

Web Apps using VS and Azure

Lab 03

by

Andrea Hatch, Nishava Inc.

Deep Azure @McKesson

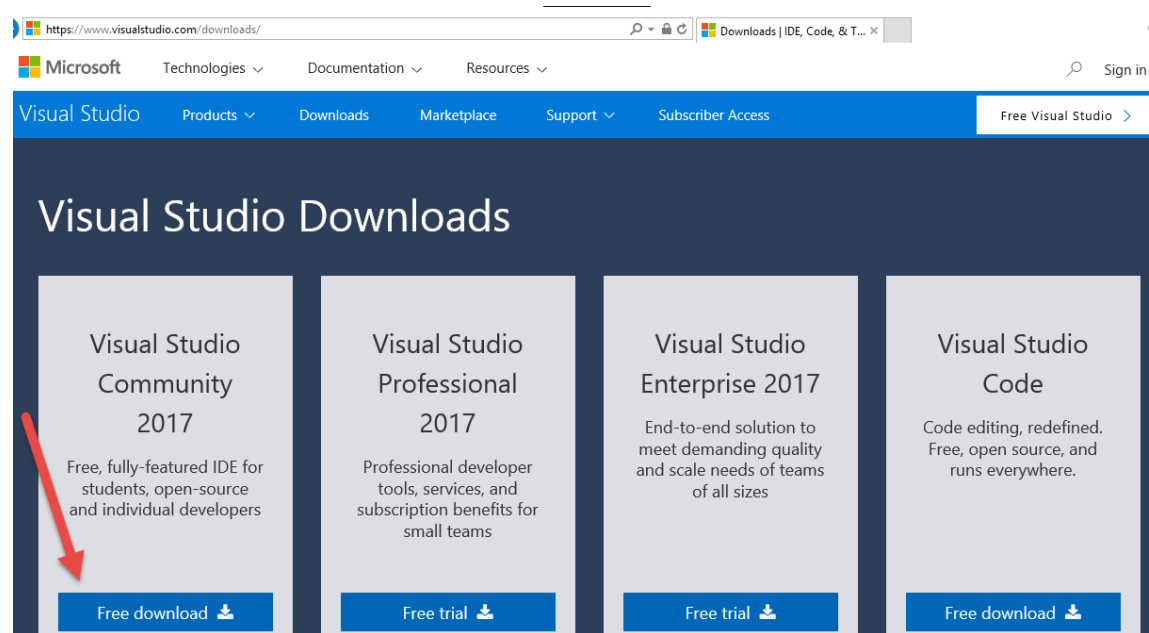
Overview

- Download Visual Studio and connect with Azure
- Add Ons, Extensions and Updates
- Create a web app in VS to deploy to Azure
- Create a web app in Azure and Update the web app in VS

1. Download Visual Studio and connect with Azure

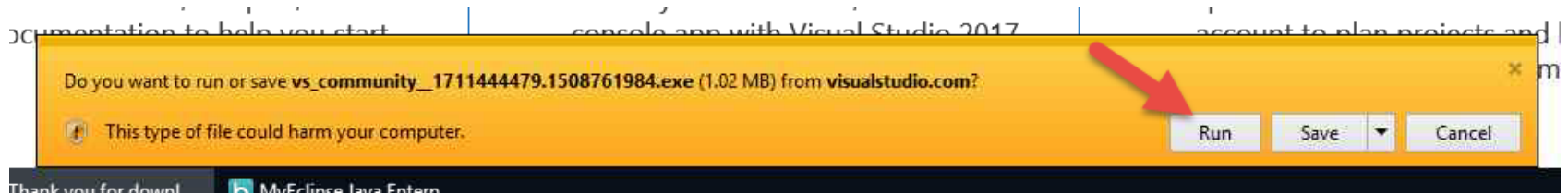
Download Visual Studio

- <https://www.visualstudio.com/downloads/>
- Select the Free download under Visual Studio Community 2017
- Note: If you already have an older version of VS that will work too



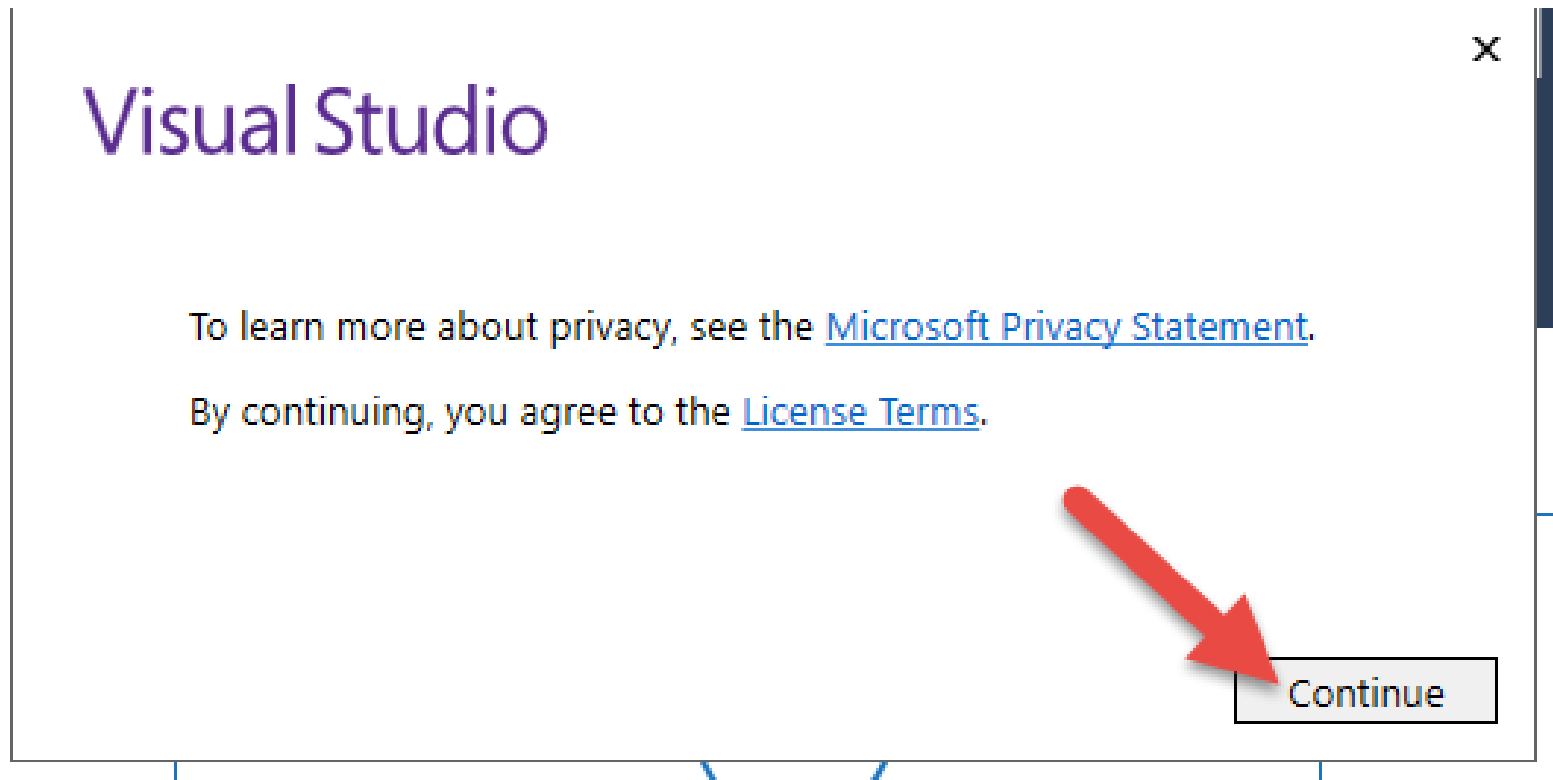
Download Visual Studio

- Select 'Run' on the pop-up to install Studio



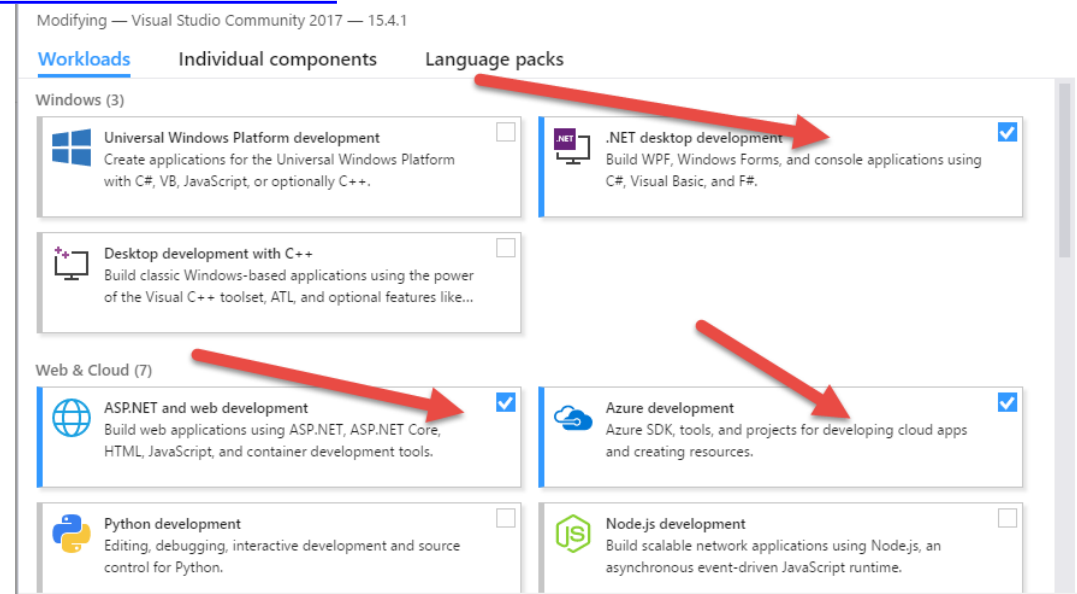
Download Visual Studio

- Agree to the licensing terms by selecting continue



Download Visual Studio

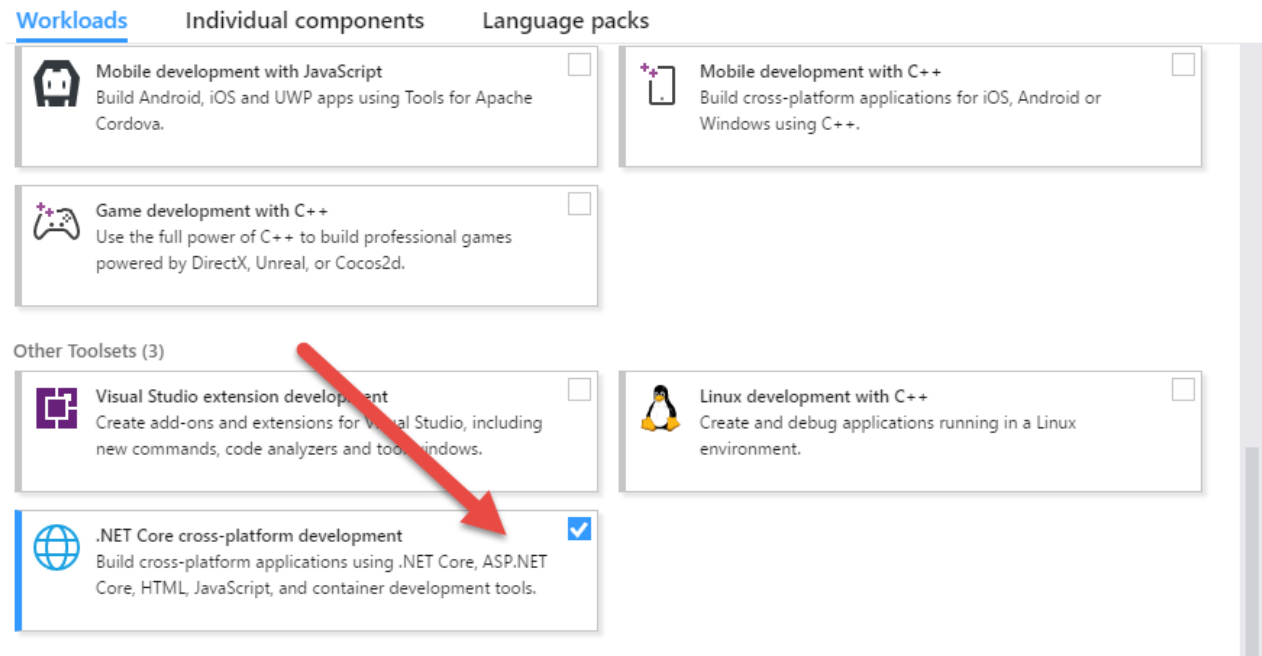
- On 2017 - you will then be asked if you would like to add any workloads to your VB install, select the .NET desktop development, ASP.NET and web development and Azure development then Scroll down
- If you have 2015 or 2013 already you will need to install the respective SDK's for Azure
 - 2015: <http://go.microsoft.com/fwlink/?linkid=518003>
 - 2013: <http://go.microsoft.com/fwlink/?LinkID=324322>



Download Visual Studio

- Under Other Toolsets make sure to also select .NET Core cross-platform development
- Note: if you miss selecting anything you can always add them later by Tools – Get Tools and Features

Modifying — Visual Studio Community 2017 — 15.4.1



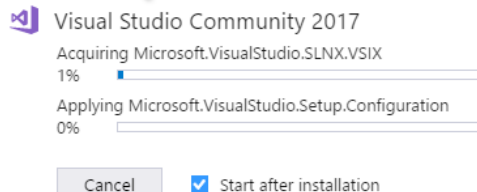
Download Visual Studio

- Your download may take a while to complete and should look similar to this.

Visual Studio

Products

Installed



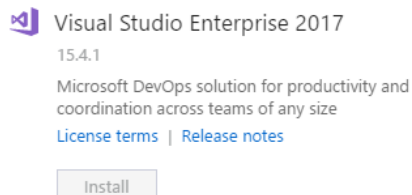
Visual Studio Community 2017

Acquiring Microsoft.VisualStudio.SLNX.VSIX
1%

Applying Microsoft.VisualStudio.Setup.Configuration
0%

☒ Start after installation

Available

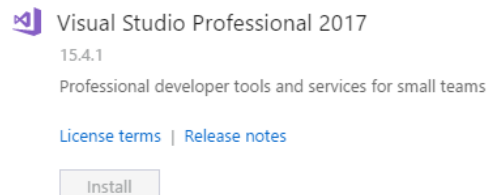


Visual Studio Enterprise 2017

15.4.1

Microsoft DevOps solution for productivity and coordination across teams of any size

[License terms](#) | [Release notes](#)



Visual Studio Professional 2017

15.4.1

Professional developer tools and services for small teams

[License terms](#) | [Release notes](#)

Welcome!

We invite you to go online to hone your skills and find additional tools to support your development workflow.

Learn

Whether you're new to development or an experienced developer, we have you covered with our tutorials, videos, and sample code.

Marketplace

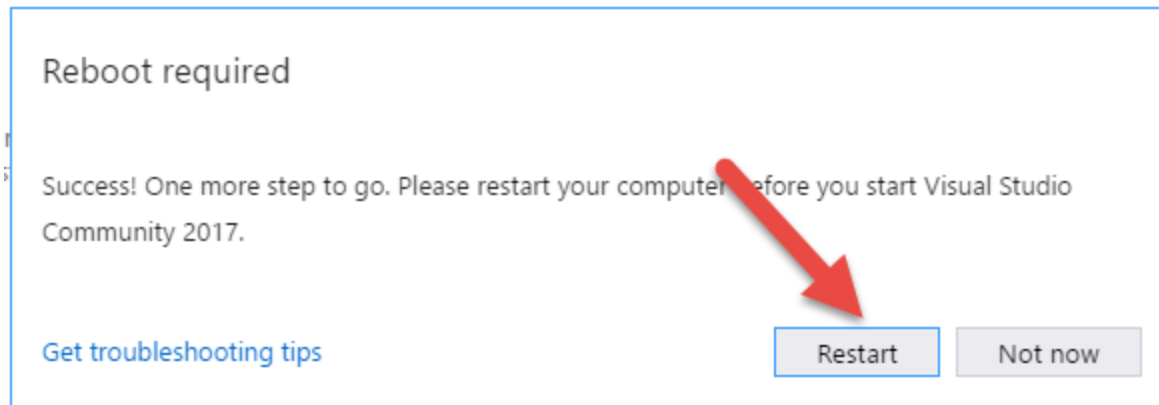
Use Visual Studio extensions to add support for new technologies, integrate with other products and services, and fine-tune your experience.

Need some help?

Check out the [Microsoft Developer Community](#) where developers provide feedback and answers to many common problems.

Download Visual Studio

- You may be asked to restart before you can use Visual Studio once the installation is complete.



Connect Azure and VS

- On your first log in of VS you can connect your Azure account

Visual Studio

Welcome!

Connect to all your developer services.

Sign in to start using your Azure credits, publish code to a private Git repository, sync your settings, and unlock the IDE.

