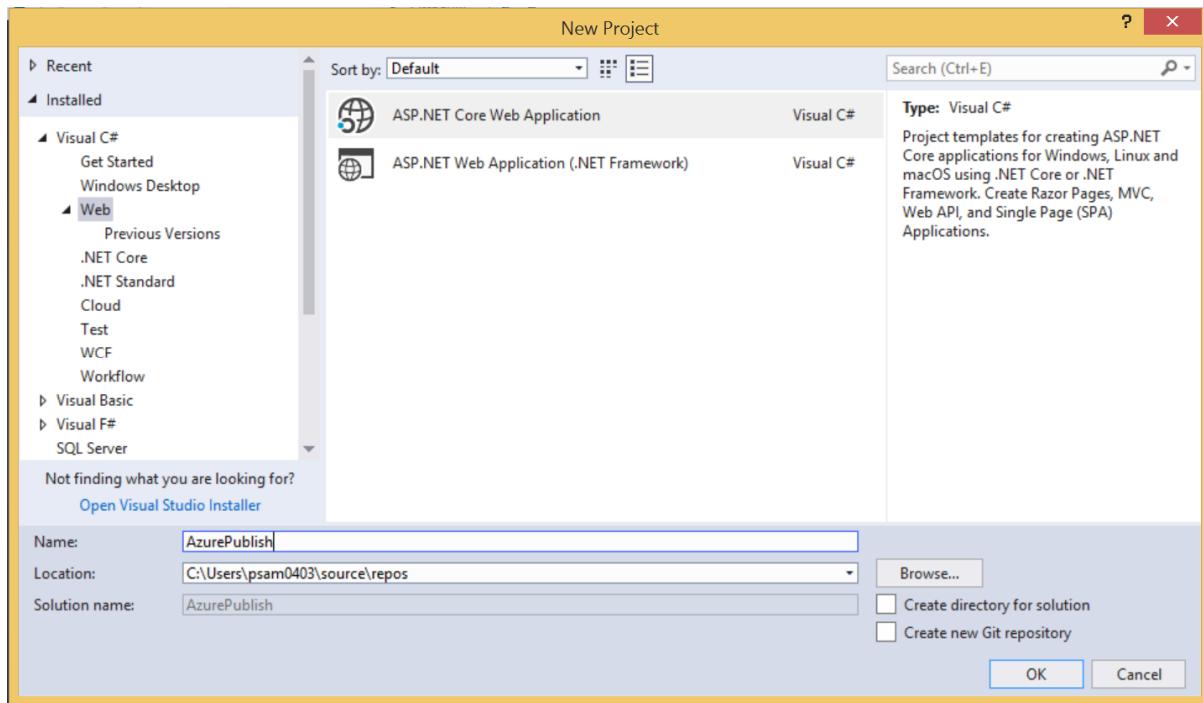
How to Publish ASP.NET Core App to Azure through FTP

In this tutorial I'm going to show how to:

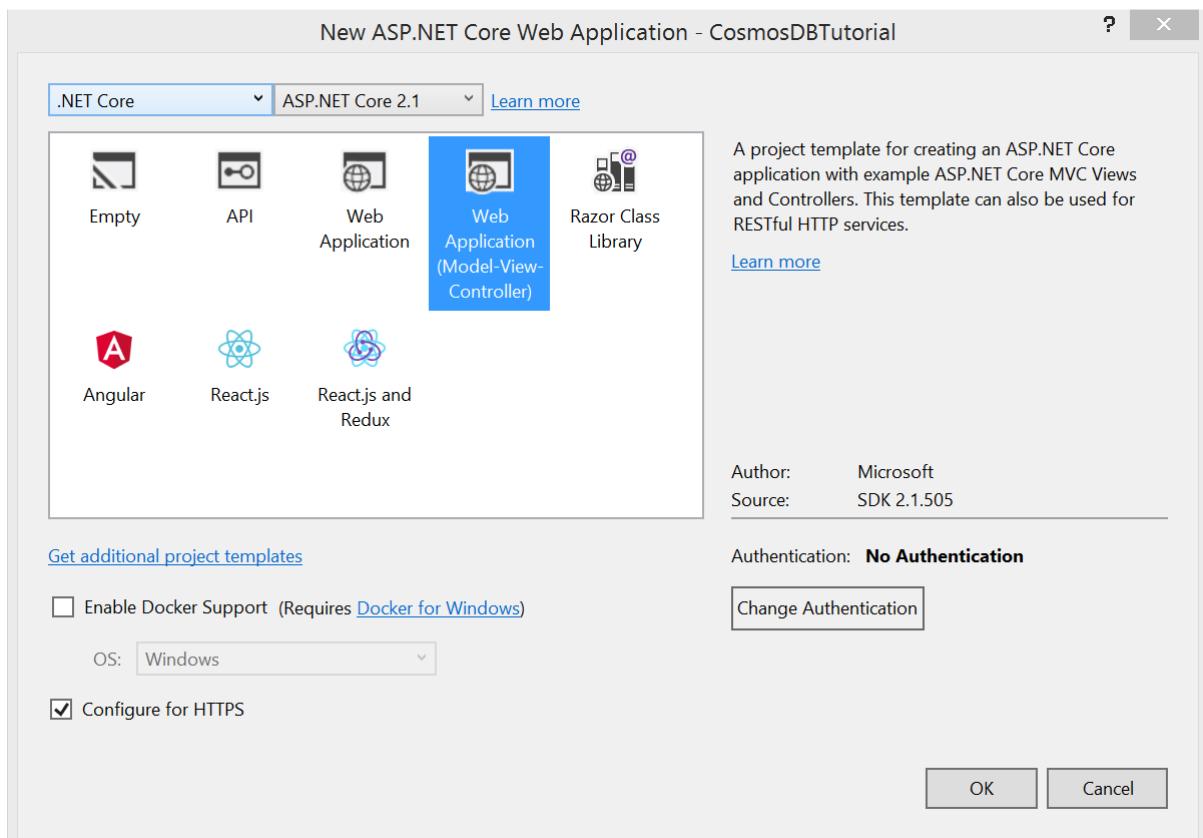
- Create an ASP.NET Core MVC Application
- Create an App Service Plan
- Create an App Service
- Publish the code to Azure through FTP

Step 01: Create an ASP.NET Core MVC Application

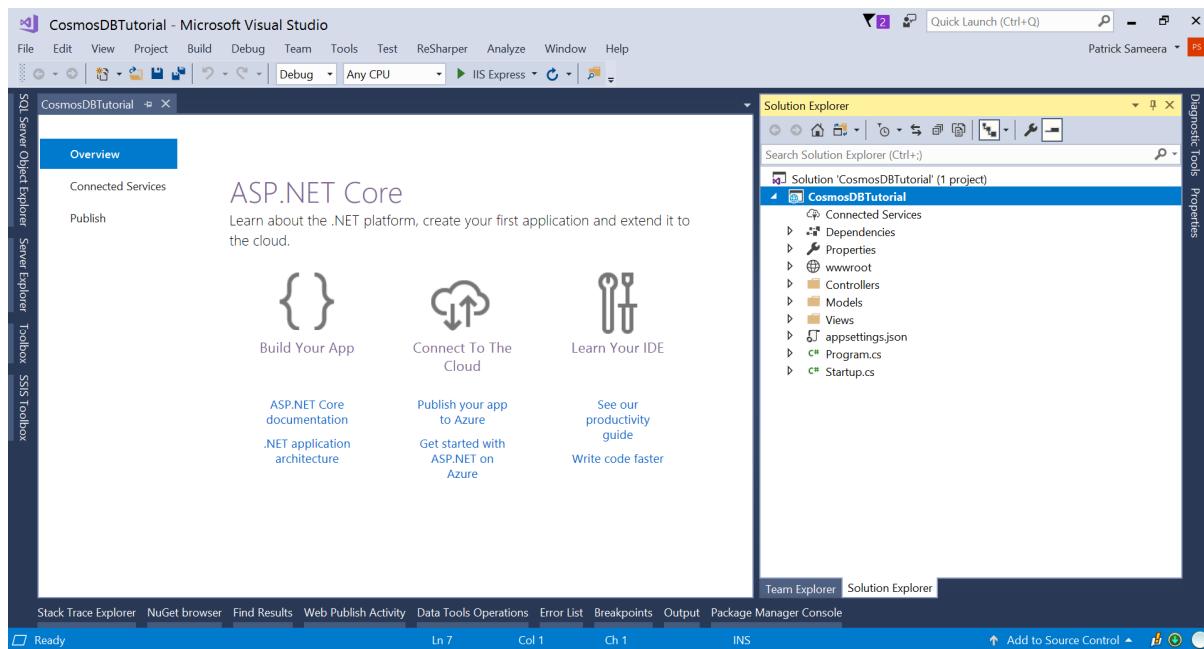
Create a new ASP .Net Core Web Project – Name it AzurePublish.