[Learn more](#)



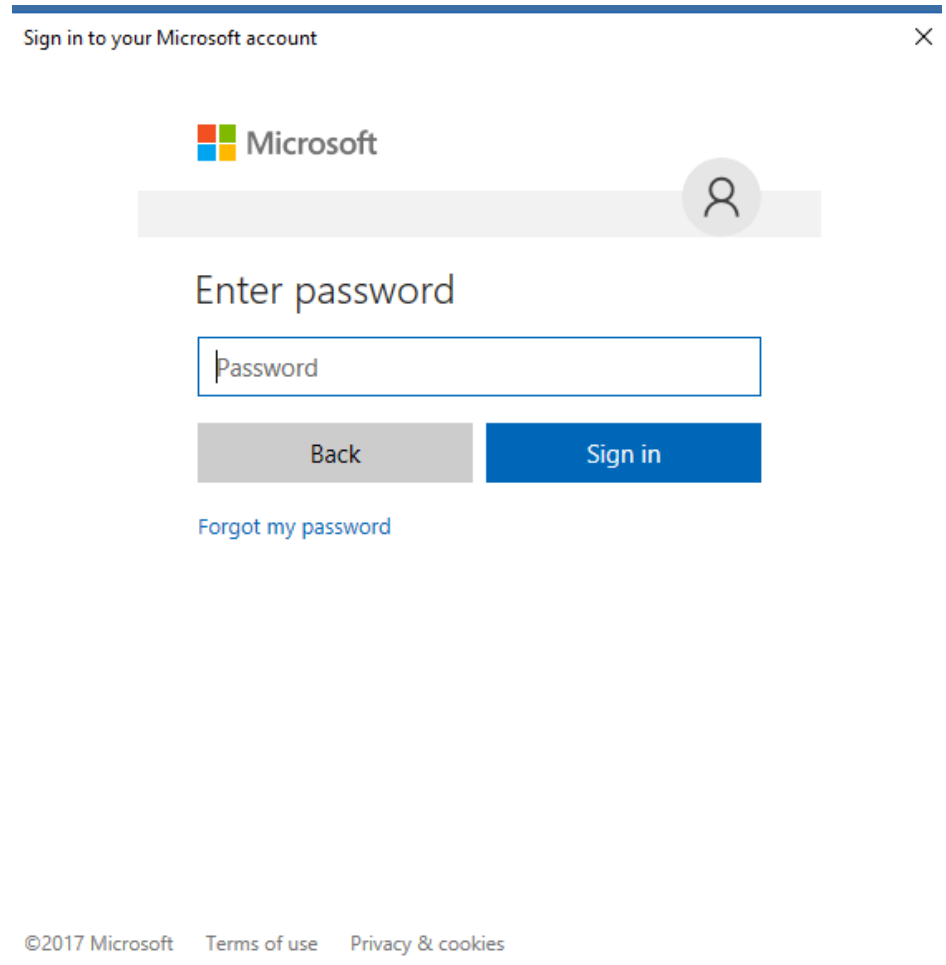
Sign in

Don't have an account? [Sign up](#)

[Not now, maybe later.](#)

Connect Azure and VS

- You should recognize this screen from signing in to your Azure account in previous homeworks



Sign in to your Microsoft account

Microsoft

Enter password

Password

Back Sign in

[Forgot my password](#)

©2017 Microsoft Terms of use Privacy & cookies

Connect Azure and VS

- You can confirm you are connected and then pick a theme for your VS and then select 'Start Visual Studio'

Visual Studio

Hello,

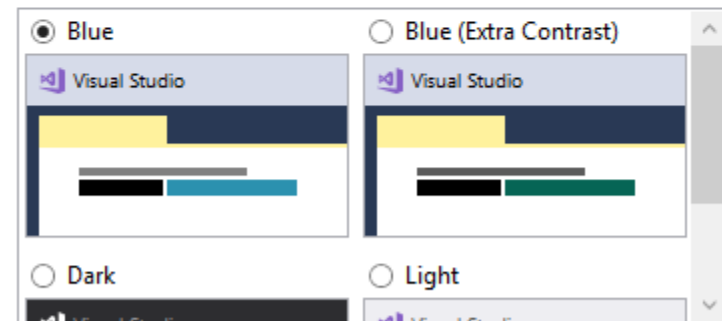


[View your Visual Studio profile](#)

Start with a familiar environment

Development Settings: General

Choose your color theme



You can always change these settings later.

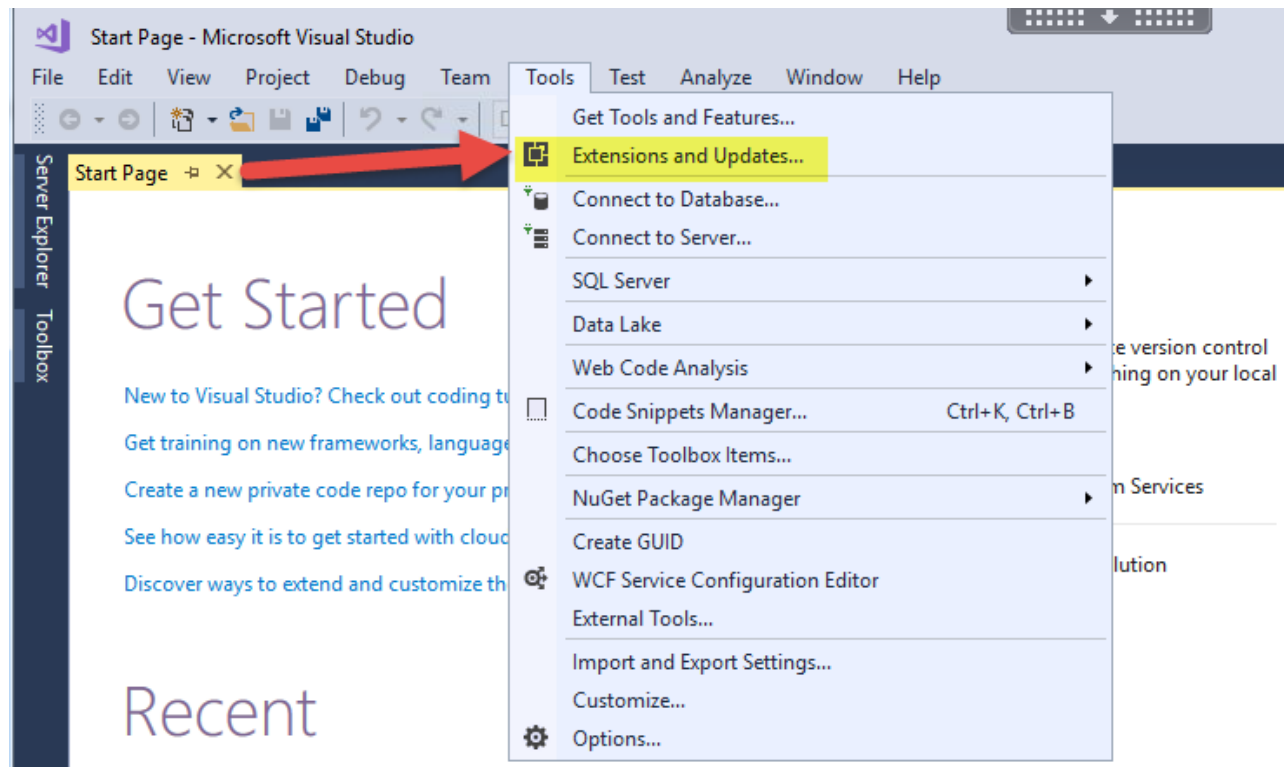
2. Add On's, Extensions and Updates

Add On's to Download

- You will need to download the following in order to work with Git and .NET
 - GIT: <https://git-scm.com/downloads>
 - .NET Core 2.0:
<https://download.microsoft.com/download/F/6/E/F6ECBBCC-B02F-424E-8E03-D47E9FA631B7/DotNetCore.1.0.1-VS2015Tools.Preview2.0.3.exe>

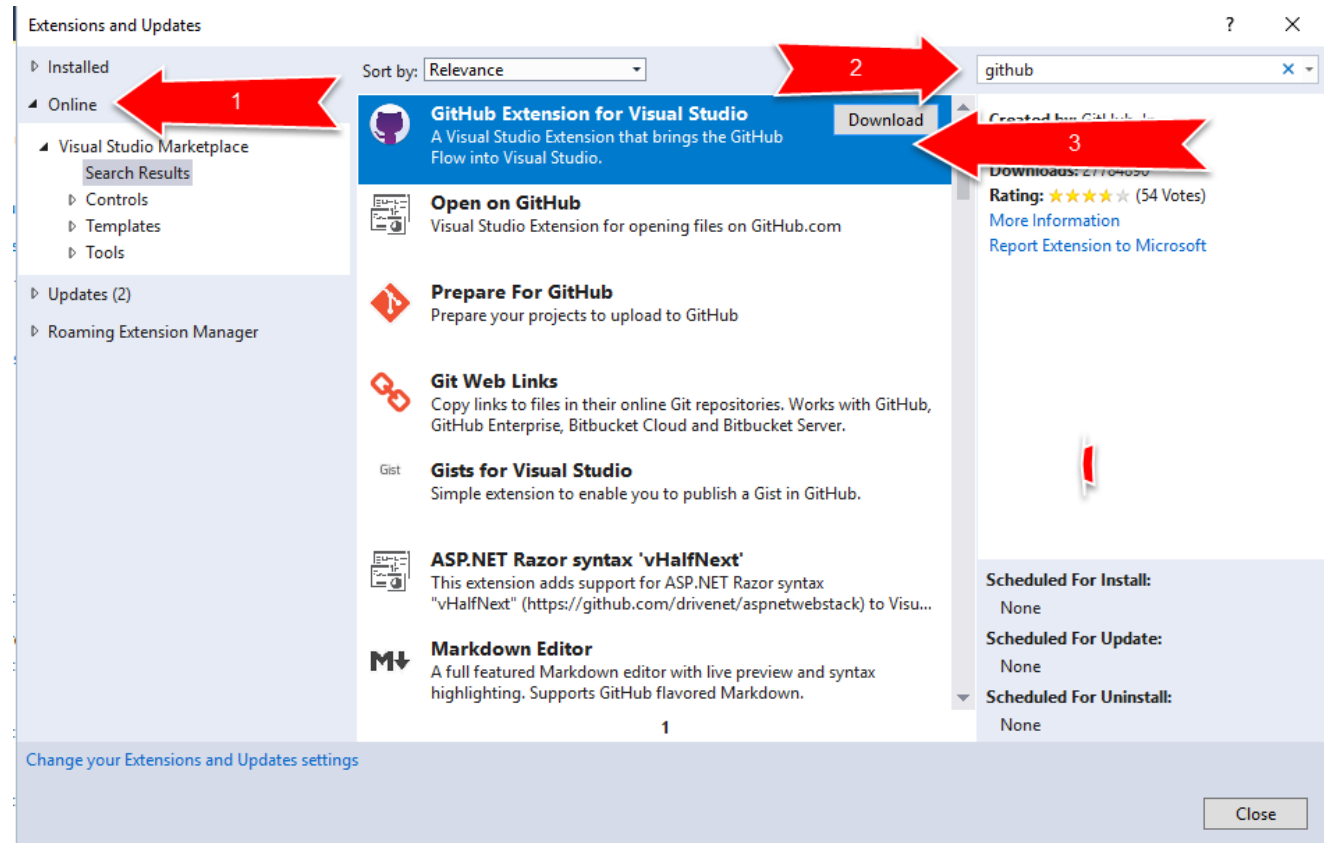
Add the GitHub Extension to VS

- In VS, under Tools select 'Extensions and Updates'



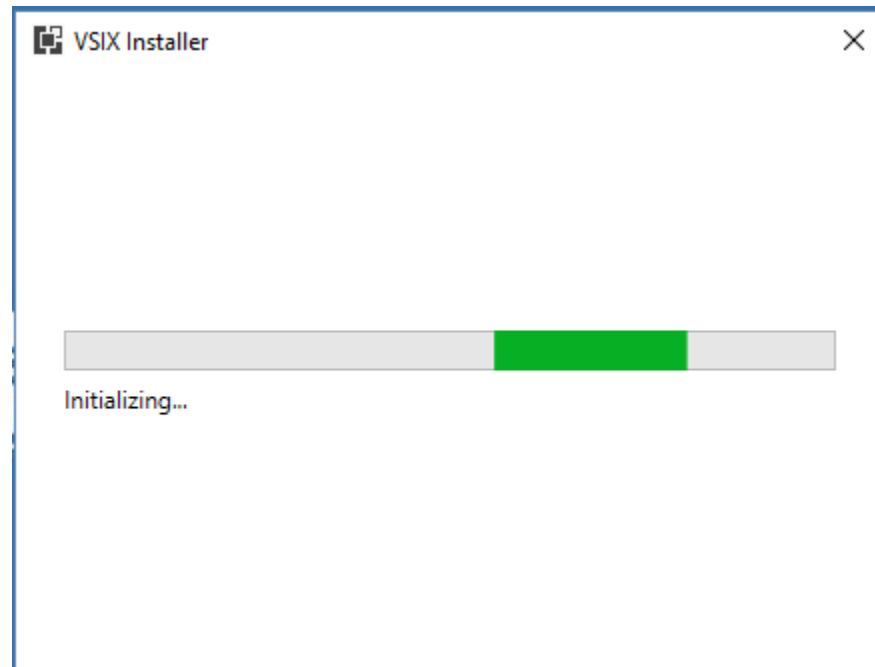
Add the GitHub Extension to VS

- Under Online search for github and then download it



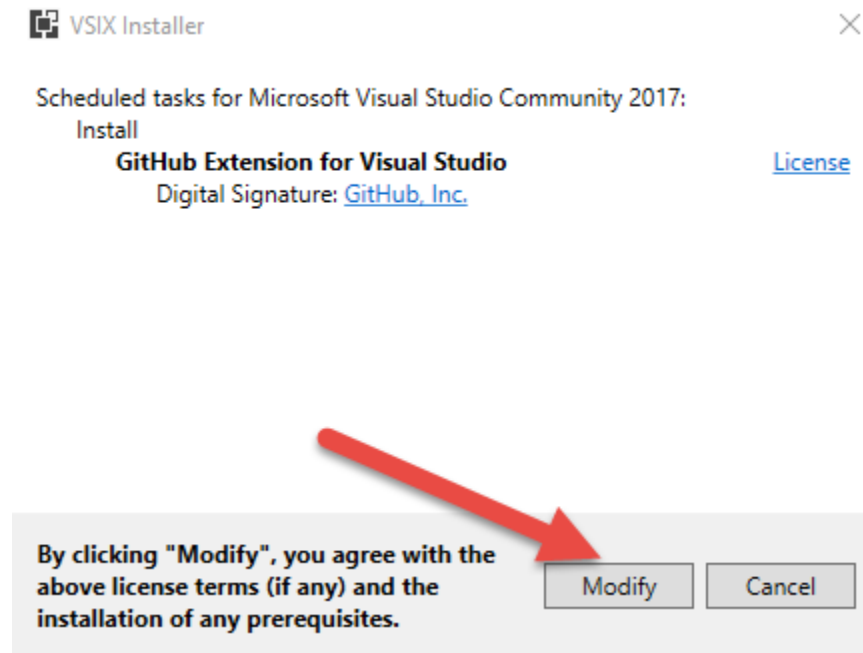
Add the GitHub Extension to VS

- You must close your VS and the installation should begin



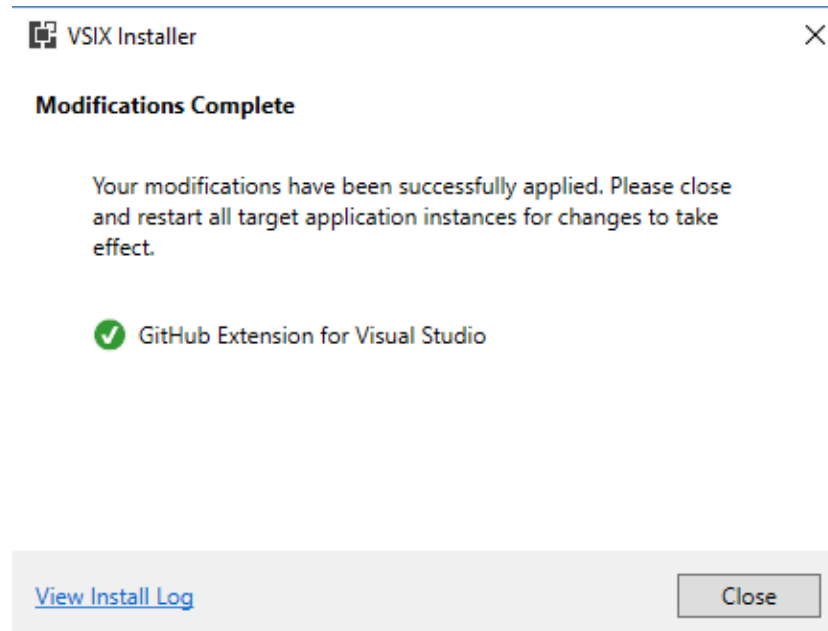
Add the GitHub Extension to VS

- Select 'Modify' so that the installation can update your VS



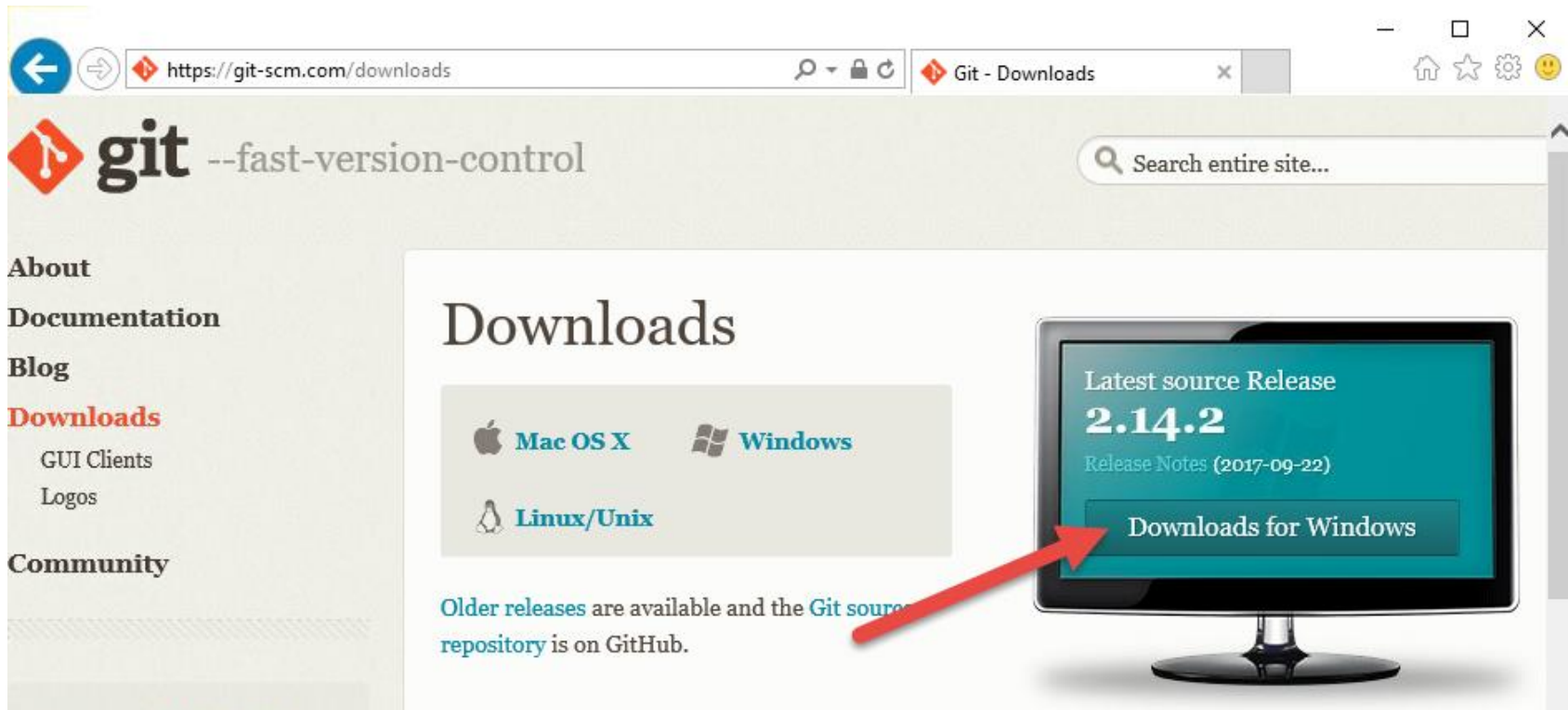
Add the GitHub Extension to VS

- Your installation will be complete once you receive a message like the following. Close this and open VS again.



Download Git

- <https://git-scm.com/downloads>
- Select 'Downloads for Windows' and scroll trough keeping the default settings



Download Git

- Select 'Run' to start your download or select from the other downloads if you do not have a 64 bit machine



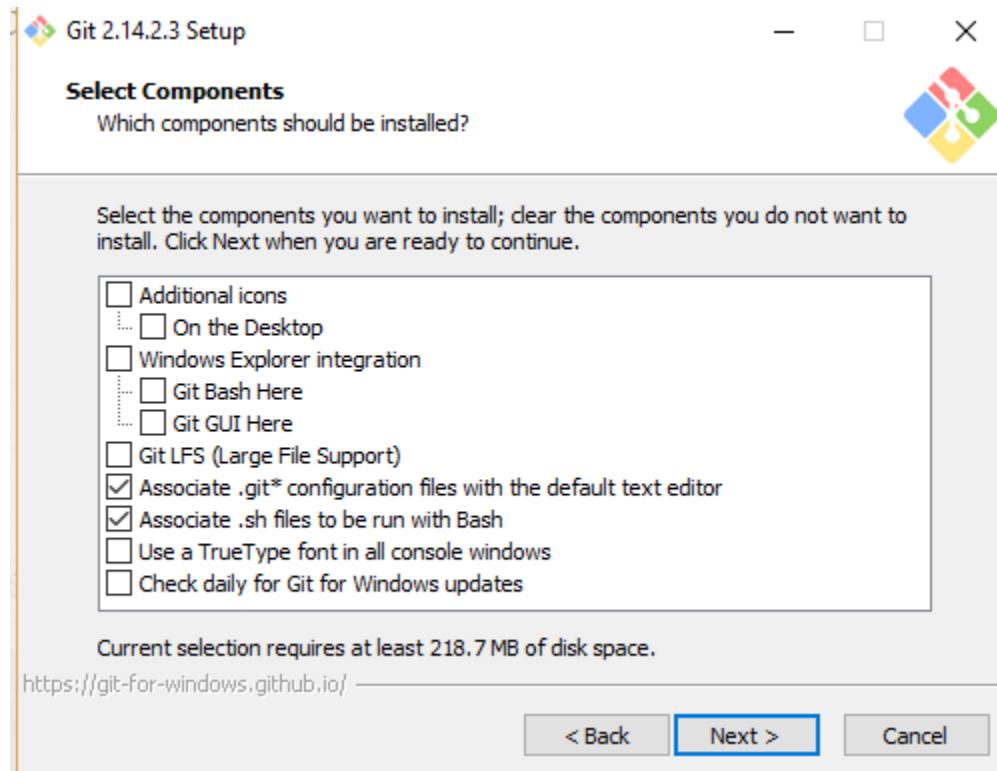
Download Git

- Keep defaults and select 'Next'



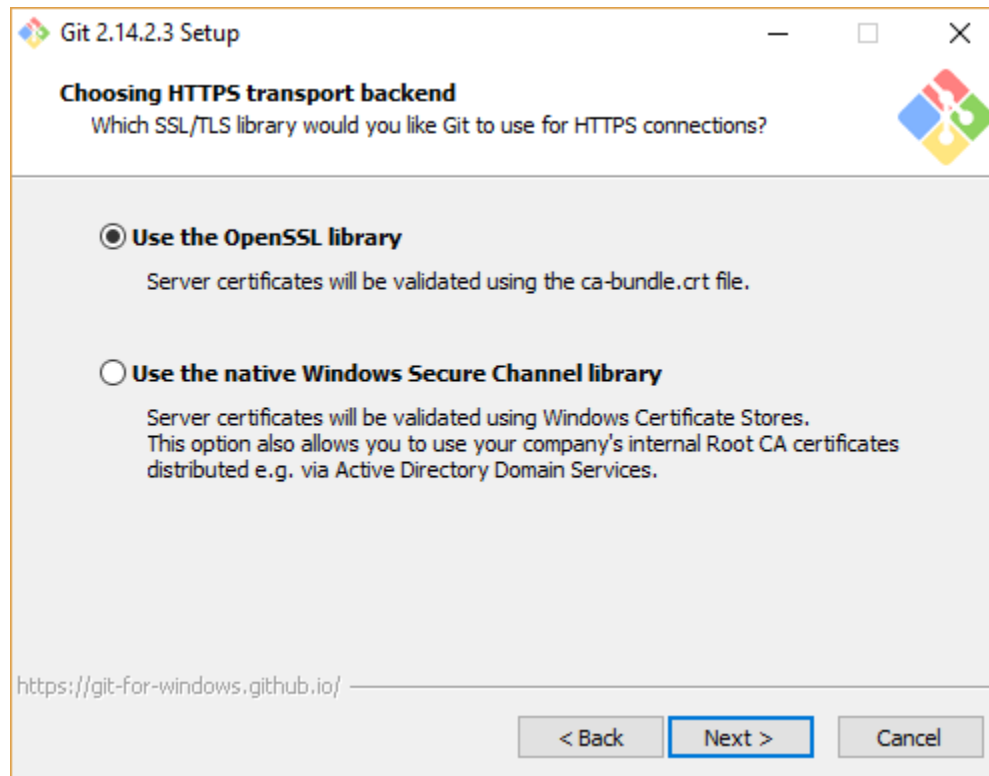
Download Git

- Keep defaults and select 'Next'



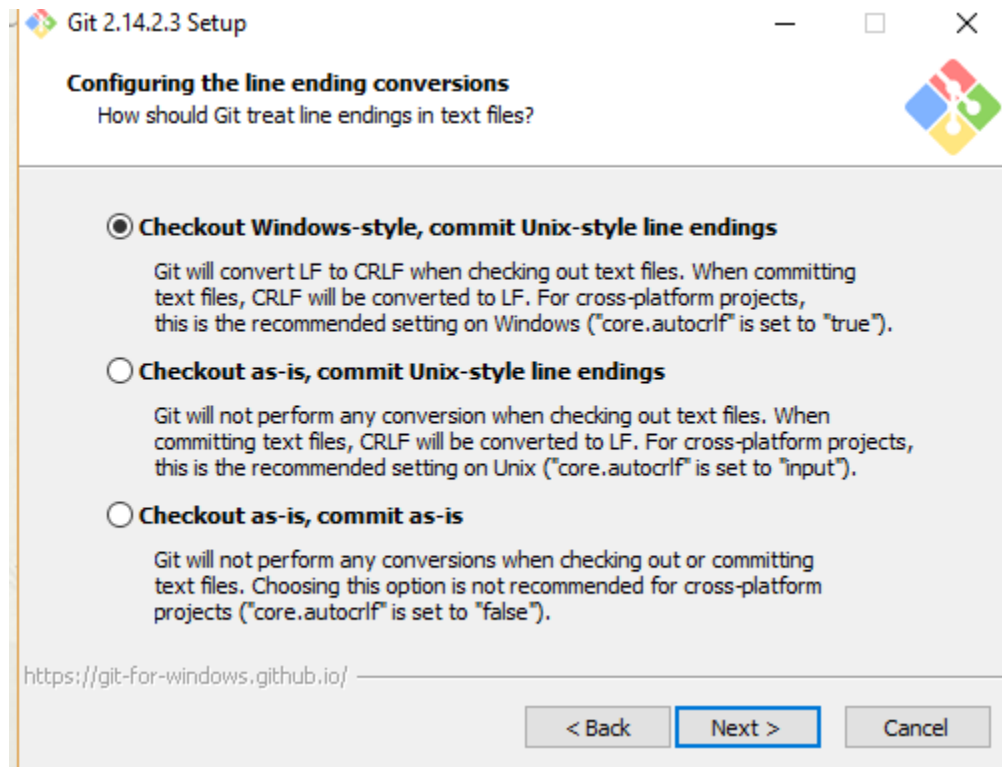
Download Git

- Keep defaults and select 'Next'



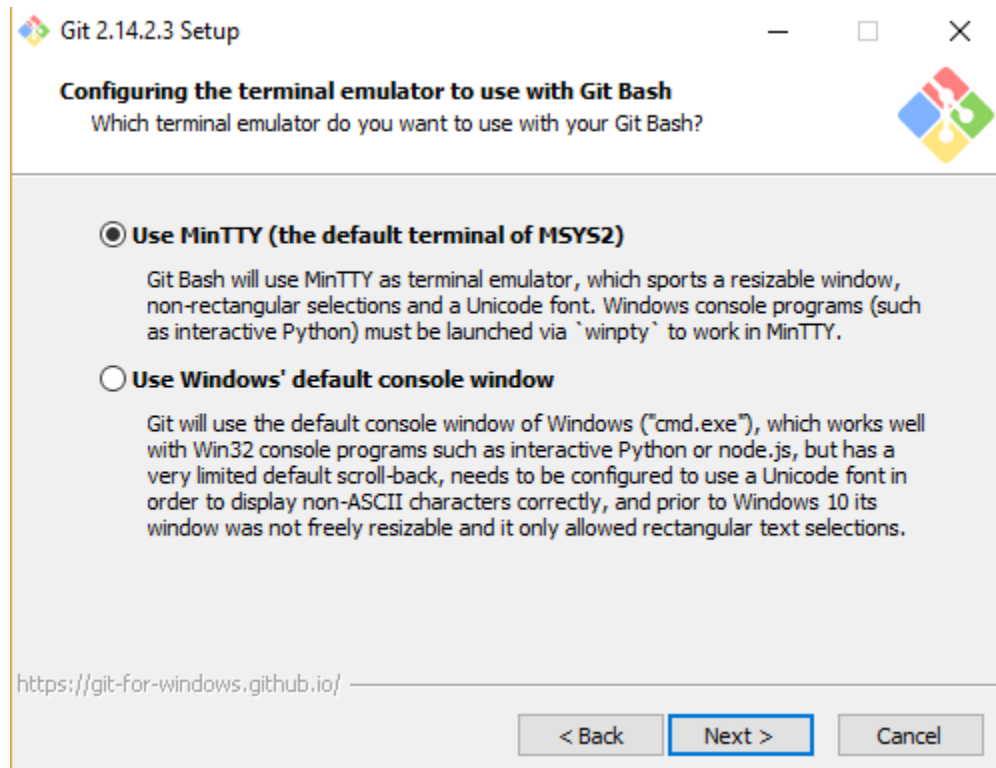
Download Git

- Keep defaults and select 'Next'



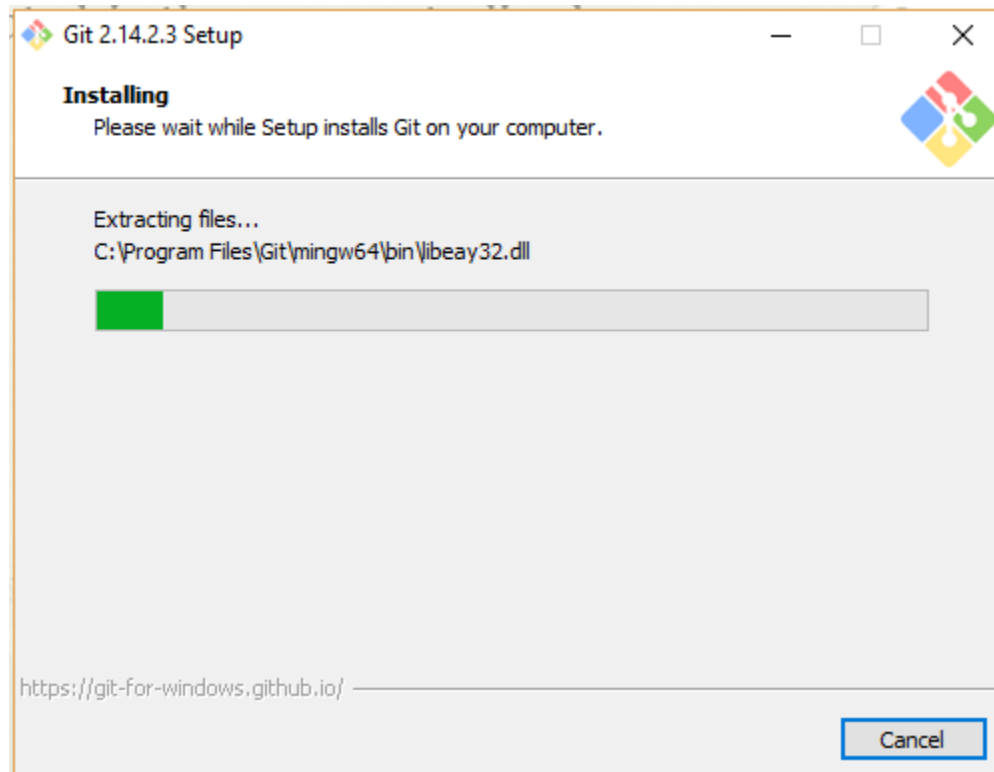
Download Git

- Keep defaults and select 'Next'



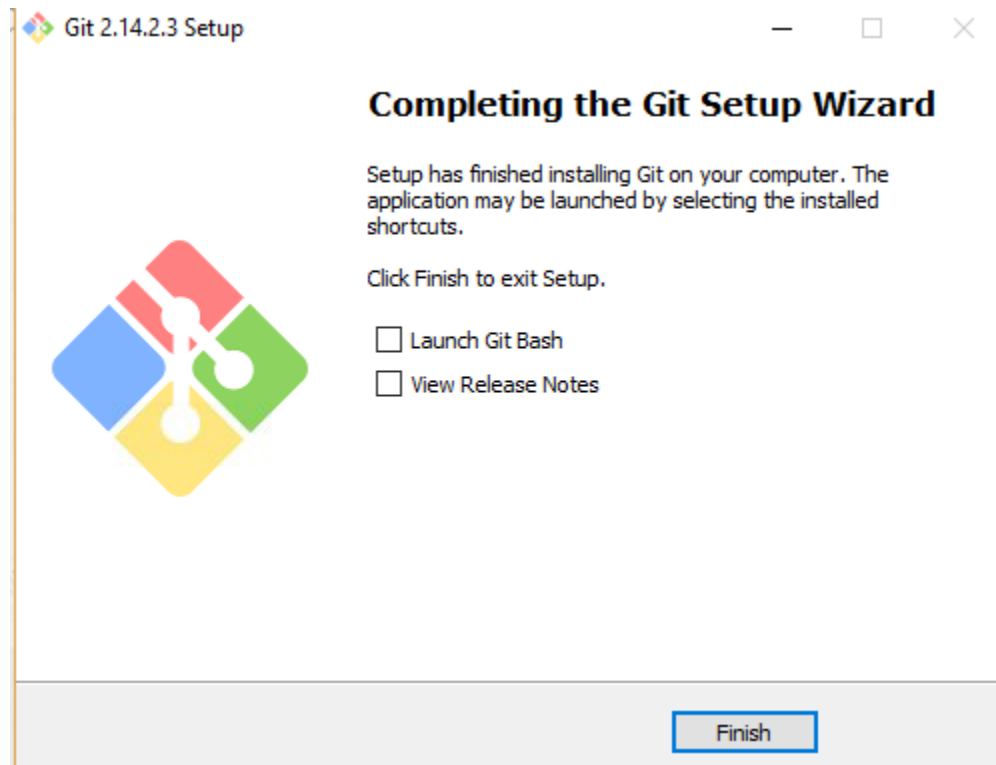
Download Git

- Your installation will begin...



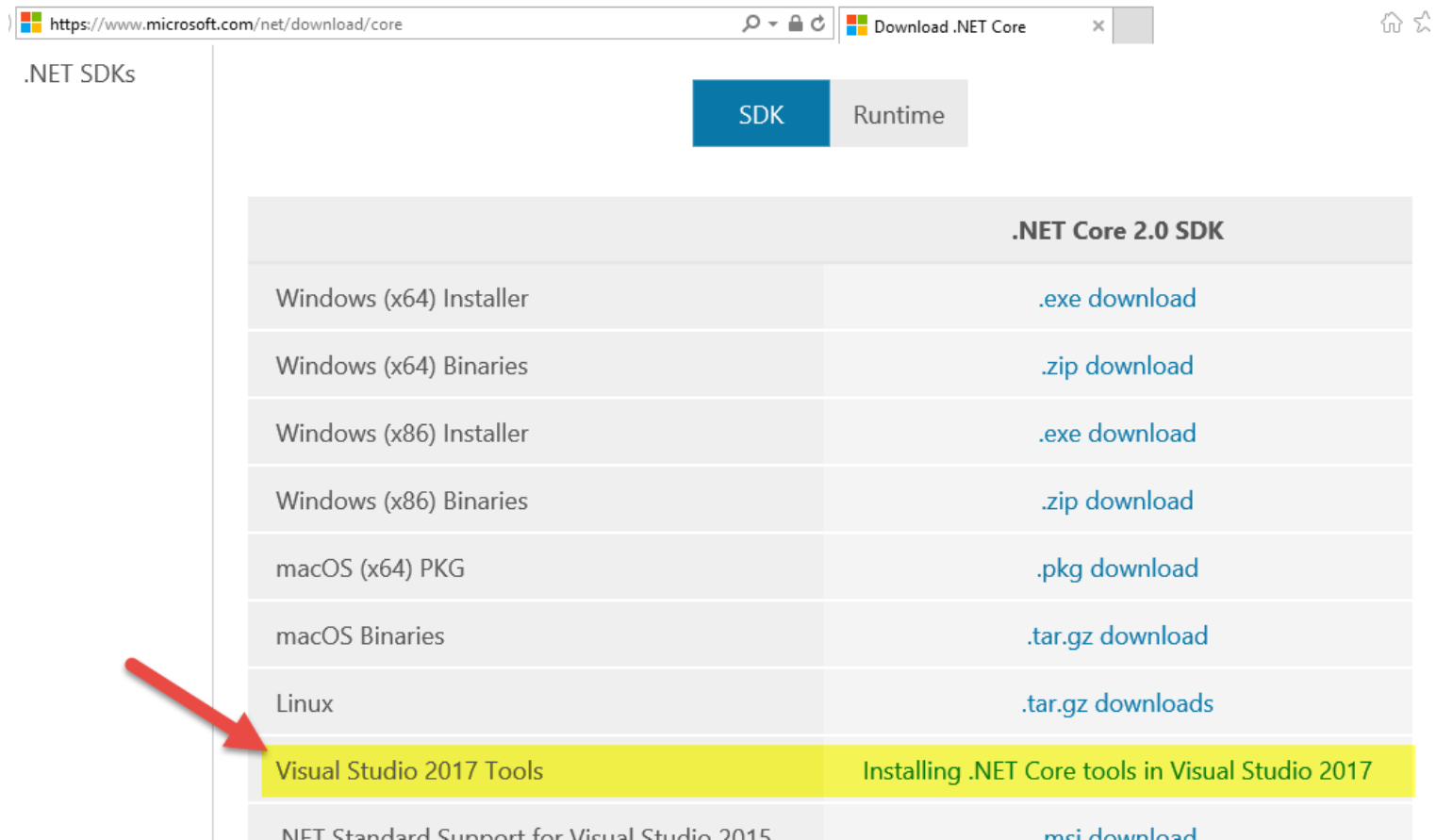
Download Git

- Once complete select 'Finish'



Download .NET Core SDK

- <https://www.microsoft.com/net/download/core>



https://www.microsoft.com/net/download/core

Download .NET Core

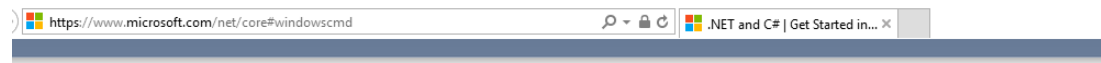
.NET SDKs

SDK Runtime

.NET Core 2.0 SDK	
Windows (x64) Installer	.exe download
Windows (x64) Binaries	.zip download
Windows (x86) Installer	.exe download
Windows (x86) Binaries	.zip download
macOS (x64) PKG	.pkg download
macOS Binaries	.tar.gz download
Linux	.tar.gz downloads
Visual Studio 2017 Tools	Installing .NET Core tools in Visual Studio 2017
.NET Standard Support for Visual Studio 2015	msi download

Download .NET Core SDK

- Select 'Download .NET Core SDK' and then select 'Run'



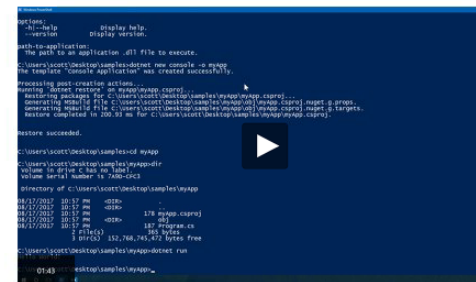
Install .NET and build your first app on Windows

1 Install .NET Core SDK

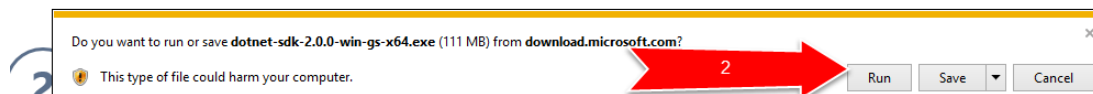
To start creating .NET apps you just need to download the .NET Core SDK for Windows.