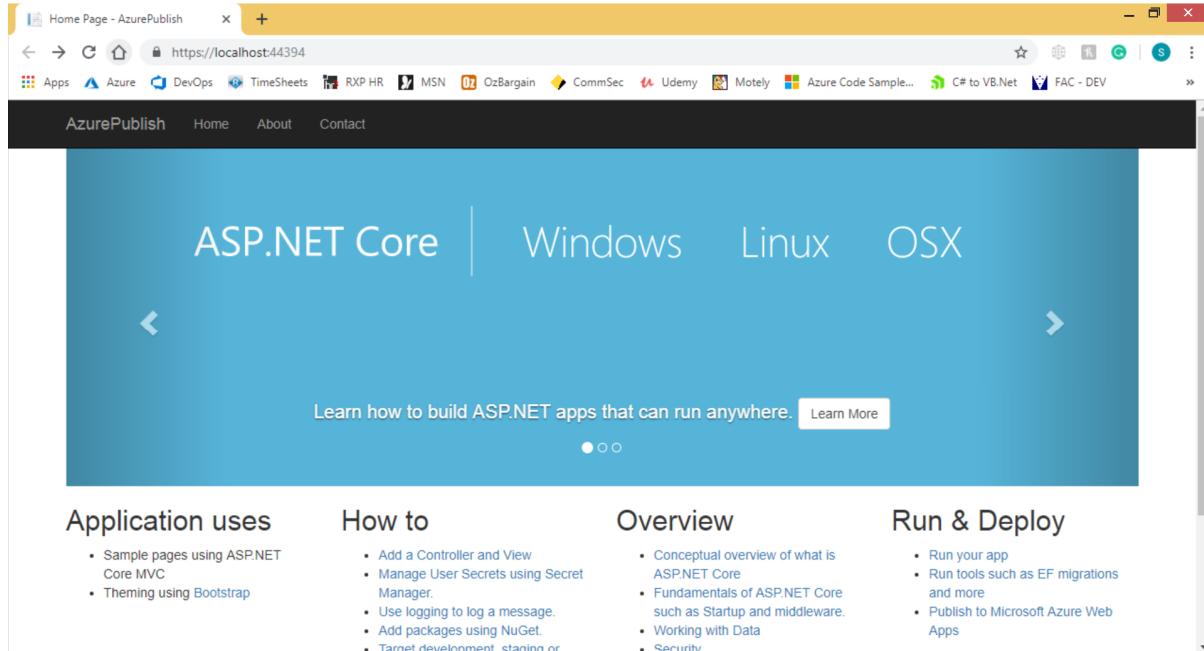
- Net Core 2.1
- MVC



Once Project gets created Click F5 to run the Project, just to make sure it compiles properly.

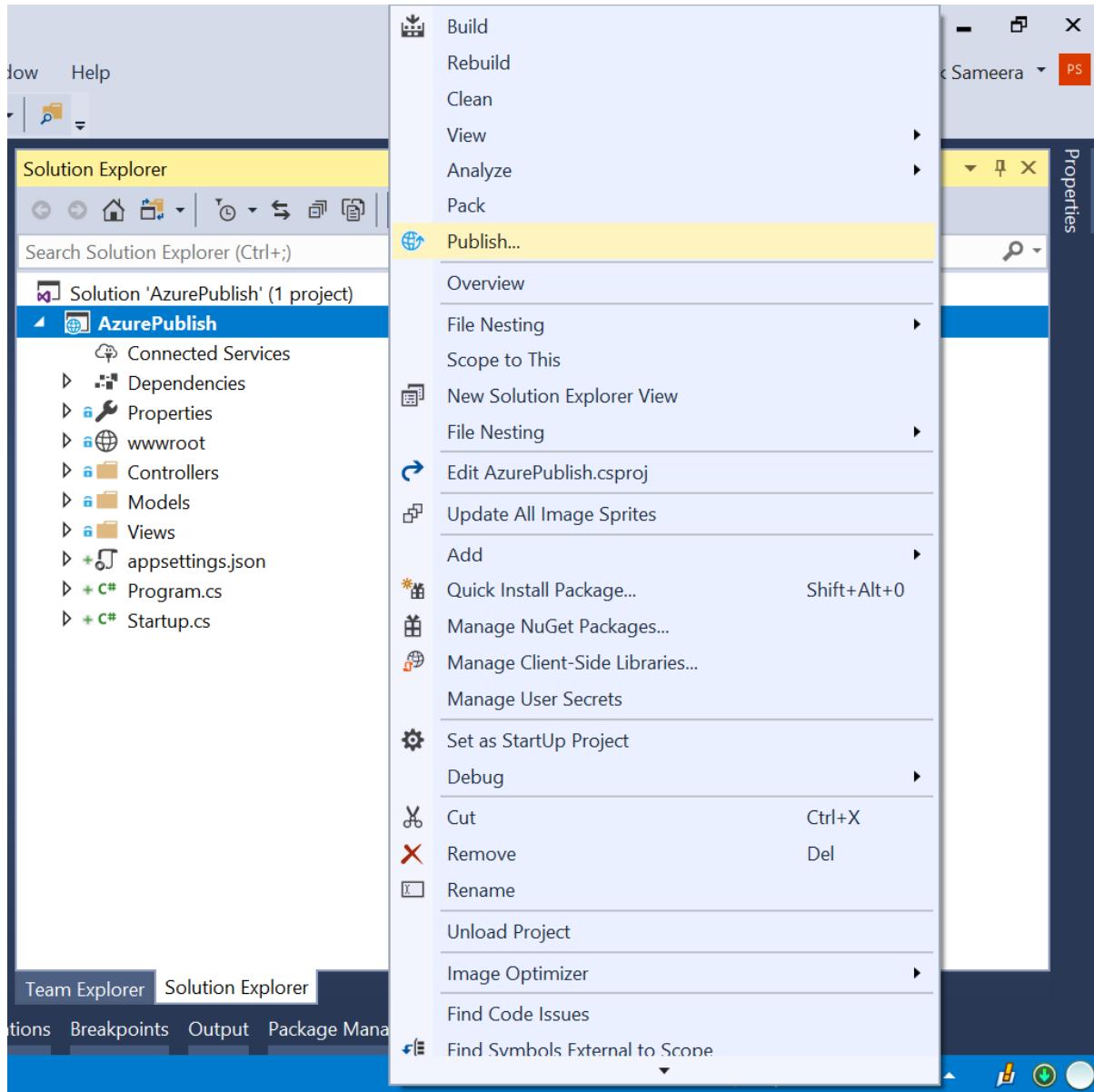


It runs properly.

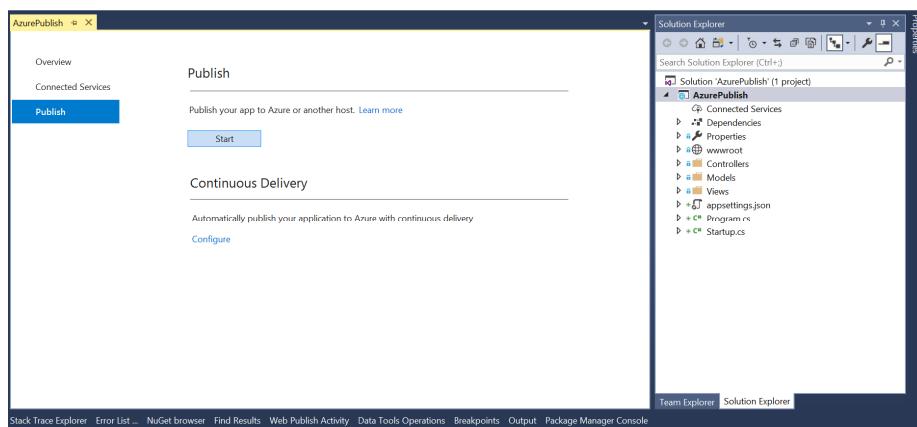


Next we need to Publish the App to a local Folder.

Right click on the project and click Publish.



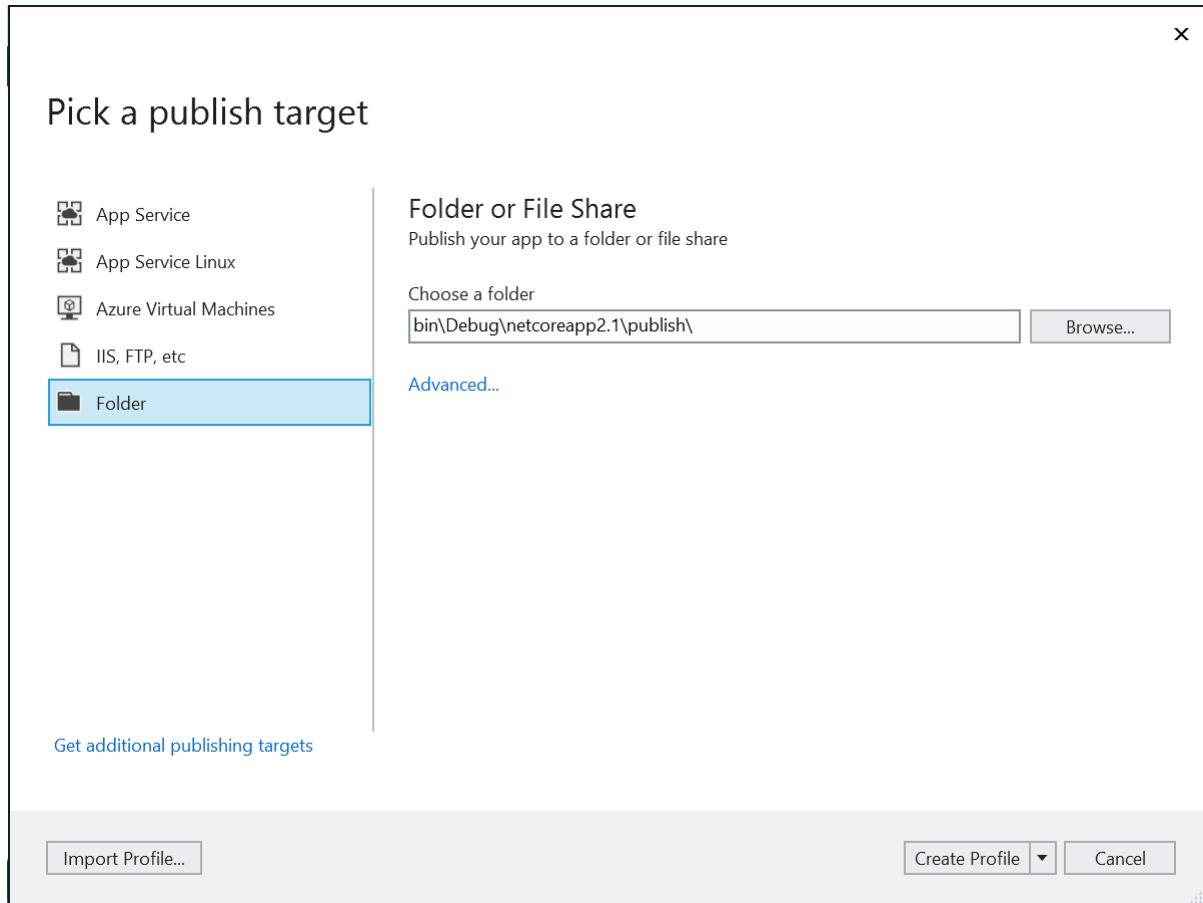
If we haven't set a Publish Profile, then click Start.



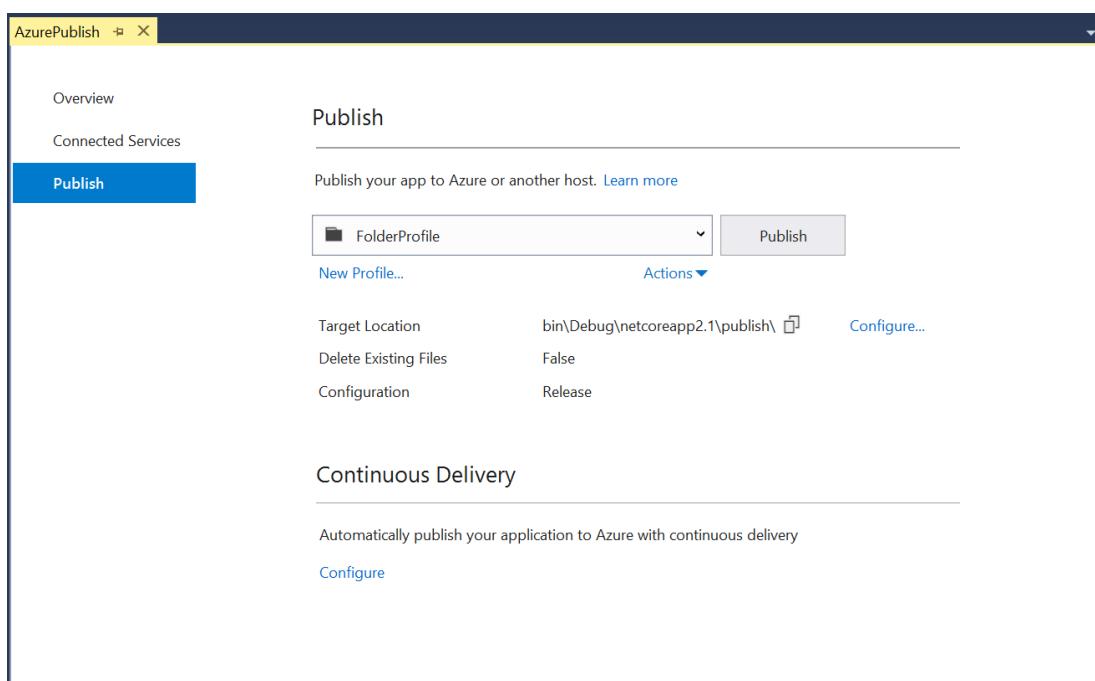
Select Folder option and set the location and click Create Profile button.