Download .NET Core SDK

This installer will install the latest stable version of the tools and put them on your PATH so you can run dotnet from the Console.

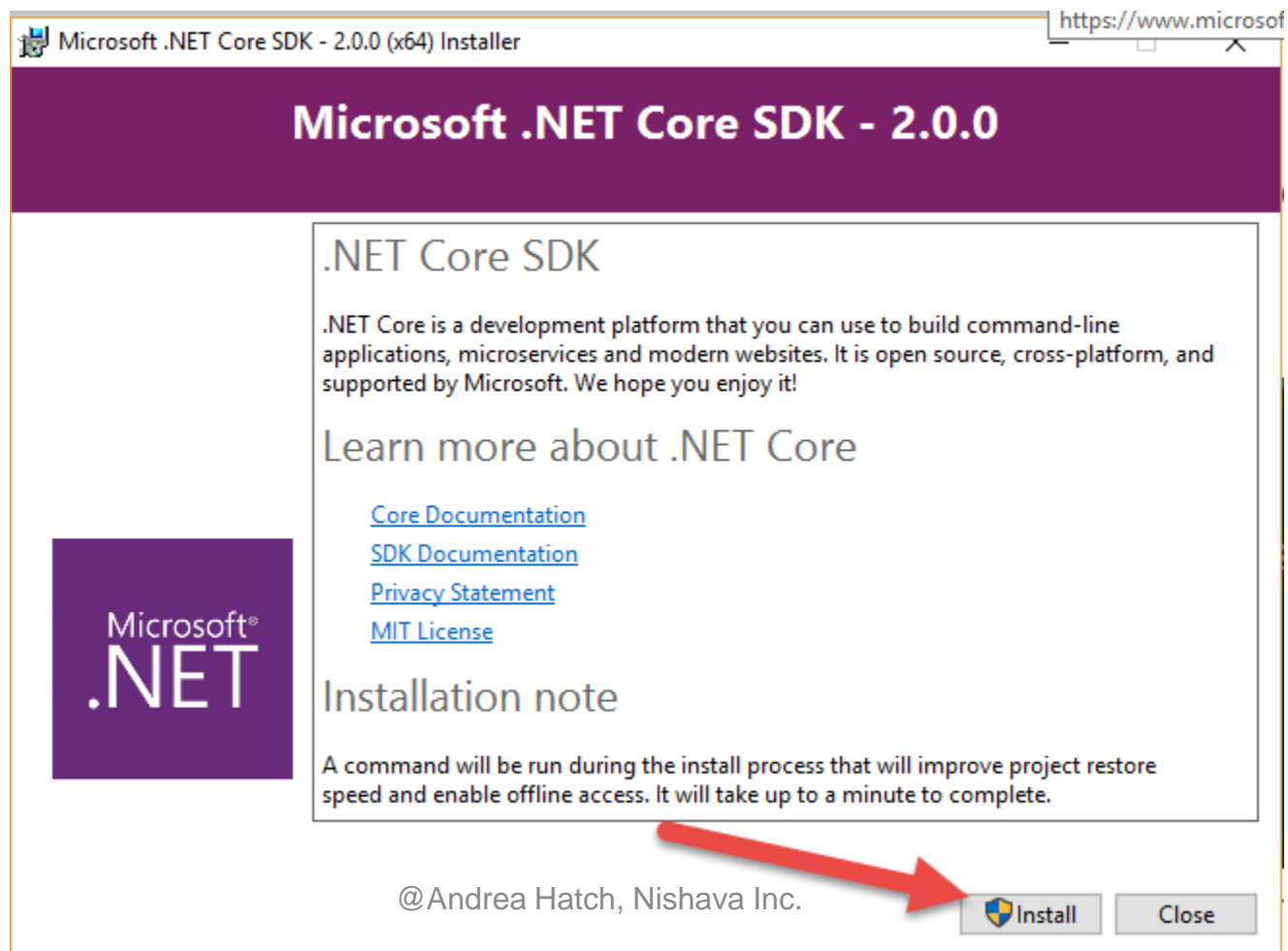


Video: Creating a Simple .NET Application on Windows



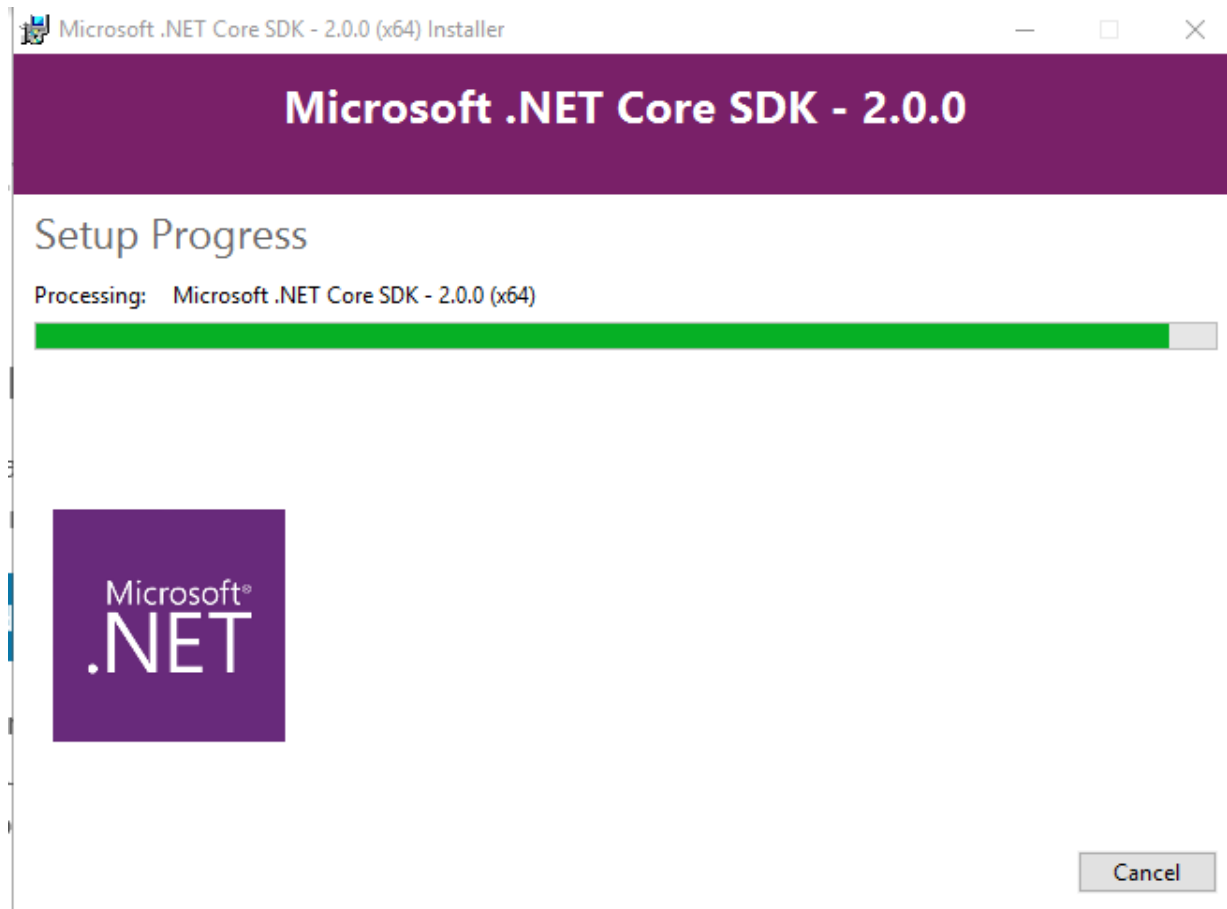
Download .NET Core SDK

- Select 'Install'



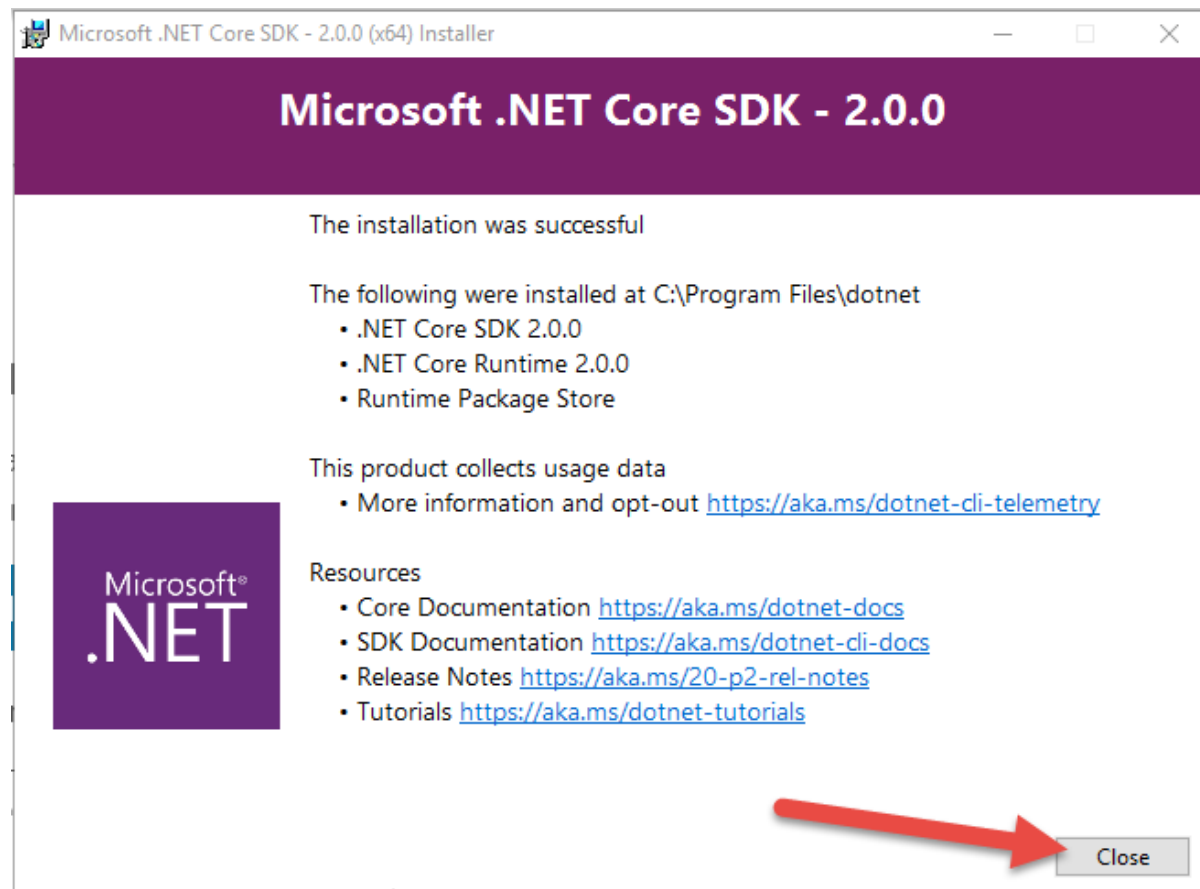
Download .NET Core SDK

- .NET Core should install



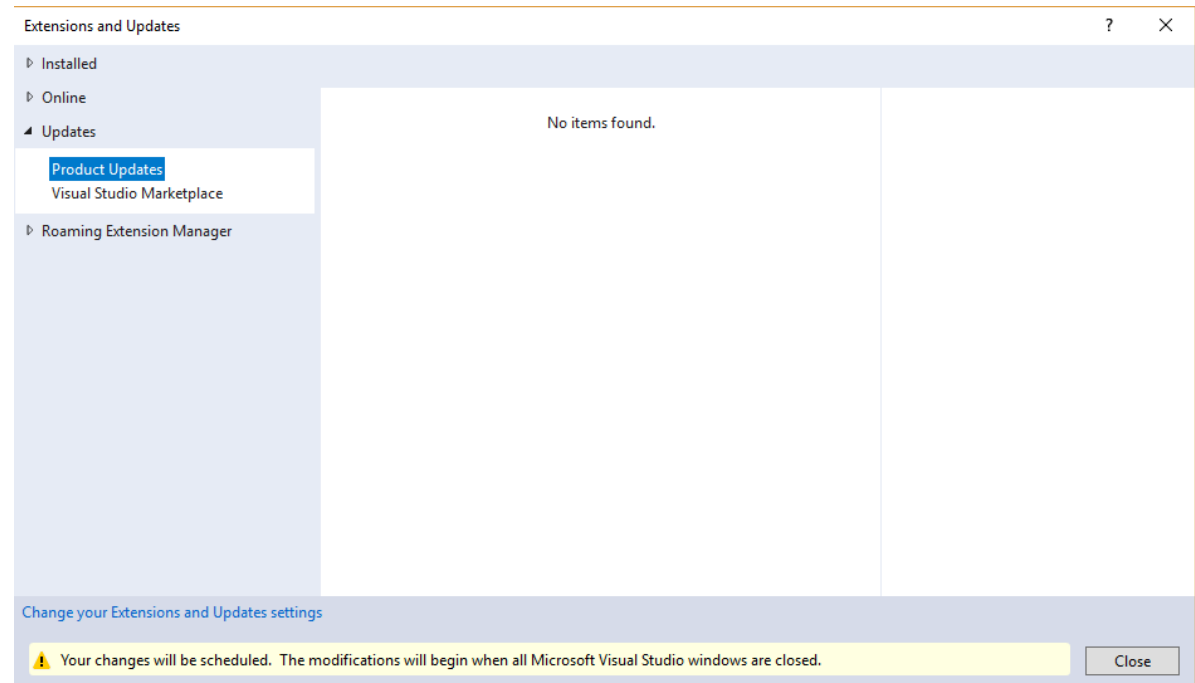
Download .NET Core SDK

- Once it is installed select 'Close'



Check Updates

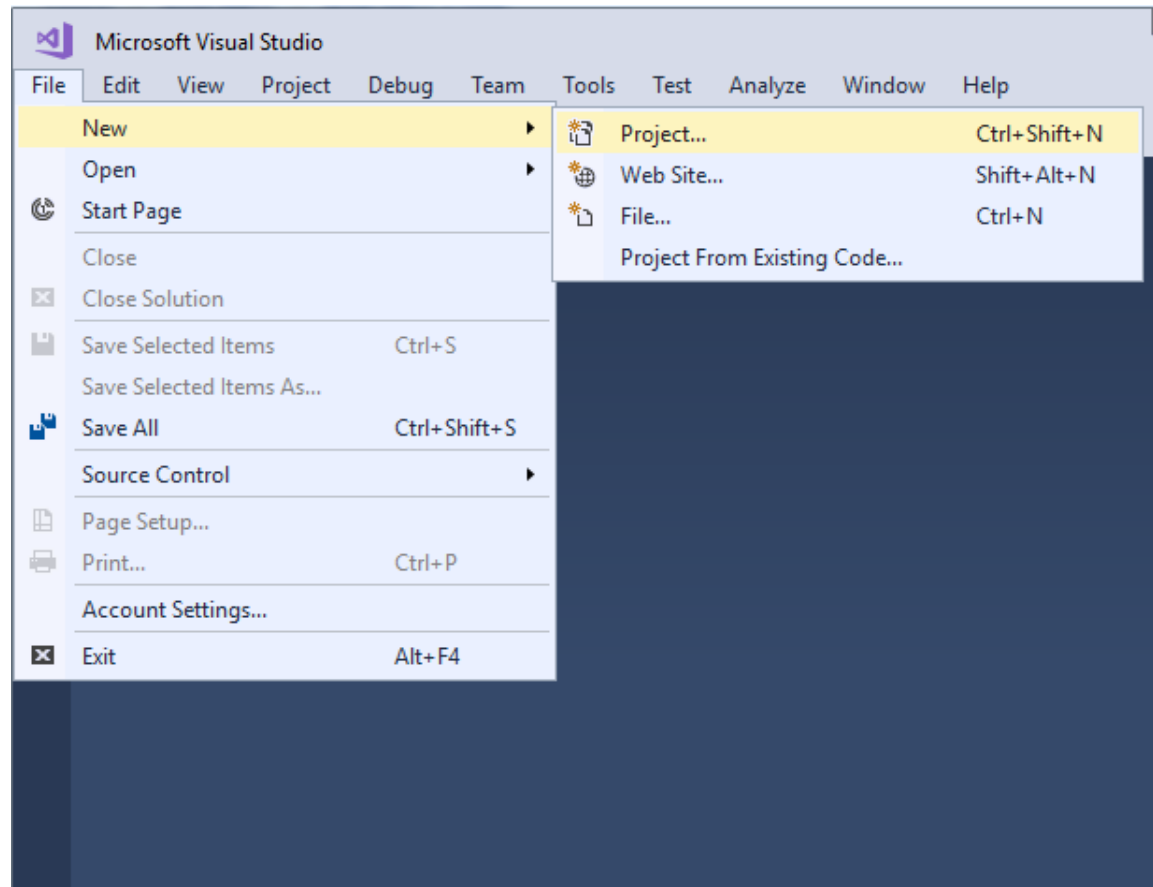
- In VS go to Tools -> Extensions and Updates
- Under Updates make sure that there are no product updates or VS Marketplace updates
- If there are, you must update these