In my case I have set `bin\Debug\netcoreapp2.1\publish\`

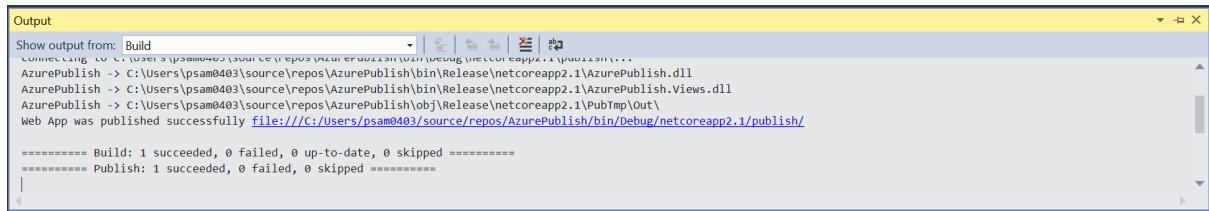
- `C:\Users\psam0403\source\repos\AzurePublish\bin\Debug\netcoreapp2.1\publish\`



Click Publish.



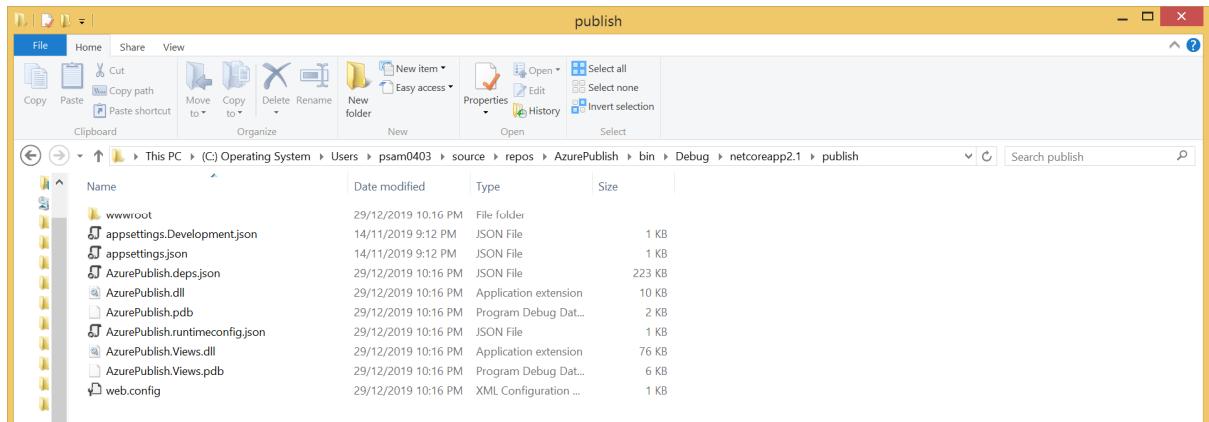
Make sure it gets published correctly.



```
Show output from: Build
Connecting to C:\Users\psam0403\source\repos\AzurePublish\bin\Release\netcoreapp2.1\AzurePublish...
AzurePublish -> C:\Users\psam0403\source\repos\AzurePublish\bin\Release\netcoreapp2.1\AzurePublish.dll
AzurePublish -> C:\Users\psam0403\source\repos\AzurePublish\bin\Release\netcoreapp2.1\AzurePublish.Views.dll
AzurePublish -> C:\Users\psam0403\source\repos\AzurePublish\obj\Release\netcoreapp2.1\pubTmp\Out\
Web App was published successfully file:///C:/users/psam0403/source/repos/AzurePublish/bin/Debug/netcoreapp2.1/publish/
=====
Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped =====
Publish: 1 succeeded, 0 failed, 0 skipped =====
```

Go to the Publish folder location and make sure published folder exists.

- C:\Users\psam0403\source\repos\AzurePublish\bin\Debug\netcoreapp2.1\publish



Step 02: Create an App Service Plan

I have a Resource Group already existing and I will be using that resource group for this tutorial.

- Name: patsam-rg
- Location: Central US

The screenshot shows the 'Resource groups' blade in the Azure portal. At the top, there are buttons for '+ Add', 'Edit columns', 'Refresh', 'Export to CSV', 'Assign tags', and 'Feedback'. Below these are filters for 'Filter by name...', 'Subscription == all', 'Location == all', and 'Add filter'. A dropdown menu shows 'No grouping'. The main table lists one record: 'patsam-rg' under 'Name', 'Visual Studio Enterprise - MPN' under 'Subscription', and 'Central US' under 'Location'. There are also 'Subscription ↑↓' and 'Location ↑↓' sort buttons.

Go to App Service Plans and click + Add button to create a new App Service Plan.

The screenshot shows the 'App Service plans' blade in the Azure portal. At the top, there are buttons for '+ Add', 'Edit columns', 'Refresh', and 'Assign tags'. A message 'Subscriptions: Visual Studio Enterprise – MPN – Don't see a subscription? [Open Directory + Subscription settings]' is displayed. Below are filters for 'Filter by name...', 'All resource groups', 'All locations', 'All tags', and 'No grouping'. The table header includes columns for 'Name ↑↓', 'Apps', 'Pricing Tier', 'Resource group ↑↓', and 'Subscription ↑↓'. The main area displays a placeholder icon and the text 'No app service plans to display'. A note below explains what App Service plans are and provides a link to learn more. A prominent blue button at the bottom center says 'Create app service plan'.

Provide relevant details.

- Resource group: patsam-rg
- App Service Plan Name: patsam-ps
- Operating system: Windows
- Sku and size: Free F1

The screenshot shows the 'App Service Plan' creation page in the Azure portal. The 'Basics' tab is selected. Under 'Project Details', the subscription is set to 'Visual Studio Enterprise - MPN', resource group to 'patsam-rg', and app name to 'patsam-ps'. The operating system is set to 'Windows'. The region is 'Central US'. In the 'Pricing Tier' section, 'Free F1' is selected. At the bottom, there are buttons for 'Review + create', '< Previous', and 'Next : Tags >'.

Click on Change Size to select the pricing tier. For this example, I'll be using Free F1.

The screenshot shows the 'Spec Picker' dialog. It has three main sections: 'Dev / Test' (for less demanding workloads), 'Production' (for most production workloads), and 'Isolated' (advanced networking and scale). Below these, under 'Recommended pricing tiers', three options are shown: 'F1 Shared infrastructure, 1 GB memory, 60 minutes/day compute, Free' (highlighted with a blue border), 'D1 Shared infrastructure, 1 GB memory, 240 minutes/day compute, 13.03 AUD/Month (Estimated)', and 'B1 100 total ACU, 1.75 GB memory, A-Series compute equivalent, 44.10 AUD/Month (Estimated)'. There is also a link to 'See additional options'. Below this, the 'Included hardware' section lists 'Azure Compute Units (ACU)', 'Memory', and 'Storage'. At the bottom is a large 'Apply' button.

Patrick Sameera Jayalath

Click Review + Create button.

The screenshot shows the 'Microsoft.Web-ASP-Portal-75fa8464-8519 - Overview' page. On the left, there's a navigation menu with 'Overview', 'Inputs', 'Outputs', and 'Template'. The main area has a green checkmark icon and the message 'Your deployment is complete'. Below it, deployment details are listed: Deployment name: Microsoft.Web-ASP-Portal-75fa8464-8519, Subscription: Visual Studio Enterprise – MPN, Start time: 11/14/2019, 9:36:42 PM, Correlation ID: dae92244-1b0b-487b-b758-d479eead3694, and Resource group: patsam-rg. There are sections for 'Deployment details' (with a 'Download' link) and 'Next steps' (with a 'Go to resource' button). A search bar at the top left says 'Search (Ctrl+I)'.

We can see the newly created App Service Plan.

The screenshot shows the 'App Service plans' page. At the top, it says 'App Service plans' and 'RDP Services Ltd'. Below that are buttons for '+ Add', 'Edit columns', 'Refresh', and 'Assign tags'. A note says 'Subscriptions: Visual Studio Enterprise – MPN – Don't see a subscription? Open Directory + Subscription settings'. There are filters for 'Filter by name...', 'All resource groups', 'All locations', 'All tags', and 'No grouping'. The table below shows one item: 'Name' is 'patsam-sp', 'Apps' is '1', 'Pricing Tier' is 'patsam-sp (F1: Free)', 'Resource group' is 'patsam-rg', and 'Subscription' is 'Visual Studio Enterprise – M...'. There are three dots at the end of the subscription row.