3. Create a web app in VS to deploy to Azure

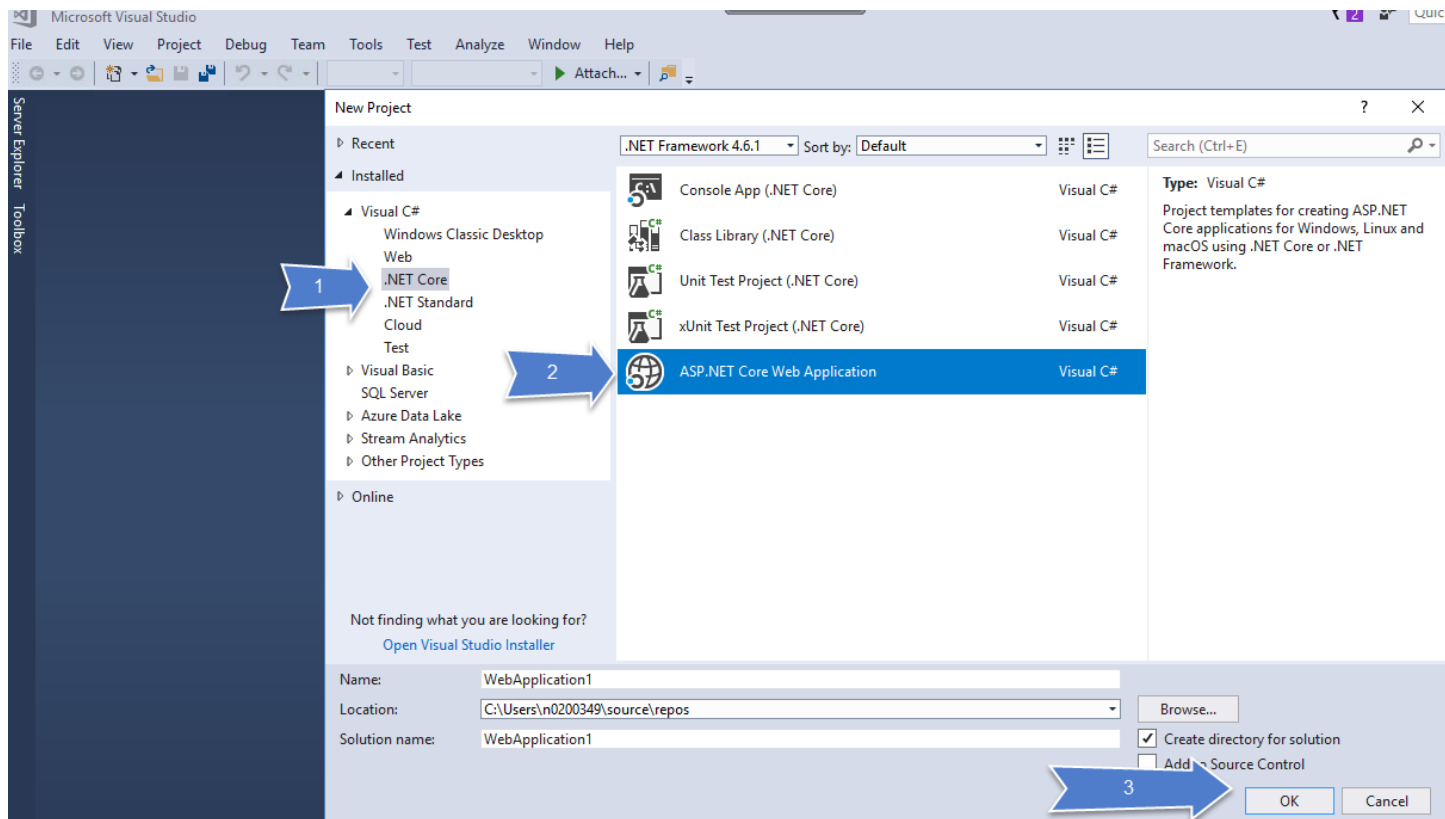
Creating a web app to deploy to Azure

- In VS, select File -> New -> Project



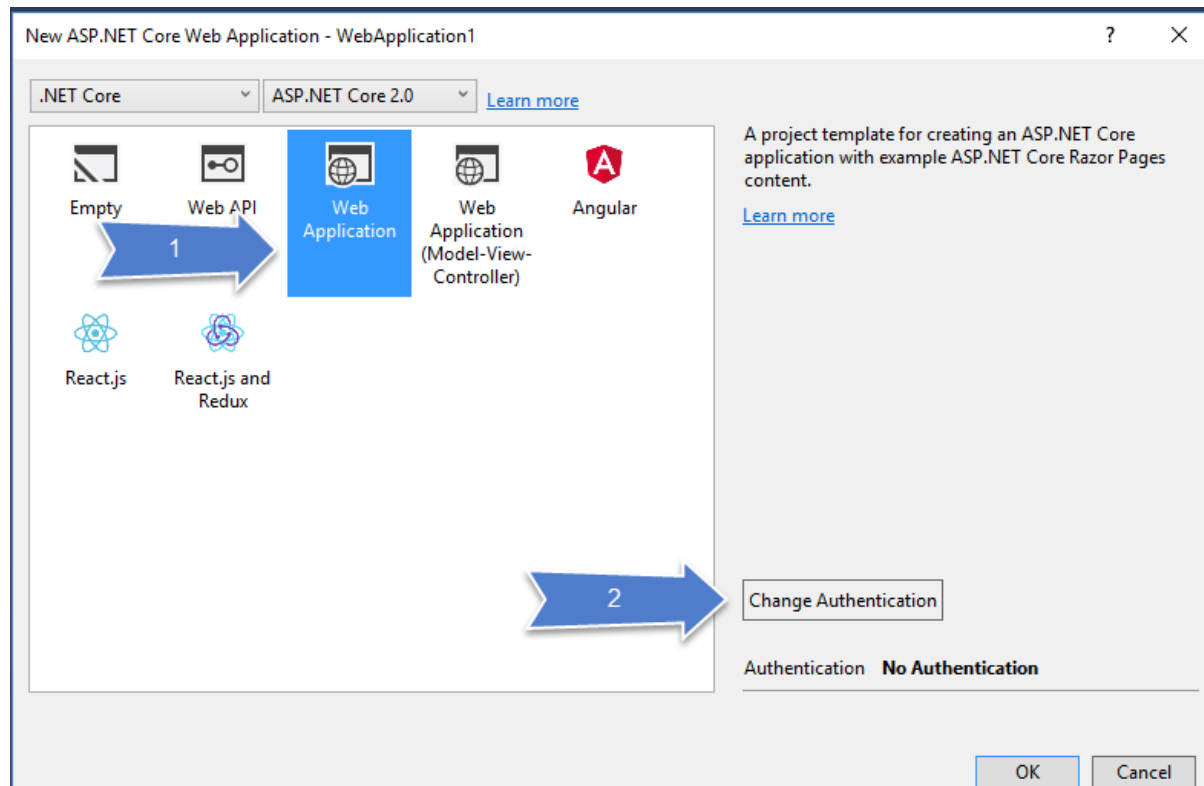
Creating a web app to deploy to Azure

- Select '.NET Core' under Visual C# and then 'ASP.NET Core Web Application' and then select 'OK'



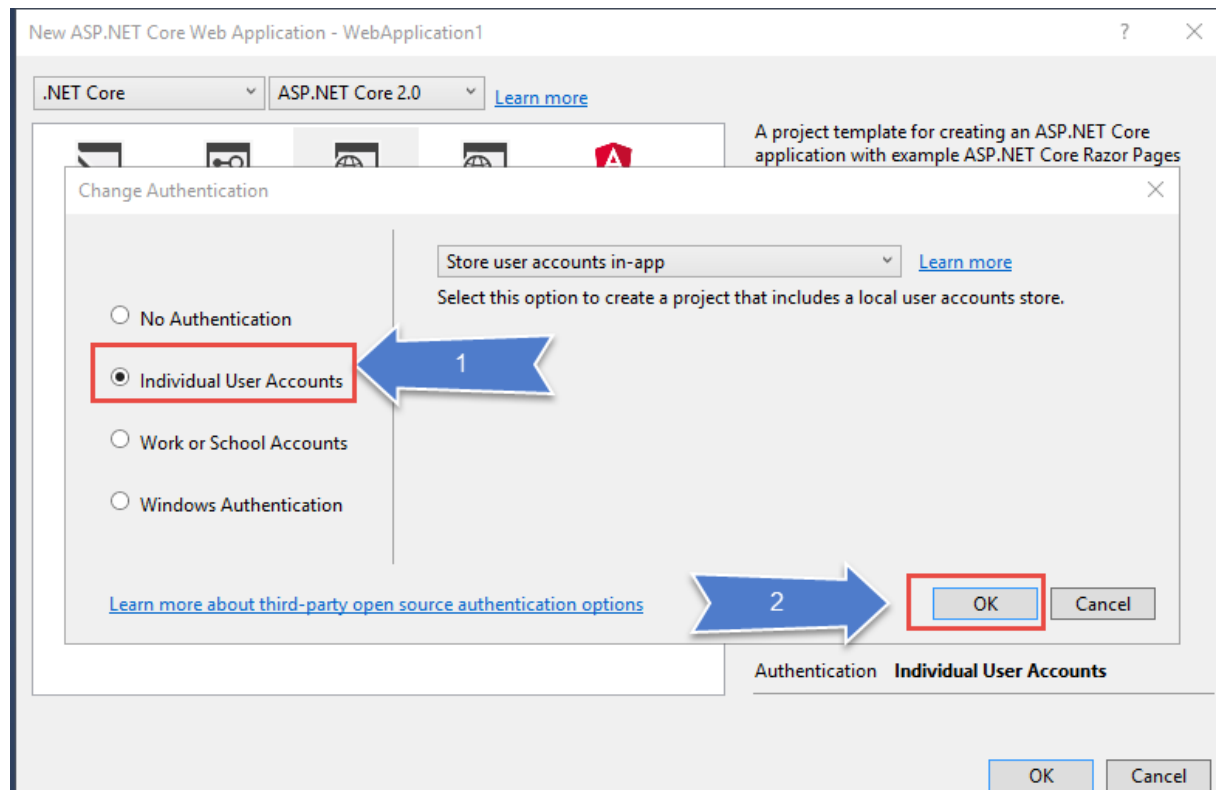
Creating a web app to deploy to Azure

- Select 'Web Application' and then 'Change authentication' on the right



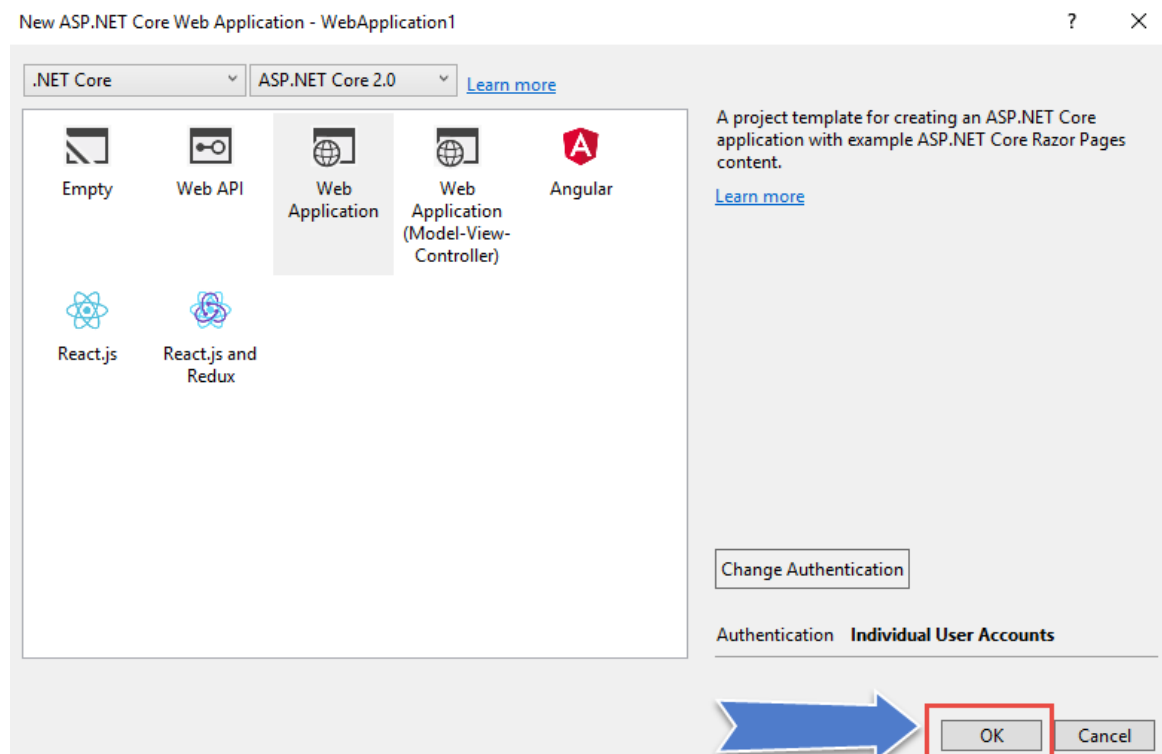
Creating a web app to deploy to Azure

- Change the authentication to Individual User Accounts and then select 'OK'



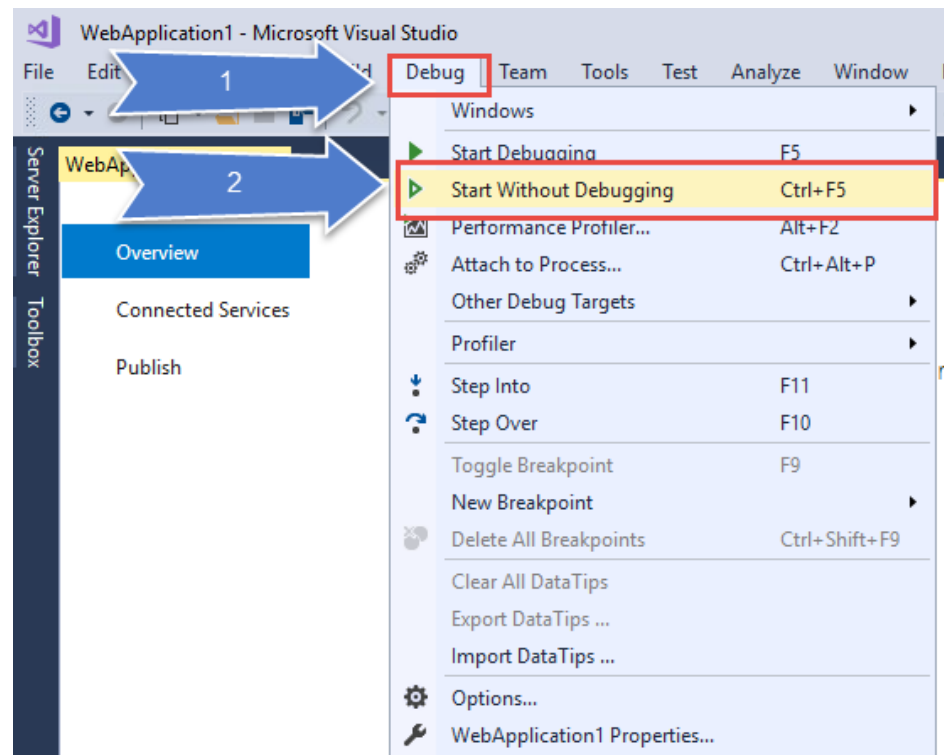
Creating a web app to deploy to Azure

- Select 'OK' on the Web Application screen and then your project will be created



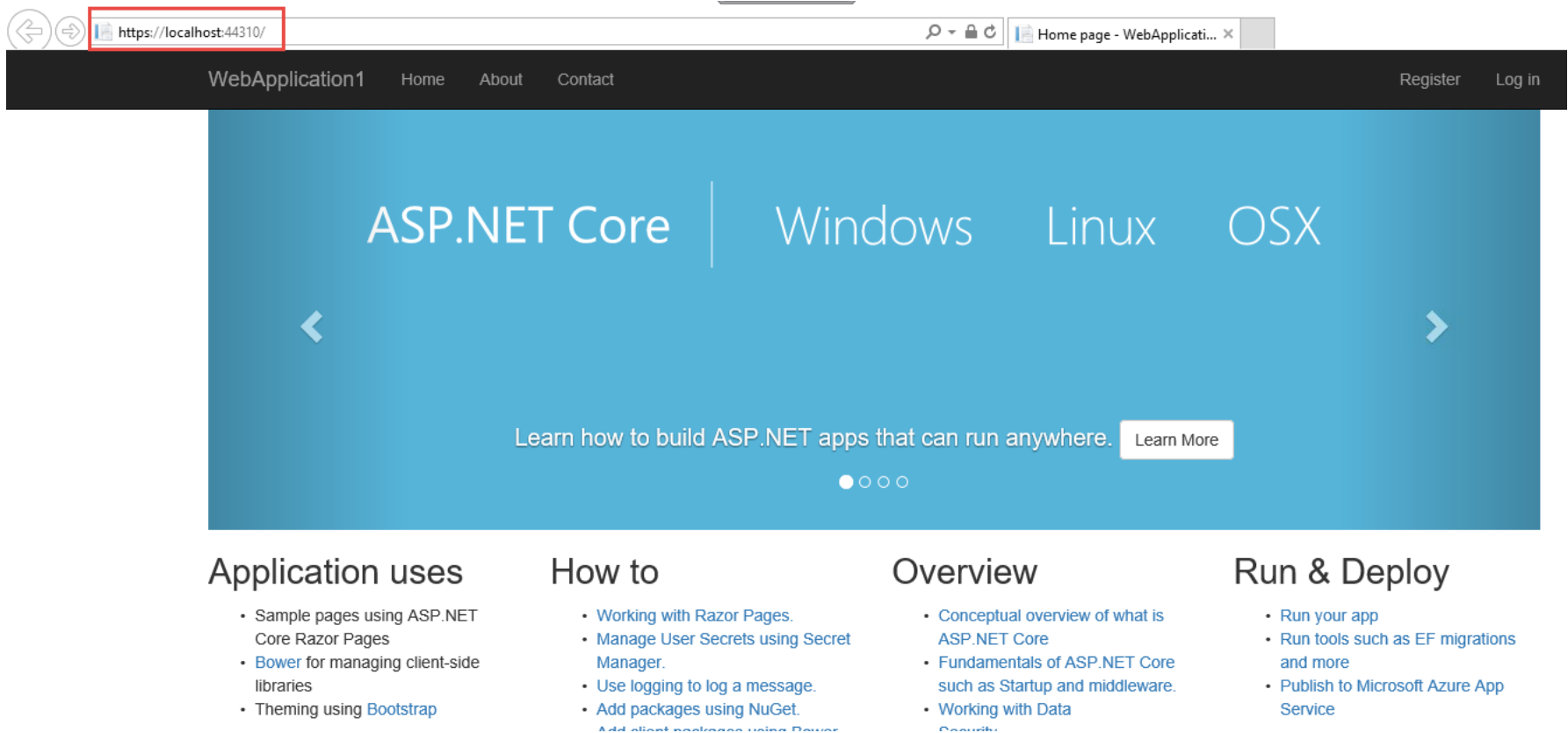
Running the app locally

- Select Debug -> Start without Debugging



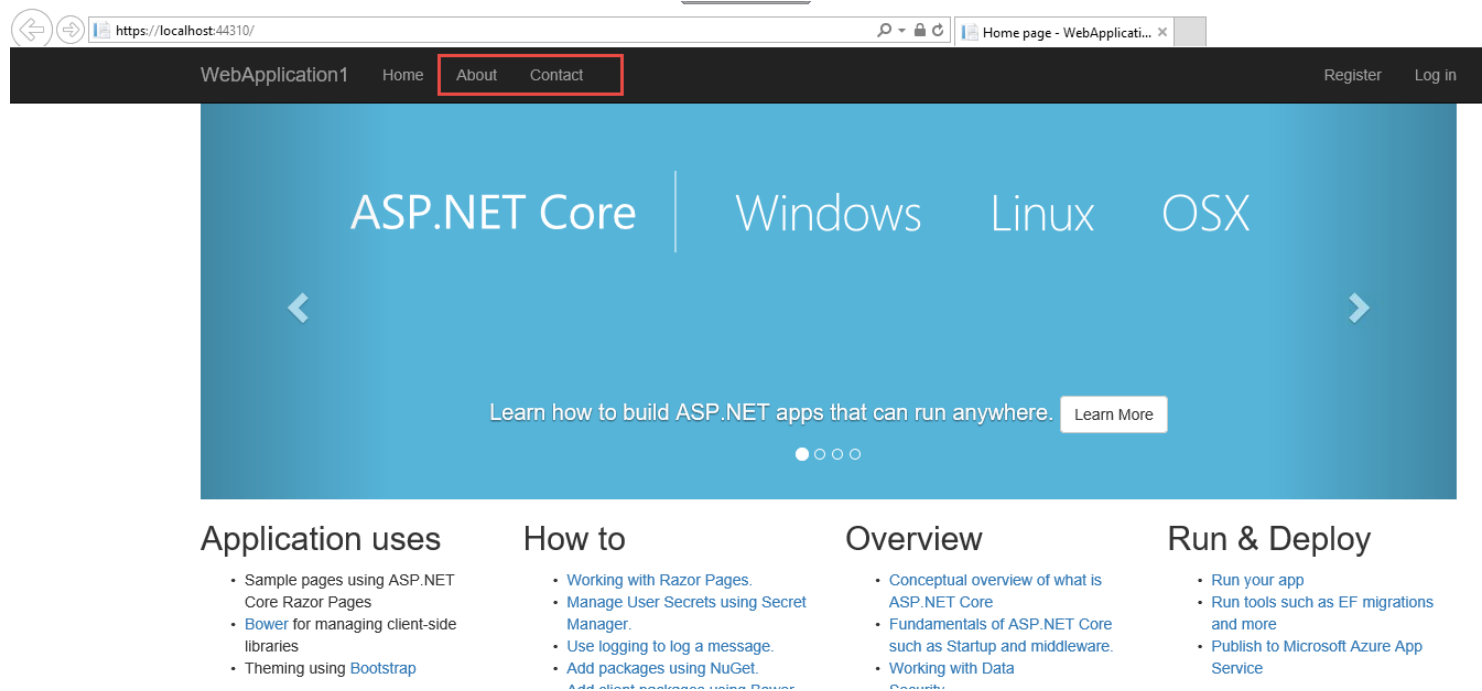
Running the app locally

- You should then see your app running on your local host



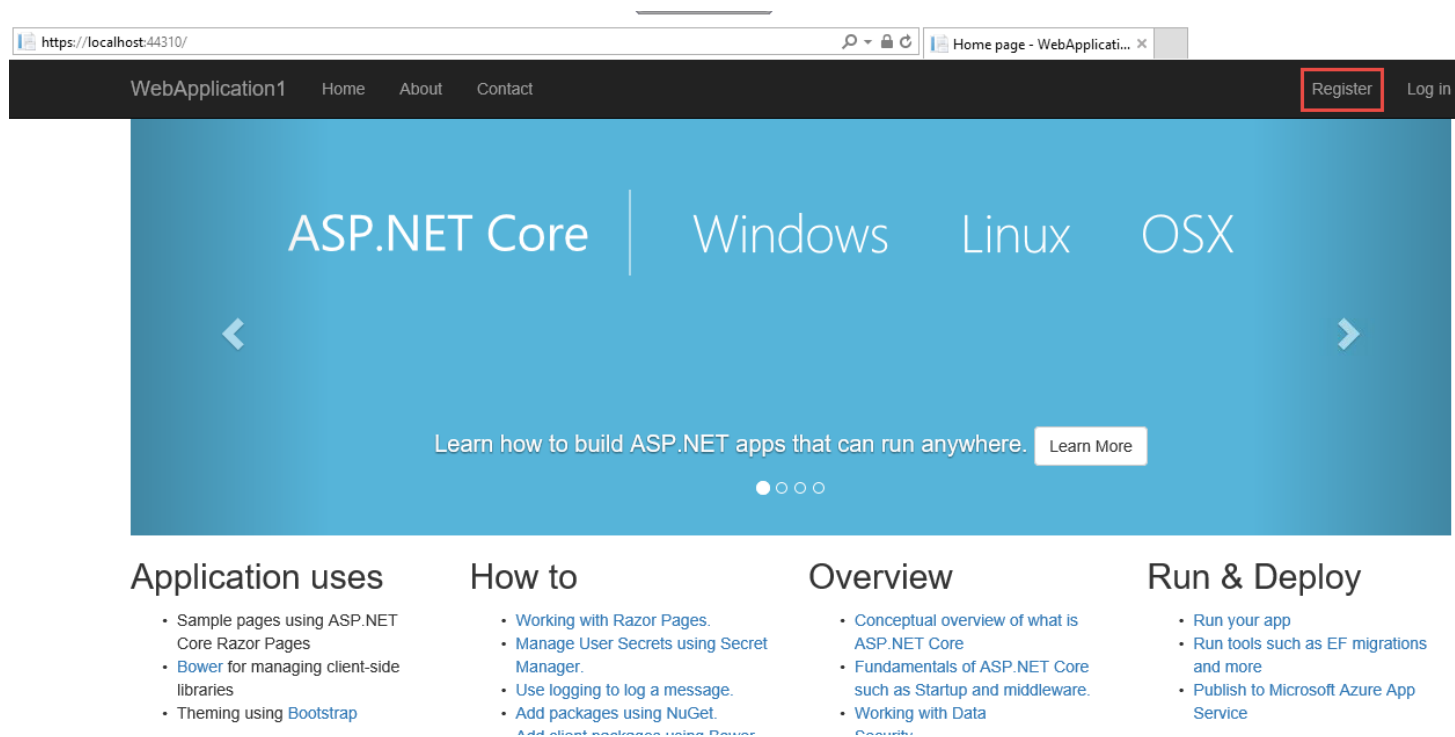
Running the app locally

- To test that the app is working select 'About' or 'Contact' on the top of your app screen



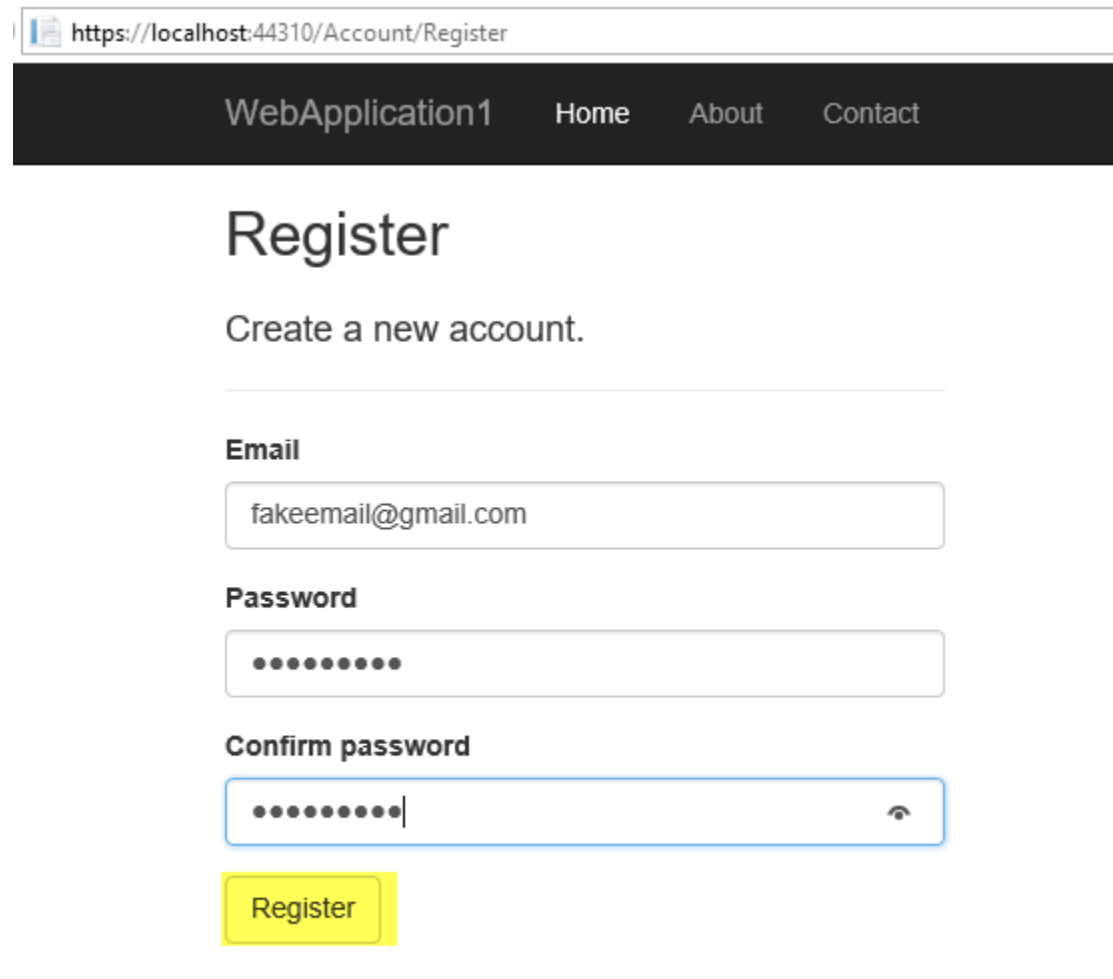
Running the app locally

- You can register your app by selecting Register on the top of the application



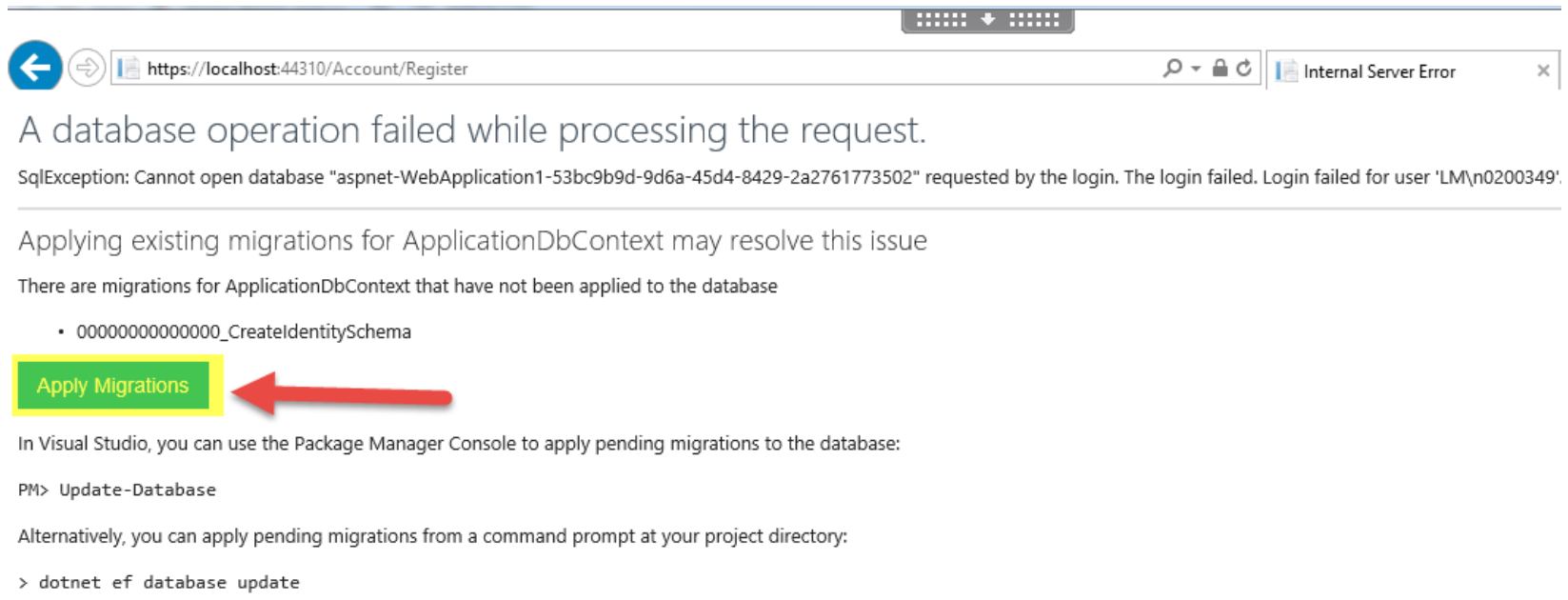
Running the app locally

- Add any fake email and password then select Register



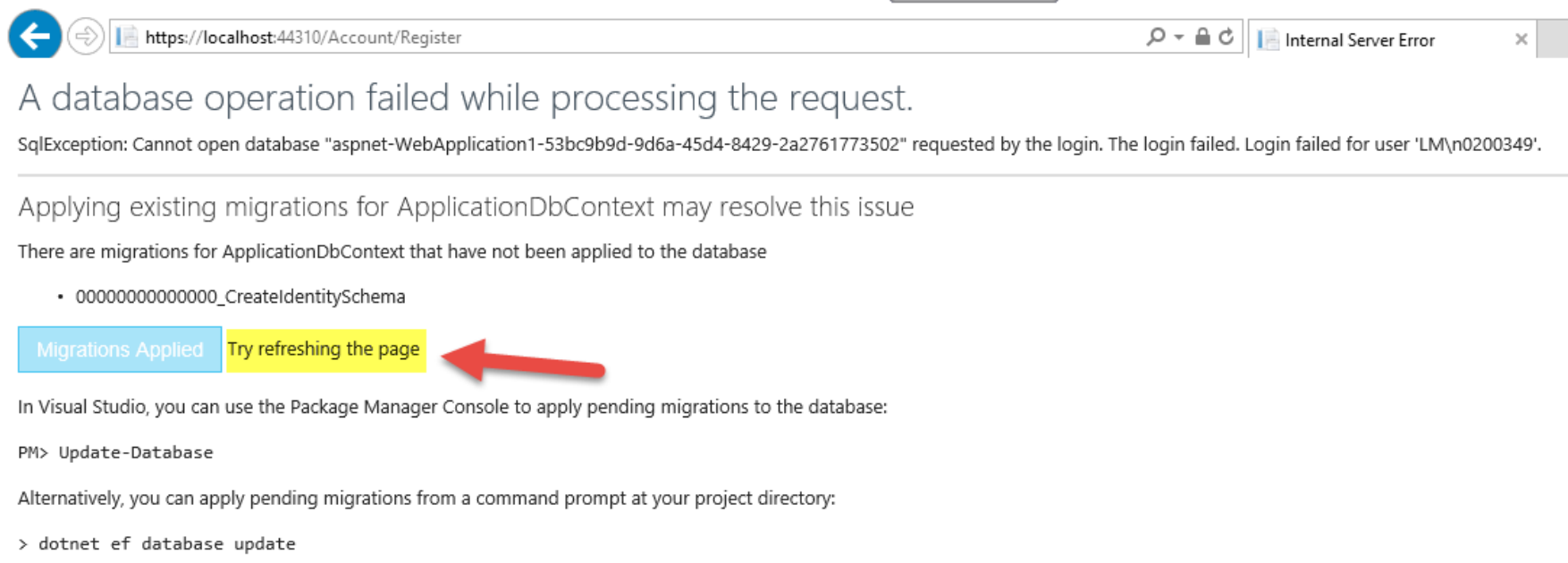
The screenshot shows a web browser window with the address bar displaying `https://localhost:44310/Account/Register`. The page has a dark navigation bar with the text "WebApplication1" and links for "Home", "About", and "Contact". The main content area is titled "Register" and includes the instruction "Create a new account." Below this are three input fields: "Email" containing "fakeemail@gmail.com", "Password" filled with dots, and "Confirm password" also filled with dots and featuring a toggle icon on the right. A yellow "Register" button is positioned at the bottom of the form.