Step 03: Create an App Service

Go to App Service and click + Add button to create a new App Service.

The screenshot shows the 'App Services' blade in the Azure portal. At the top, there are navigation links for 'Home > App Services' and a 'RXP Services Ltd' account indicator. Below the header is a toolbar with buttons for '+ Add', 'Edit columns', 'Refresh', 'Export to CSV', 'Assign tags', 'Start', 'Restart', 'Stop', 'Delete', 'Feedback', and 'More'. A search bar labeled 'Filter by name...' is highlighted with a blue border. To its right are filters for 'Subscription == all', 'Resource group == all', 'Location == all', and a 'No grouping' dropdown. Below the toolbar, a message says 'Showing 0 to 0 of 0 records.' followed by a table header with columns: Name ↑↓, Status ↑↓, Location ↑↓, Pricing Tier ↑↓, App Service Plan ↑↓, Subscription ↑↓, and App. In the center of the page is a large circular icon with a network-like pattern. Below the icon, the text 'No app services to display' is centered. At the bottom, there is descriptive text about building web, mobile, and API apps using .NET, Java, Node.js, PHP, and Python, followed by a 'Learn more' link. A prominent blue 'Create app service' button is located at the bottom center of the page.

Provide relevant details.

- Resource group: patsam-rg
- App Service Name: patsam
- Publish: Code
- Runtime stack: .Net Core 2.1 (since we created a .Net Core 2.1 MVC Web Application)
- Operating system: Windows
- App Service Plan: patsam-sp (App Service Plan we created earlier)

The screenshot shows the 'Basics' step of the 'Web App' creation wizard in the Azure portal. The page title is 'Home > App Services > Web App'. The sub-section title is 'Web App'. The tabs at the top are 'Basics' (selected), 'Monitoring', 'Tags', and 'Review + create'. The main content area is titled 'Project Details' and includes fields for 'Subscription' (Visual Studio Enterprise – MPN) and 'Resource Group' (patsam-rg). Below this is the 'Instance Details' section with fields for 'Name' (patsam), 'Publish' (Code selected), 'Runtime stack' (.NET Core 2.1 (LTS)), 'Operating System' (Windows selected), and 'Region' (Central US). The 'App Service Plan' section shows a dropdown for 'Windows Plan (Central US)' which is currently empty. At the bottom are buttons for 'Review + create', '< Previous', and 'Next : Monitoring >'.

Click Review + Create button.

The screenshot shows the 'Microsoft.Web-WebApp-Portal-1a7942a8-8d9e - Overview' page. On the left, there's a navigation sidebar with 'Overview', 'Inputs', 'Outputs', and 'Template'. The main area displays a green checkmark icon and the message 'Your deployment is complete'. Below this, it shows deployment details: Deployment name: Microsoft.Web-WebApp-Portal-1a7942a8-8d9e, Start time: 11/14/2019, 9:45:40 PM, Subscription: Visual Studio Enterprise – MPN, Correlation ID: 6c1d0f77-207d-49d9-b08d-a14b6b072, and Resource group: patsam-rg. There are also sections for 'Deployment details' (with a download link) and 'Next steps' (with a 'Go to resource' button).

We can see the newly created App Service.

The screenshot shows the 'App Services' blade. At the top, there are buttons for '+ Add', 'Edit columns', 'Refresh', 'Export to CSV', 'Assign tags', 'Start', 'Restart', 'Stop', 'Delete', 'Feedback', and 'Leave preview'. Below this is a search bar and filter options for 'Filter by name...', 'Subscription == all', 'Resource group == all', 'Location == all', and 'Add filter'. A dropdown menu shows 'No grouping'. The table below lists one record: 'patsam' (Status: Running, Location: Central US, Pricing Tier: Free, App Service Plan: patsam-sp, Subscription: Visual Studio Enterpri..., App Type: Web App).

Next go inside the newly created App Service. Click on it.

The screenshot shows the 'App Services' blade in the Azure portal. It lists one record: 'patsam'. The details are as follows:

| Name | Status | Location | Pricing Tier | App Service Plan | Subscription | App Type |
|--------|---------|------------|--------------|------------------|--------------------------------|----------|
| patsam | Running | Central US | Free | patsam-sp | Visual Studio Enterprise - MPN | Web App |

Click on Browse button to check whether we have created the App Service successfully.

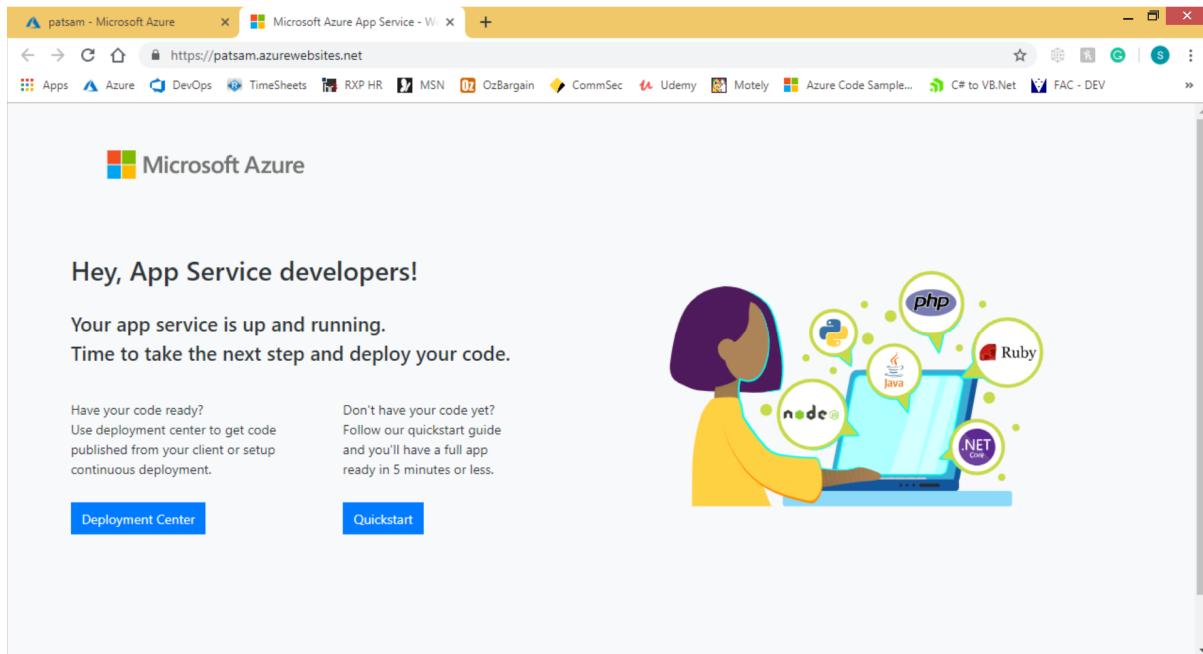
The screenshot shows the 'Overview' page for the 'patsam' app service. Key details displayed include:

- Resource group: patsam-rg
- Status: Running
- Location: Central US
- Subscription: Visual Studio Enterprise - MPN
- URL: https://patsam.azurewebsites.net
- App Service Plan: patsam-sp (F1: Free)
- FTP/deployment username: patsam\deploy-patrick-sameera
- FTP hostname: ftp://waws-prod-dm1-155.ftp.azurewebsites.windows.net
- FTPS hostname: https://waws-prod-dm1-155.ftp.azurewebsites.windows.net

On the left sidebar, the 'Overview' tab is selected. Other tabs include Activity log, Access control (IAM), Tags, Diagnose and solve problems, Security, Deployment (Quickstart, Deployment slots, Deployment Center), Settings, and Configuration.