- Your page will display an error until you select Apply Migrations



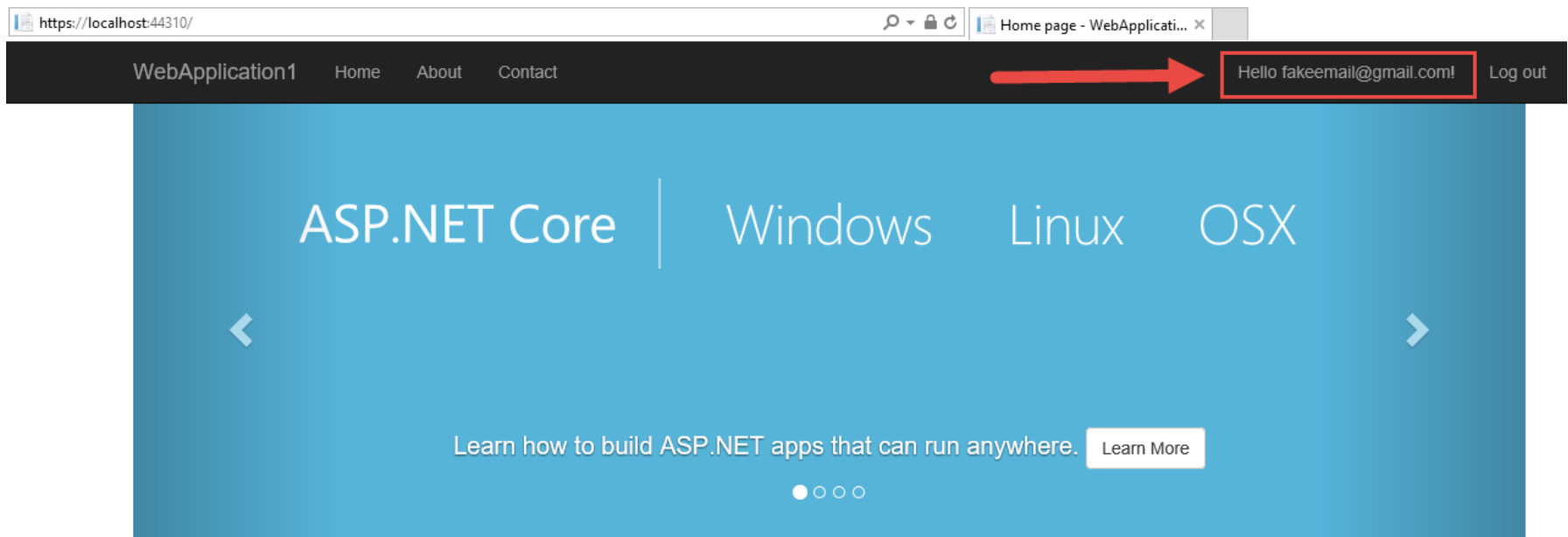
Running the app locally

- Once the migrations are applied you will be asked to refresh your screen



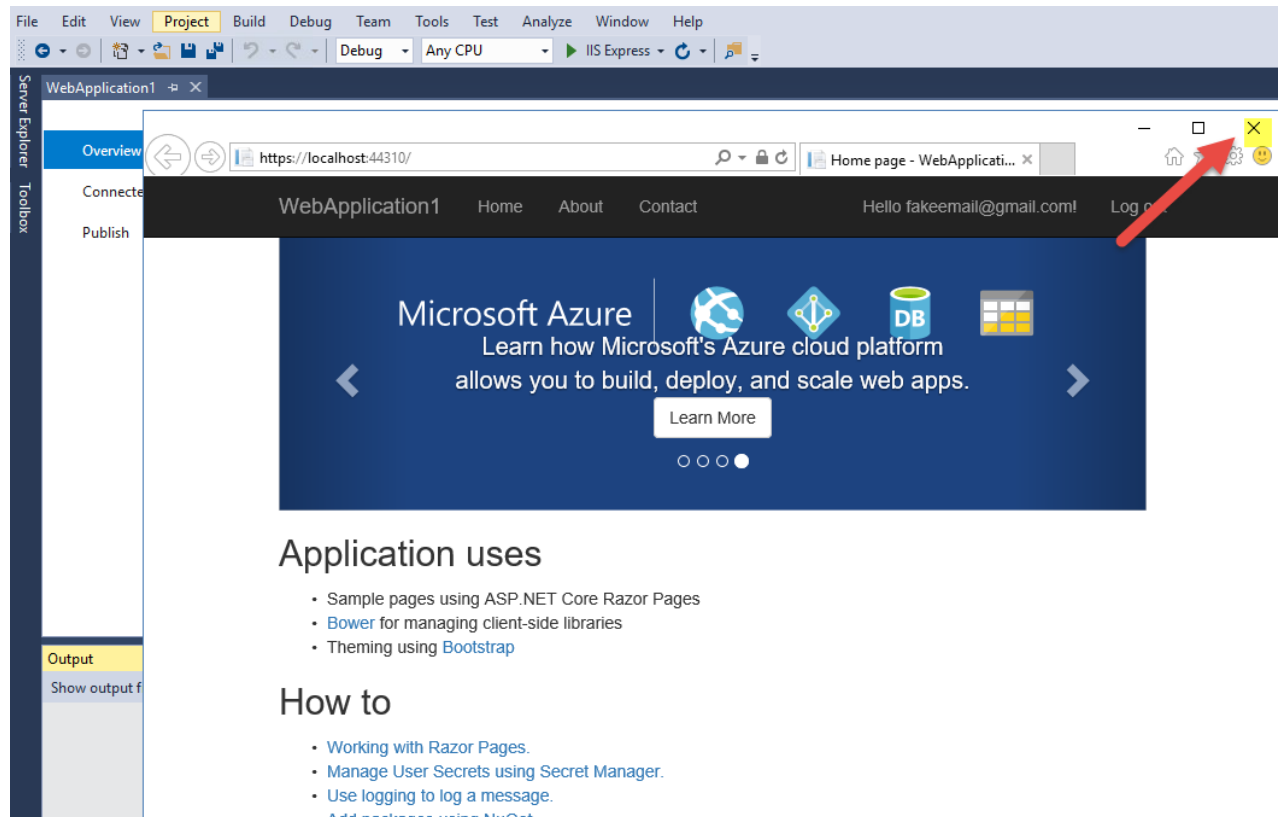
Running the app locally

- Your screen will now display the homepage again but this time with your email account



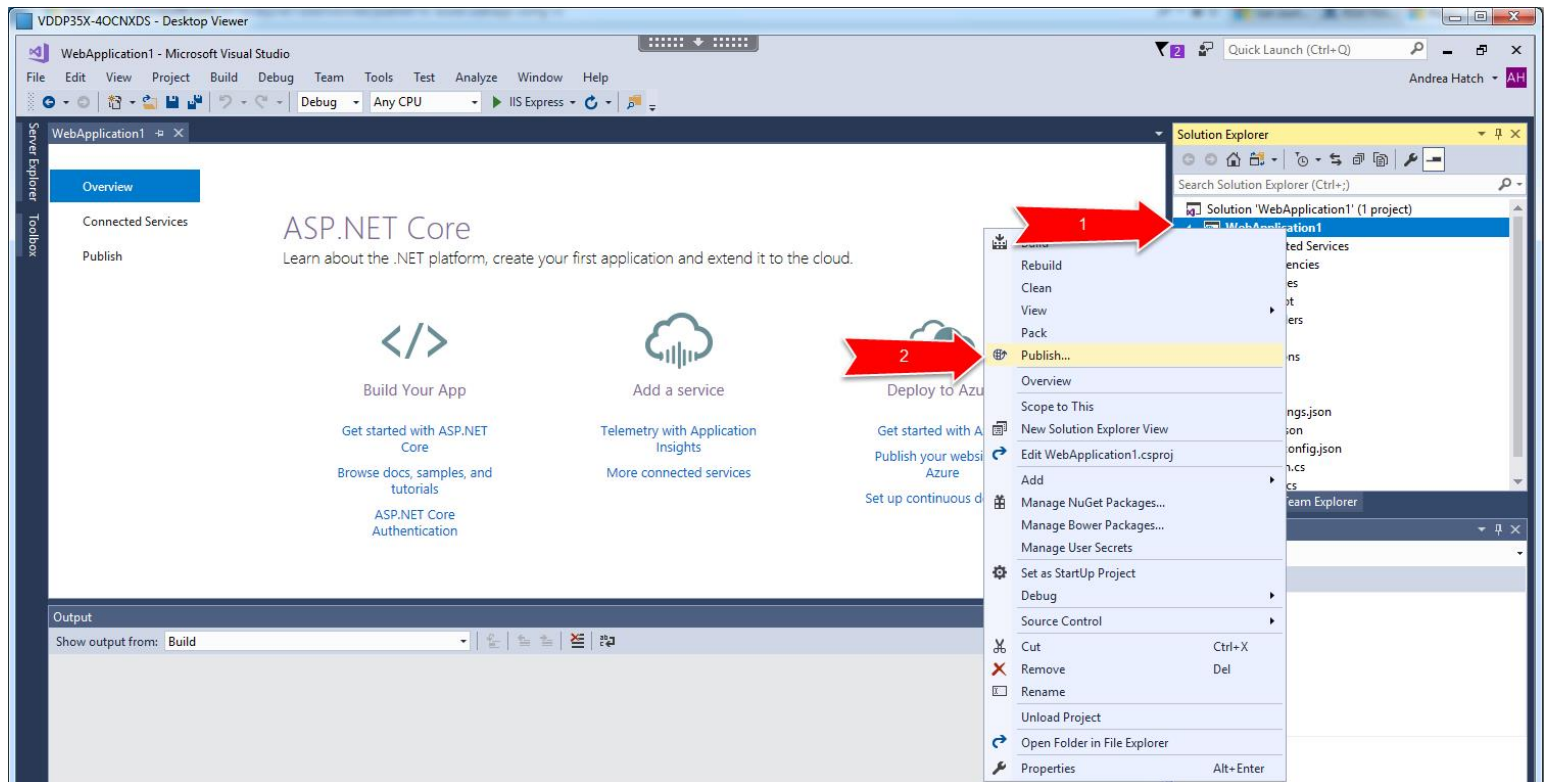
Running the app locally

- Click out of viewing your application by selecting the x button on the browser



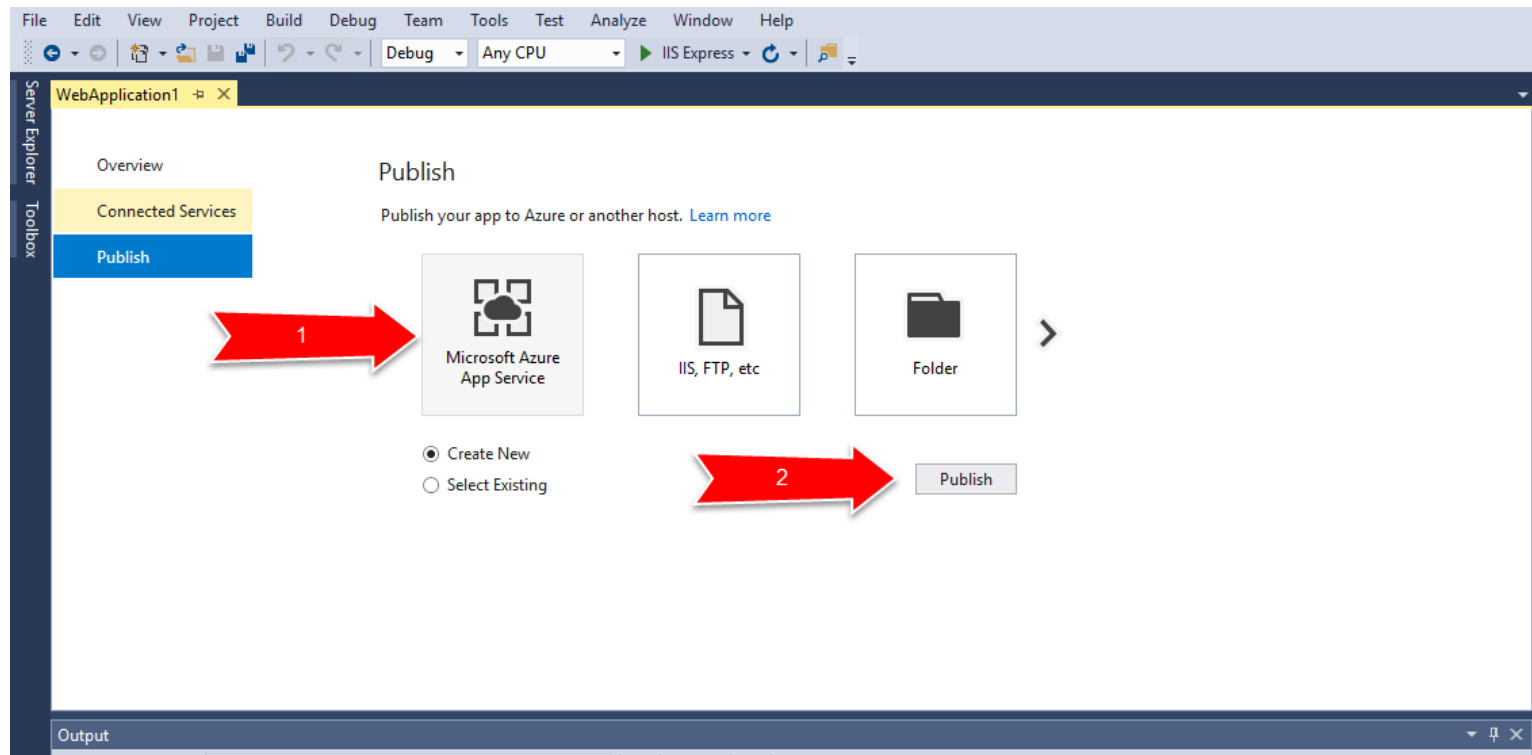
Deploy to Azure

- Next lets deploy your app to Azure by right clicking on your application name under the Solution Explorer and then selecting 'Publish'



Deploy to Azure

- Select Microsoft Azure on the Publish screen and then select 'Publish'



Deploy to Azure

- Add a unique name for your App, select the free trial subscription, create a new resource group and DO NOT select 'Create' just yet

Create App Service
Host your web and mobile applications, REST APIs, and more in Azure

Microsoft account
andreaahoward90@gmail.com

Hosting
Services

App Name: AndreasWebApp [Change Type](#)

Subscription: Free Trial

Resource Group: AHResourceGroup (eastus) [New...](#)

App Service Plan: WebApplication120171023111933Plan* [New...](#)

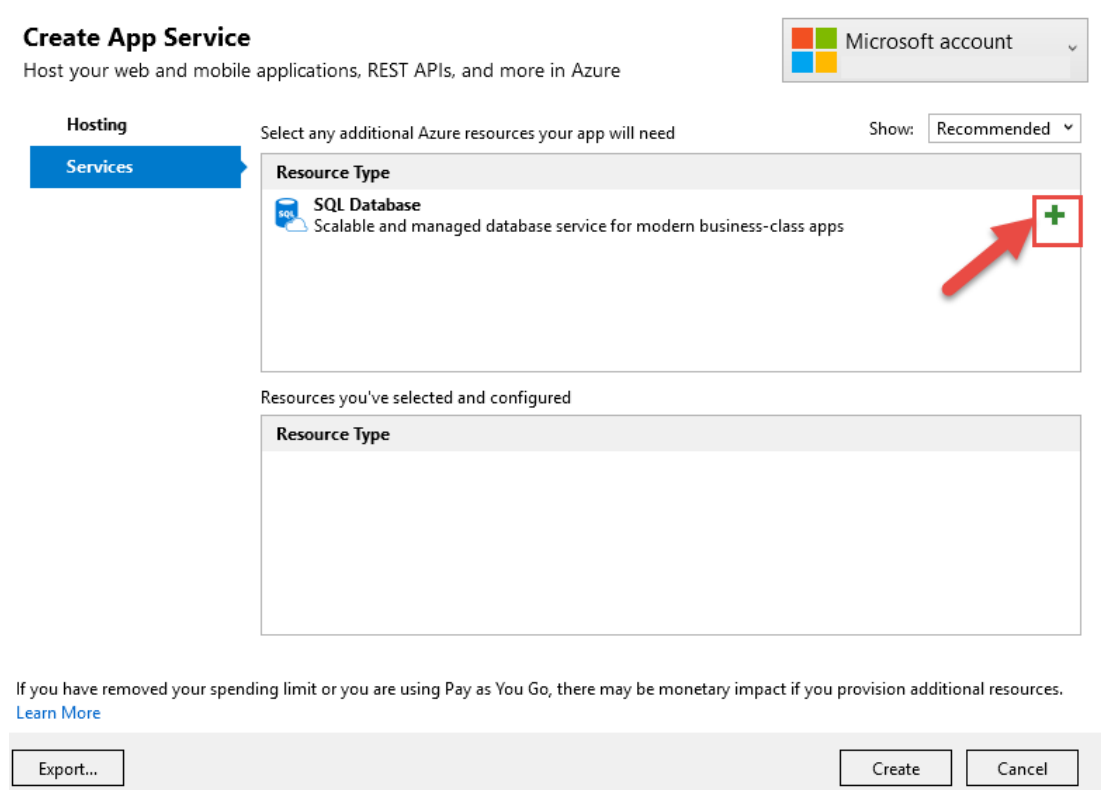
Clicking the Create button will create the following Azure resources
[Explore additional Azure services](#)
App Service - AndreasWebApp
App Service Plan - WebApplication120171023111933Plan

If you have removed your spending limit or you are using Pay as You Go, there may be monetary impact if you provision additional resources.
[Learn More](#)

[Export...](#) [Create](#) [Cancel](#)

Deploy to Azure

- Select the 'Services' tab and then select the green plus sign to add a Database




Create App Service
Host your web and mobile applications, REST APIs, and more in Azure


Microsoft account

Hosting
Services

Select any additional Azure resources your app will need Show: Recommended

Resource Type

 **SQL Database**
Scalable and managed database service for modern business-class apps



Resources you've selected and configured

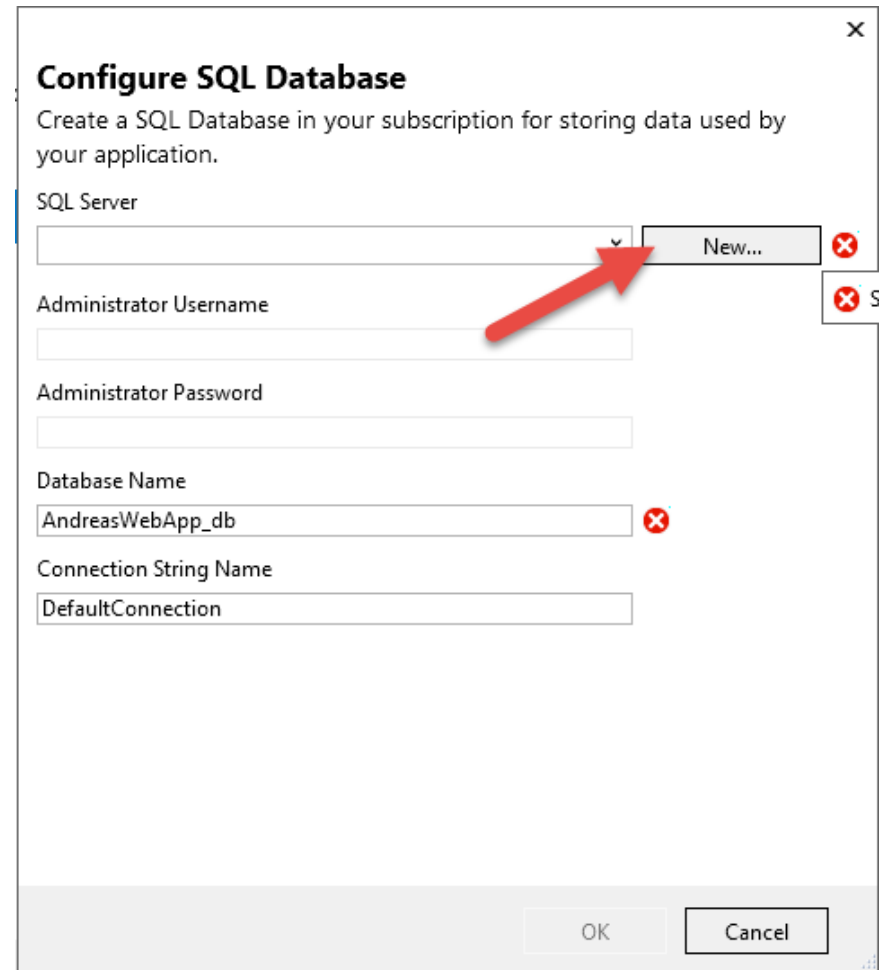
Resource Type

If you have removed your spending limit or you are using Pay as You Go, there may be monetary impact if you provision additional resources.
[Learn More](#)

Export... Create Cancel

Deploy to Azure

- Select 'New' to create a new SQL Database



Configure SQL Database
Create a SQL Database in your subscription for storing data used by your application.