We can see it's working fine.

- <https://patsam.azurewebsites.net/>



Next go to the App Service Plan we created. Click on it.

Home > App Service plans

App Service plans

RXP Services Ltd

+ Add Edit columns Refresh Assign tags

Subscriptions: Visual Studio Enterprise – MPN – Don't see a subscription? [Open Directory + Subscription settings](#)

Filter by name... All resource groups All locations All tags No grouping

1 items

| Name | Apps | Pricing Tier | Resource group | Subscription | ... |
|-----------|------|----------------------|----------------|--------------------------------|-----|
| patsam-sp | 1 | patsam-sp (F1: Free) | patsam-rg | Visual Studio Enterprise – MPN | ... |

Under Apps tab now we can see the App Service (Web App) we created earlier.

Home > App Service plans > patsam-sp - Apps

patsam-sp - Apps App Service plan

Search (Ctrl+ /)

Overview Activity log Access control (IAM) Tags Diagnose and solve problems

Settings

Apps

Name Type Resource Group Status

| Name | Type | Resource Group | Status |
|--------|------|----------------|---------|
| patsam | app | patsam-rg | Running |

Step 04: Publish the code to Azure through FTP

Click on Deployment Centre tab under the App Service we created earlier.

The screenshot shows the Azure Deployment Center interface for an app service named 'patsam'. The left sidebar has 'Deployment Center' selected under the 'Deployment' section. The main area displays four deployment methods: OneDrive, Dropbox, External, and FTP. The 'FTP' option is highlighted with a blue border, indicating it is the active choice.

Select FTP and click Dashboard.

This screenshot is similar to the previous one, but the 'FTP' option is now highlighted with a blue border. A blue rectangular box highlights the 'Dashboard' button at the bottom right of the main content area.

It will provide you the FTP Credential details.

- Endpoint URL: `ftps://waws-prod-dm1-155.ftp.azurewebsites.windows.net/site/wwwroot`
- Username: `patsam\$patsam`
- Password: `***`

The screenshot shows the 'FTP' section of the Azure App Service configuration. It displays the following information:

- FTPS Endpoint:** `ftps://waws-prod-dm1-155.ftp.azurewebsites.windows.net` (with a `Copy` button)
- App Credentials:** This tab is selected, showing:
 - Username:** `patsam\$patsam` (with a `Copy` button)
 - Password:** A masked password field (with a `Show` and `Copy` button).
- User Credentials:** This tab is unselected.

Also, we can get Publish Profile details through Overview tab of the App Service.

Click on Get Publish Profile button to download the profile details.

The screenshot shows the Azure App Service Overview page for the 'patsam' app service. In the top right, there are buttons for Browse, Stop, Swap, Restart, Delete, Get publish profile, and Reset publish profile. The 'Get publish profile' button is highlighted. The main content area displays various deployment and monitoring settings. A callout box points to the 'Get publish profile' button with the text: 'Click here to access Application Insights for monitoring and profiling for your ASP.NET Core app. →'. Below this, it shows the Resource group (patsam-rg), Status (Running), Location (Central US), Subscription (Visual Studio Enterprise – MPN), Subscription ID (3822b9c0-bf7e-455e-a1d1-8717b1b4150b), and Tags (Click here to add tags). On the left sidebar, there are sections for Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Security, Deployment (Quickstart, Deployment slots, Deployment Center), and Settings (Configuration). At the bottom, there are three cards: 'Diagnose and solve problems' (Our self-service diagnostic and troubleshooting experience helps you), 'Application Insights' (Application insights helps you detect and diagnose quality issues in your apps, and), and 'App Service Advisor' (App Service Advisor provides insights for improving app experience on the App).

Once the download completes open the file.

- Endpoint URL: `ftps://waws-prod-dm1-155.ftp.azurewebsites.windows.net/site/wwwroot`
- Username: `patsam\$patsam`
- Password: `***`

The screenshot shows the contents of a Notepad++ file named 'patsam.PublishSettings'. The file contains XML code for a publish profile. The XML includes details such as the publish method (MSDeploy), publish URL (patsam.scm.azurewebsites.net:443), msdeploy site name ('patsam'), user name ('\$patsam'), user password ('\$hRcwC9eGXhQ26RtJpg2Ycd29E4mR6zHJ9eqRg4E689anQcpbQsvQjAhk'), destination app URL ('http://patsam.azurewebsites.net'), SQL Server DB connection string, hosting provider forum link, and control panel link. It also defines database publishing profiles for both 'patsam - FTP' and 'patsam - Read Only - FTP' methods, specifying their respective publish URLs, user names, user passwords, and connection strings. The code is color-coded for readability.

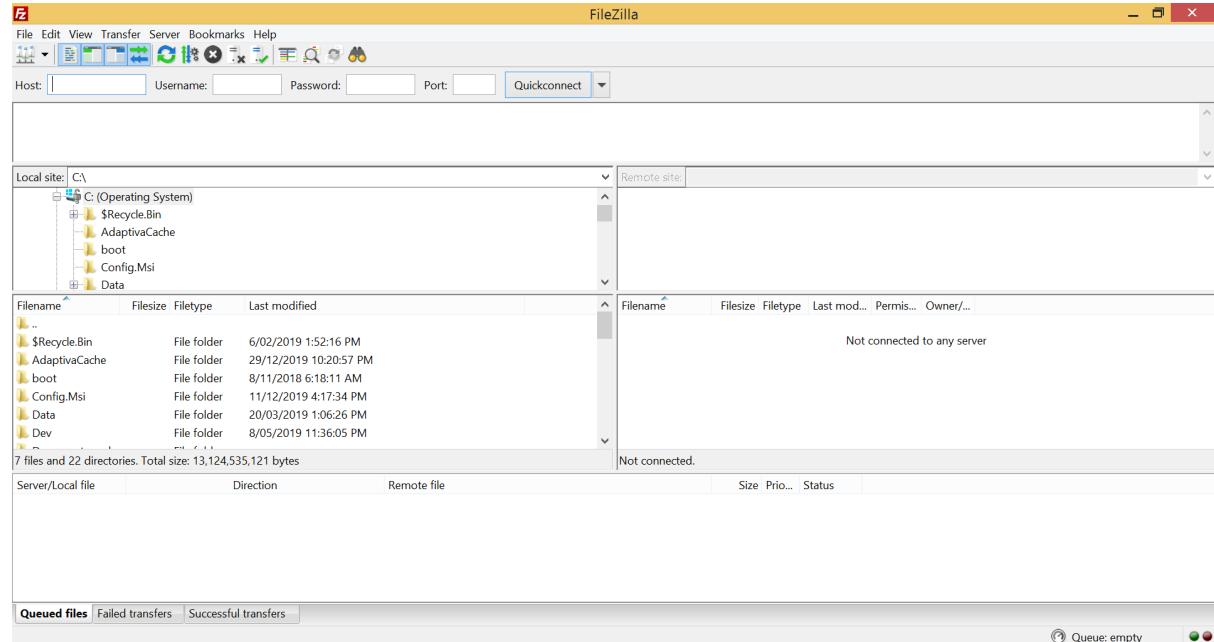
```
<publishData><publishProfile profileName="patsam - Web Deploy" publishMethod="MSDeploy" publishUrl="patsam.scm.azurewebsites.net:443" msdeploySite="patsam" userName="$patsam" userPWD="$hRcwC9eGXhQ26RtJpg2Ycd29E4mR6zHJ9eqRg4E689anQcpbQsvQjAhk" destinationAppUrl="http://patsam.azurewebsites.net" SQLServerDBConnectionString="" mySQLDBConnectionString="" hostingProviderForumLink="" controlPanelLink="http://windows.azure.com" webSystem="WebSites"></publishData></publishProfile><publishProfile profileName="patsam - FTP" publishMethod="FTP" publishUrl="ftps://waws-prod-dm1-155.ftp.azurewebsites.windows.net/site/wwwroot" ftpPassiveMode="True" userName="patsam\$patsam" userPWD="jhRcwC9eGXhQ26RtJpg2Ycd29E4mR6zHJ9eqRg4E689anQcpbQsvQjAhk" destinationAppUrl="http://patsam.azurewebsites.net" SQLServerDBConnectionString="" mySQLDBConnectionString="" hostingProviderForumLink="" controlPanelLink="http://windows.azure.com" webSystem="WebSites"></publishProfile></publishProfile><publishProfile profileName="patsam - Read Only - FTP" publishMethod="FTP" publishUrl="ftps://waws-prod-dm1-155.ftp.azurewebsites.windows.net/site/wwwroot" ftpPassiveMode="True" userName="patsam\$patsam" userPWD="jhRcwC9eGXhQ26RtJpg2Ycd29E4mR6zHJ9eqRg4E689anQcpbQsvQjAhk" destinationAppUrl="http://patsam.azurewebsites.net" SQLServerDBConnectionString="" mySQLDBConnectionString="" hostingProviderForumLink="" controlPanelLink="http://windows.azure.com" webSystem="WebSites"></publishProfile></publishProfile>
```

Next, we need to use an FTP Client to connect to Azure FTP location. I'll be using FileZilla as my FTP Client.

You can download FileZilla here:

<https://filezilla-project.org/download.php?type=server>

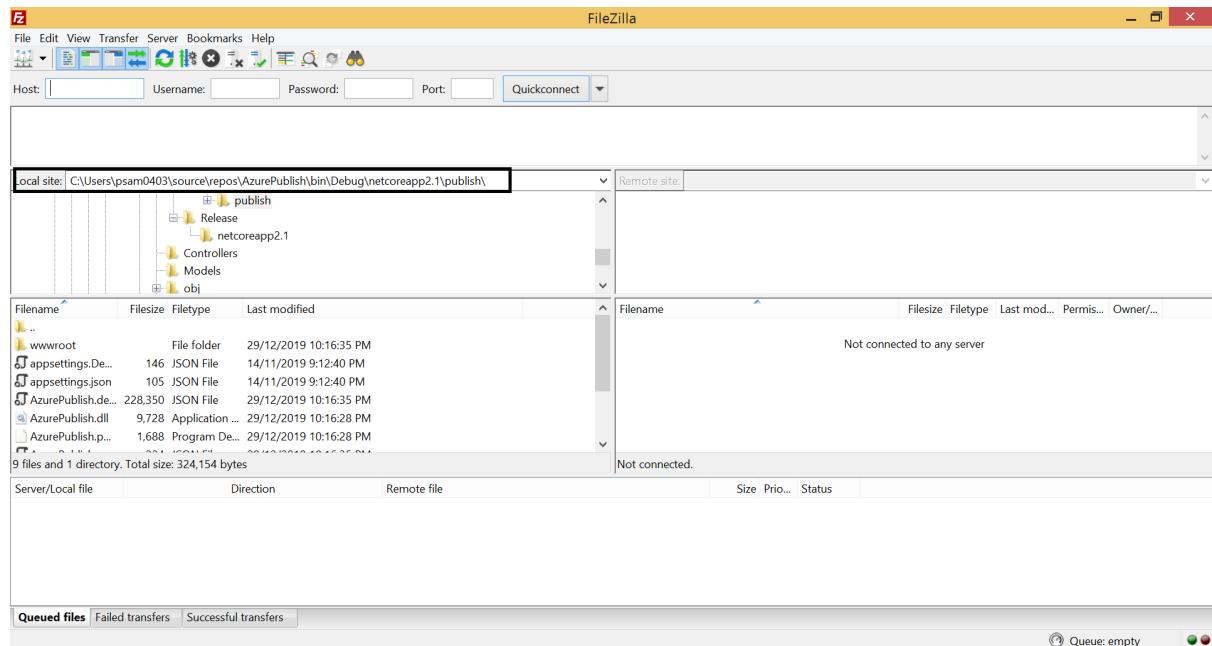
Open FileZilla.



Input Local Site and Remote Site details.

Local Site:

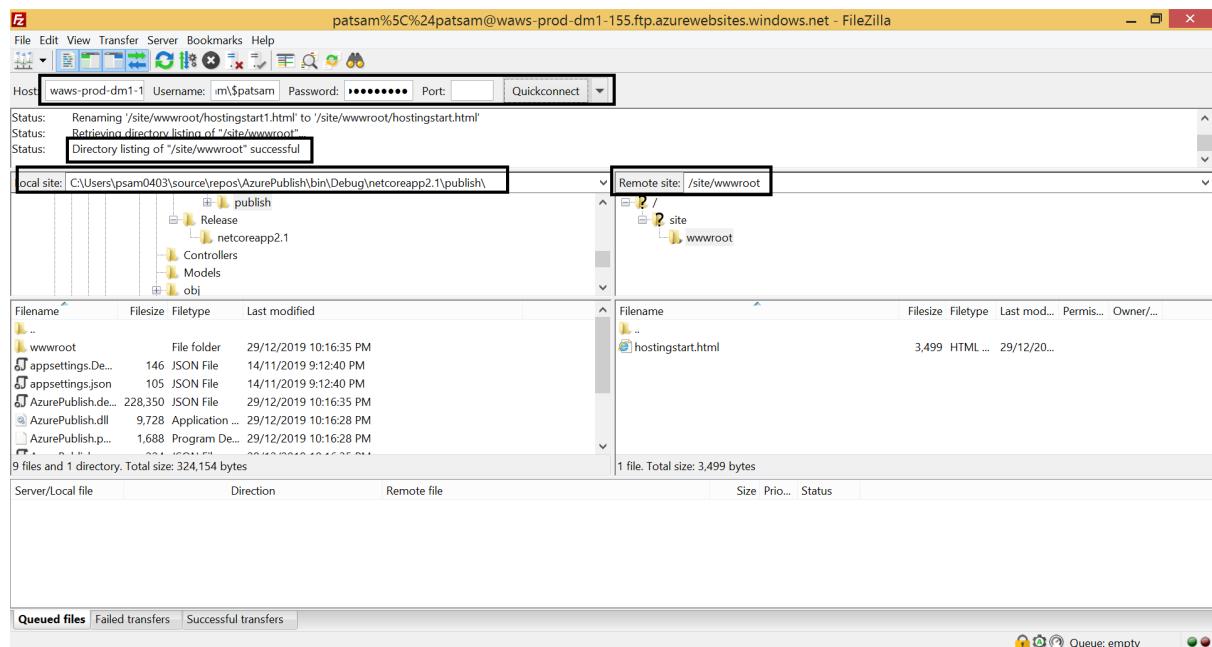
- C:\Users\psam0403\source\repos\AzurePublish\bin\Debug\netcoreapp2.1\publish



Remote Site:

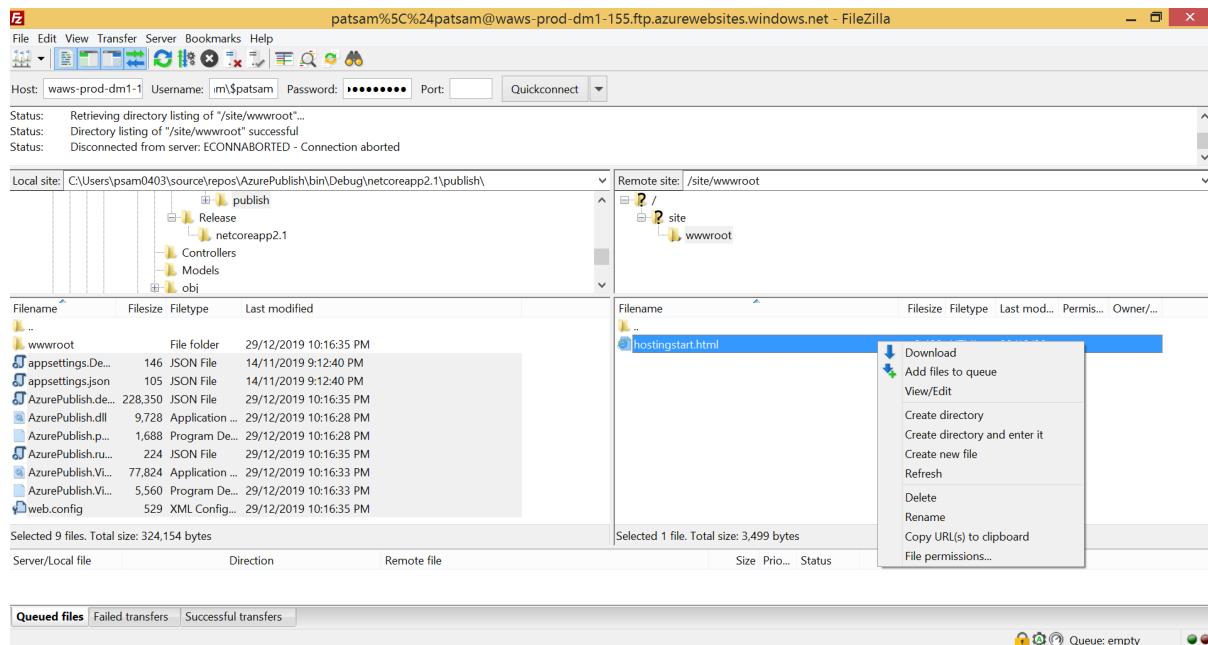
- Endpoint URL: `ftps://waws-prod-dm1-155.ftp.azurewebsites.windows.net/site/wwwroot`
- Username: patsam\\$patsam
- Password: ***

Click QuickConnect.

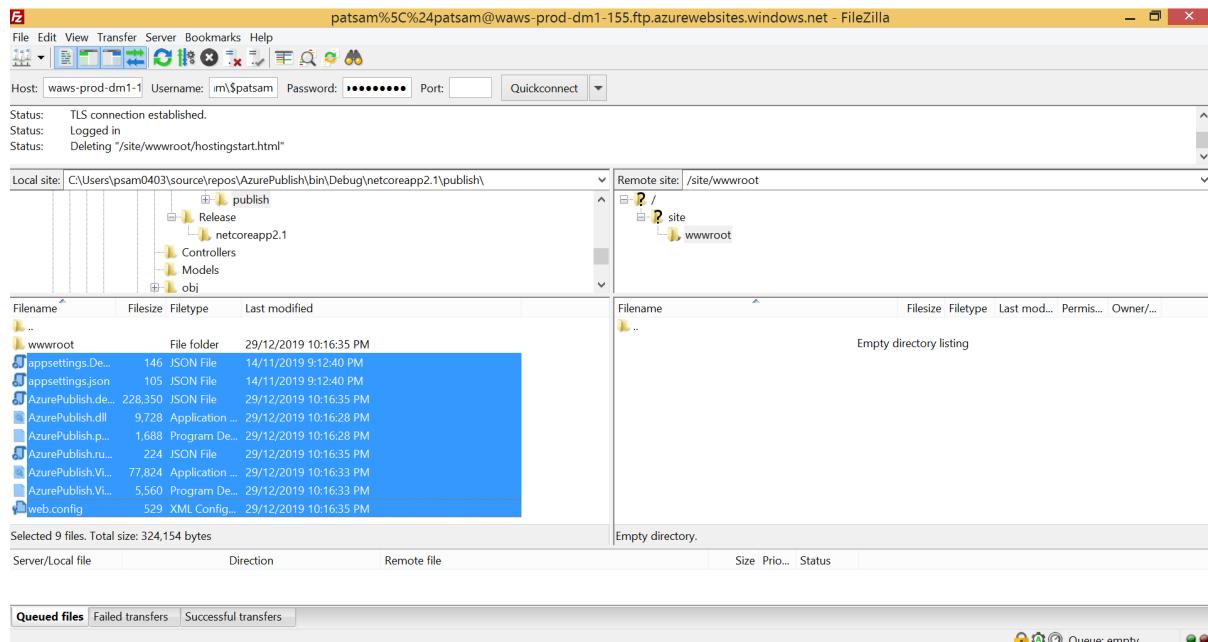


Patrick Sameera Jayalath

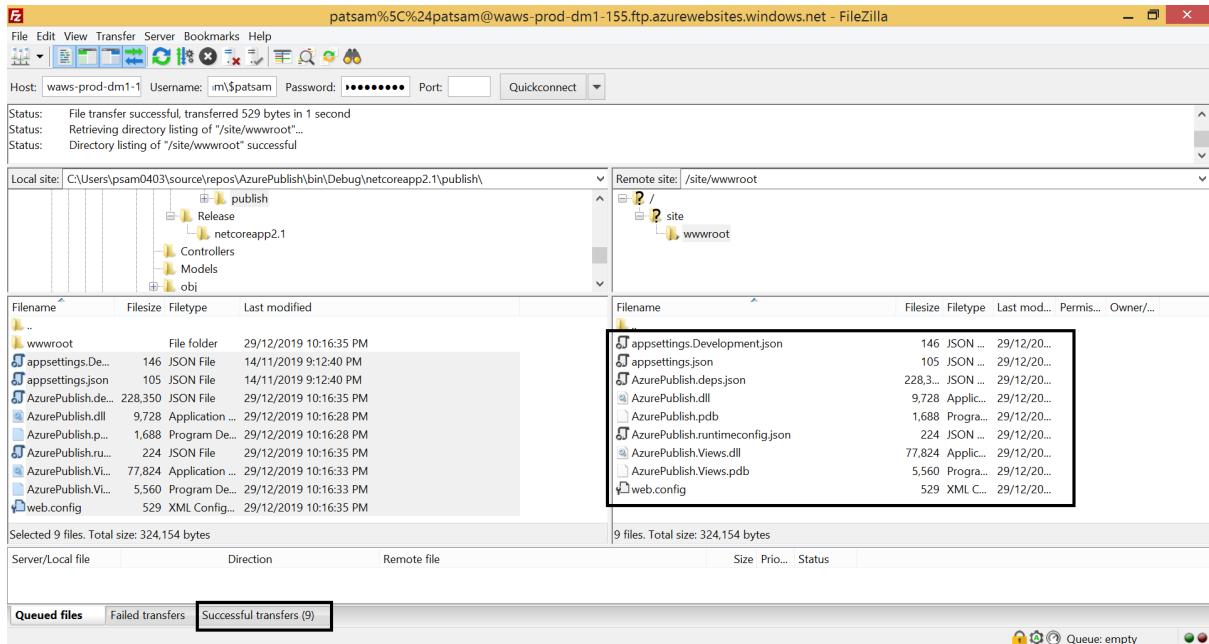
Right click on the hostingstart.html file and delete it.



Next select all the files from the Local Site and drag them to Remote Site.



Make sure file transfer completes successfully.



Browse the URL:

- <https://patsam.azurewebsites.net/>

The screenshot shows a web browser window with the URL 'https://patsam.azurewebsites.net/' in the address bar. The page content is the 'AzurePublish' application, which includes sections for 'Application uses', 'How to', 'Overview', and 'Run & Deploy'. The 'Overview' section contains a bulleted list of topics such as 'Conceptual overview of what is ASP.NET Core', 'Fundamentals of ASP.NET Core such as Startup and middleware.', and 'Working with Data'. The 'Run & Deploy' section also lists items like 'Run your app' and 'Publish to Microsoft Azure Web Apps'. At the bottom of the page, there is a copyright notice: '© 2019 - AzurePublish'.