SQL Server
[Dropdown Menu] **New...** [X]

Administrator Username
[Text Field]

Administrator Password
[Text Field]

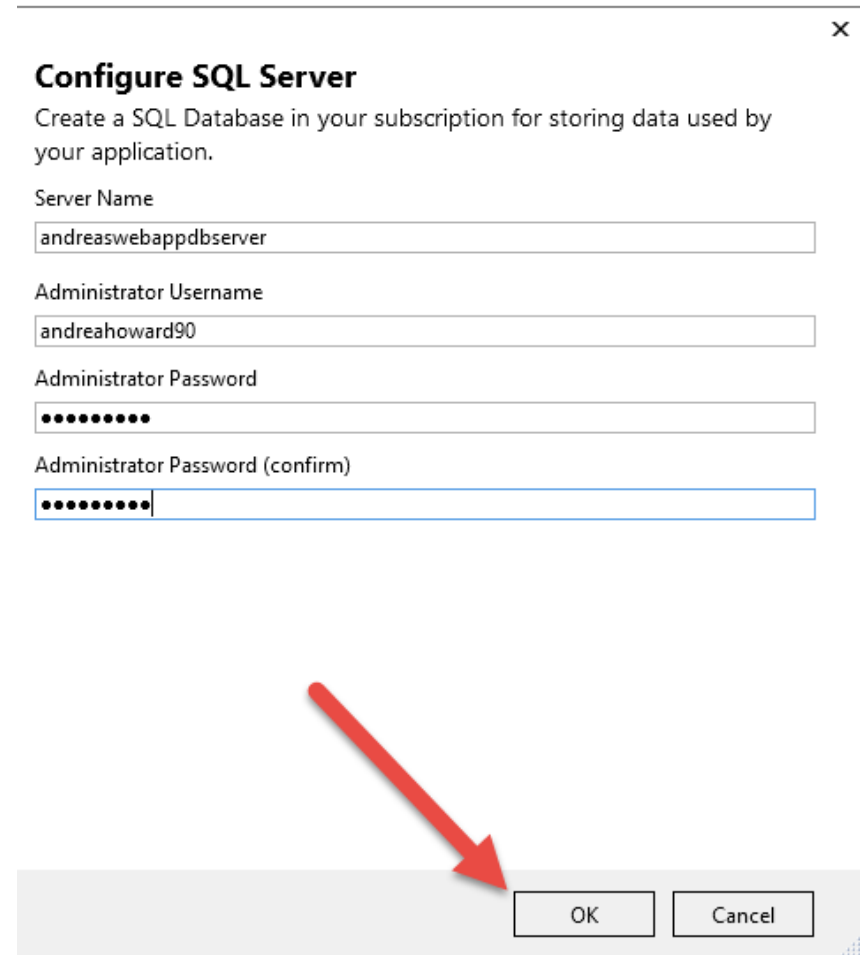
Database Name
AndreasWebApp_db [X]

Connection String Name
DefaultConnection

OK Cancel

Deploy to Azure

- Create a Username and Password for your newly created DB and then select 'OK'



Configure SQL Server X

Create a SQL Database in your subscription for storing data used by your application.

Server Name

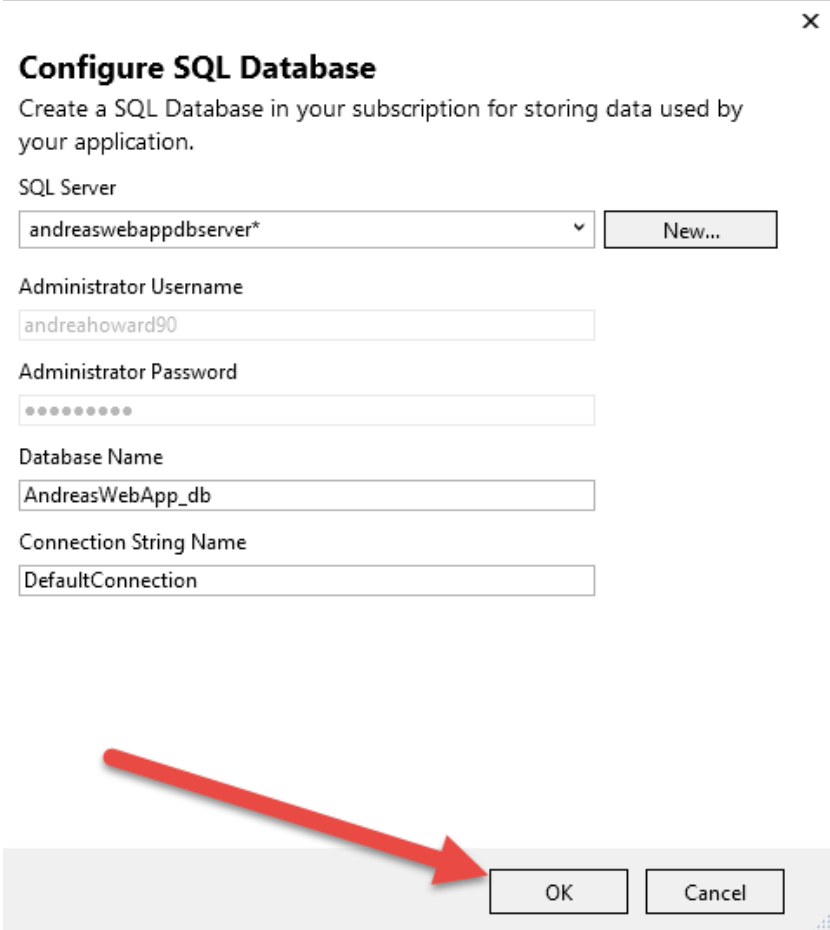
Administrator Username

Administrator Password

Administrator Password (confirm)

Deploy to Azure

- Your DB information should now appear and then select 'OK'



Configure SQL Database
Create a SQL Database in your subscription for storing data used by your application.

SQL Server
andreaswebappdbserver* New...

Administrator Username
andreahoward90

Administrator Password
••••••••

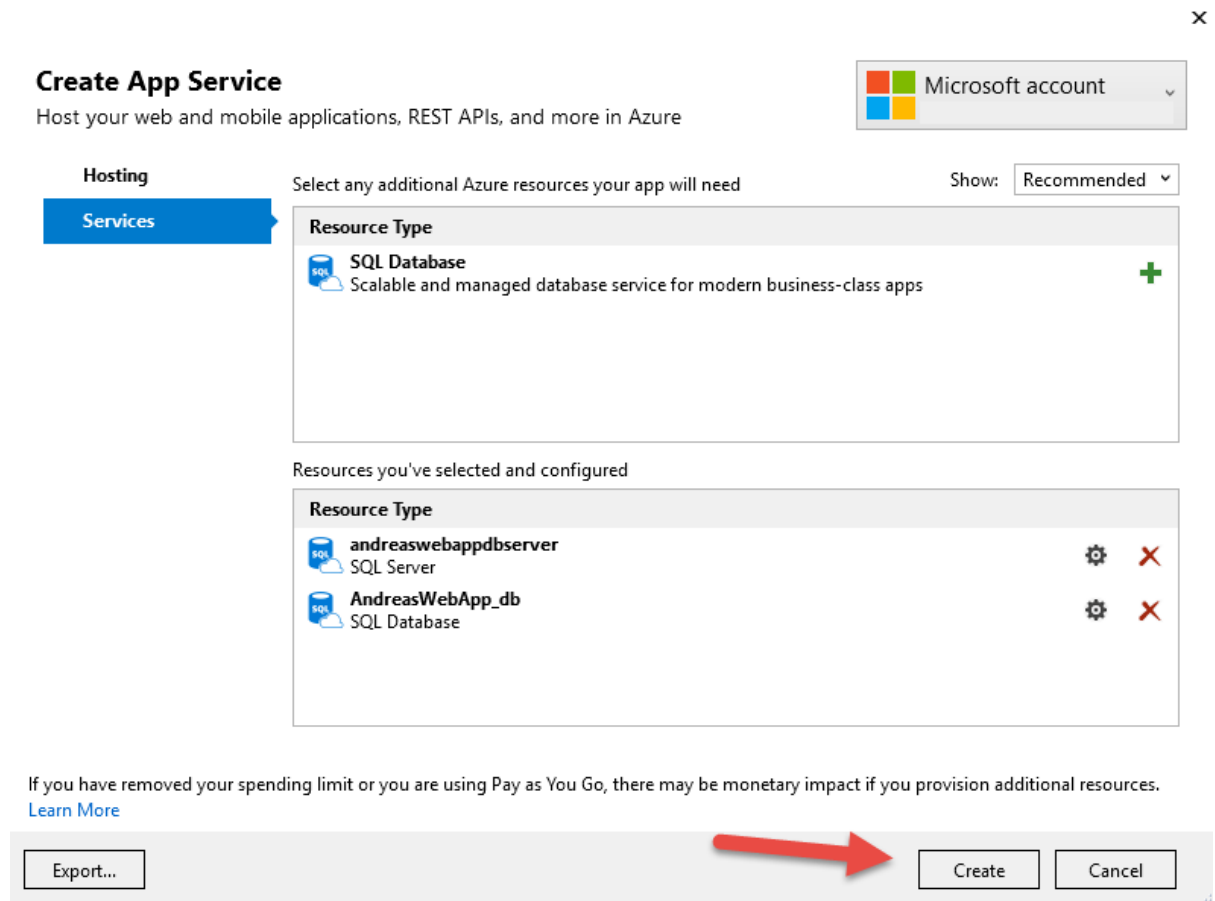
Database Name
AndreasWebApp_db

Connection String Name
DefaultConnection

OK Cancel

Deploy to Azure

- You can view your newly created Resources before selecting 'Create'




Create App Service
Host your web and mobile applications, REST APIs, and more in Azure



Microsoft account

Hosting
Services

Select any additional Azure resources your app will need Show: Recommended

Resource Type
 SQL Database Scalable and managed database service for modern business-class apps

Resources you've selected and configured

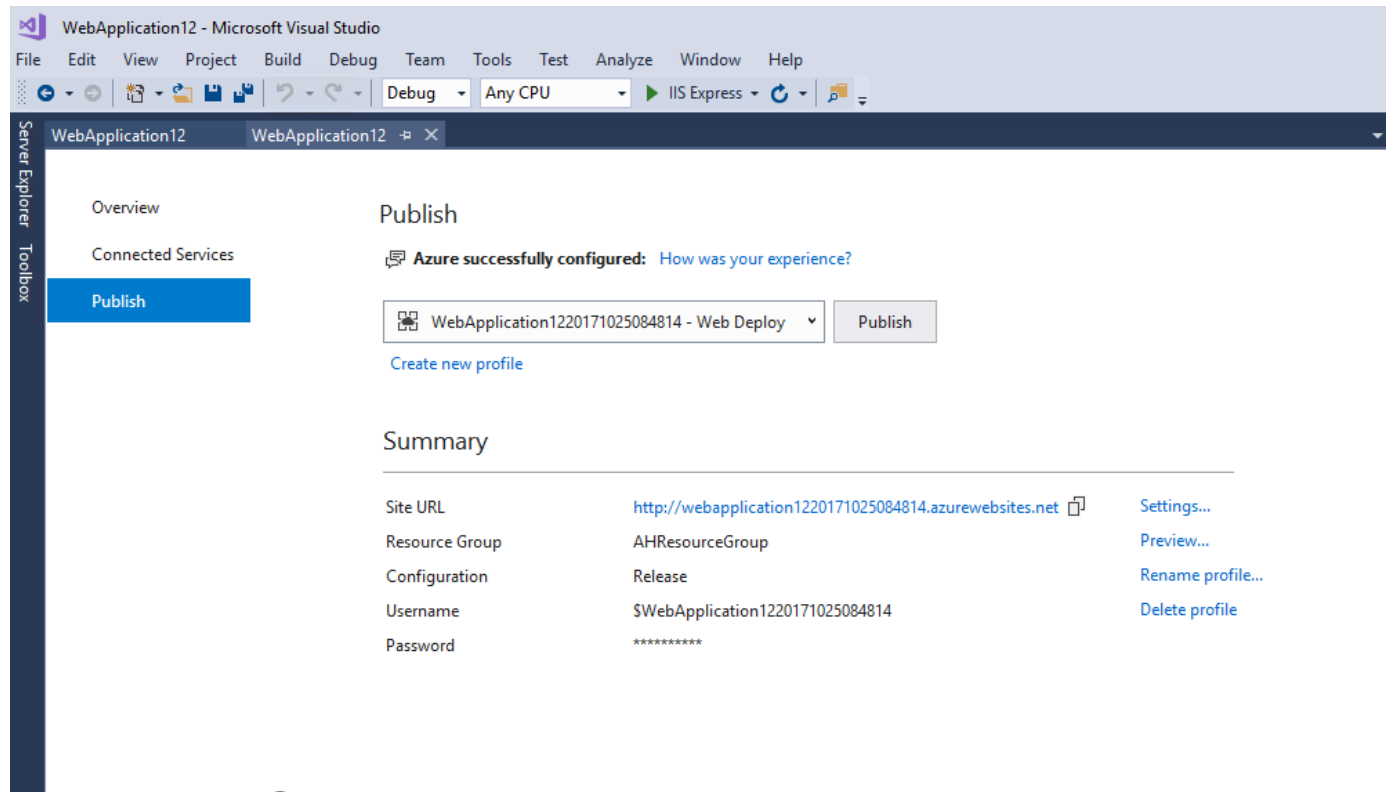
Resource Type
 andreaswebappdbserver SQL Server
 AndreasWebApp_db SQL Database

If you have removed your spending limit or you are using Pay as You Go, there may be monetary impact if you provision additional resources.
[Learn More](#)

Export... Create Cancel

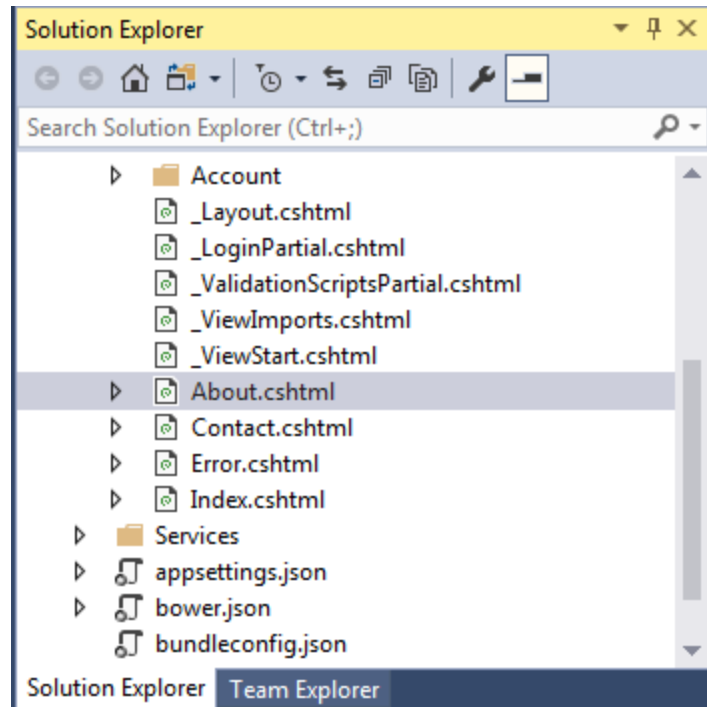
Deploy to Azure

- You will then see the Publish screen and your app should be published now. Your app will also appear in your browser.



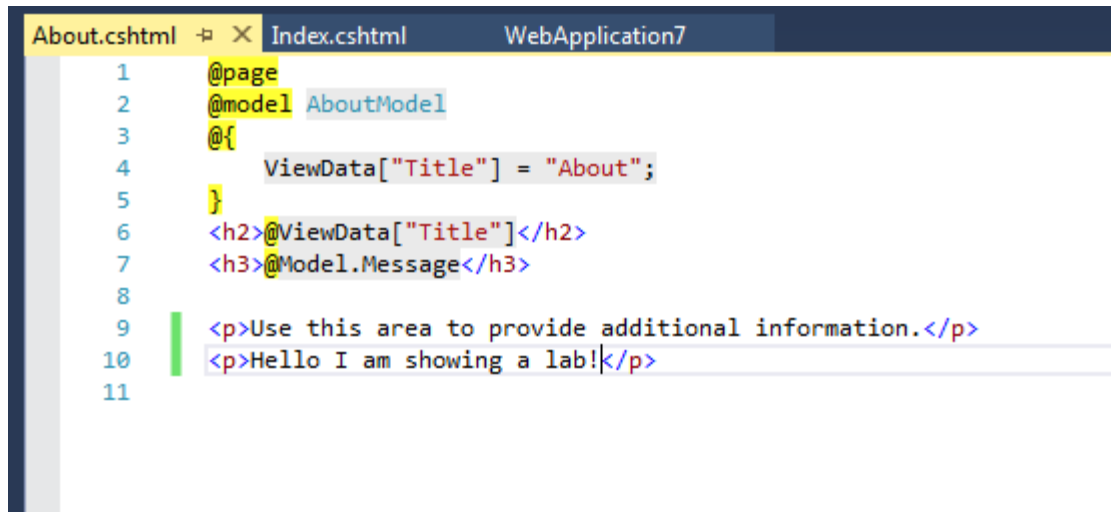
Updating the App

- In VS in the Solution Explorer open Pages-> About



Updating the App

- Update your code to add a little information about the app.
 - Ex. I wrote “Hello I am showing a lab!”

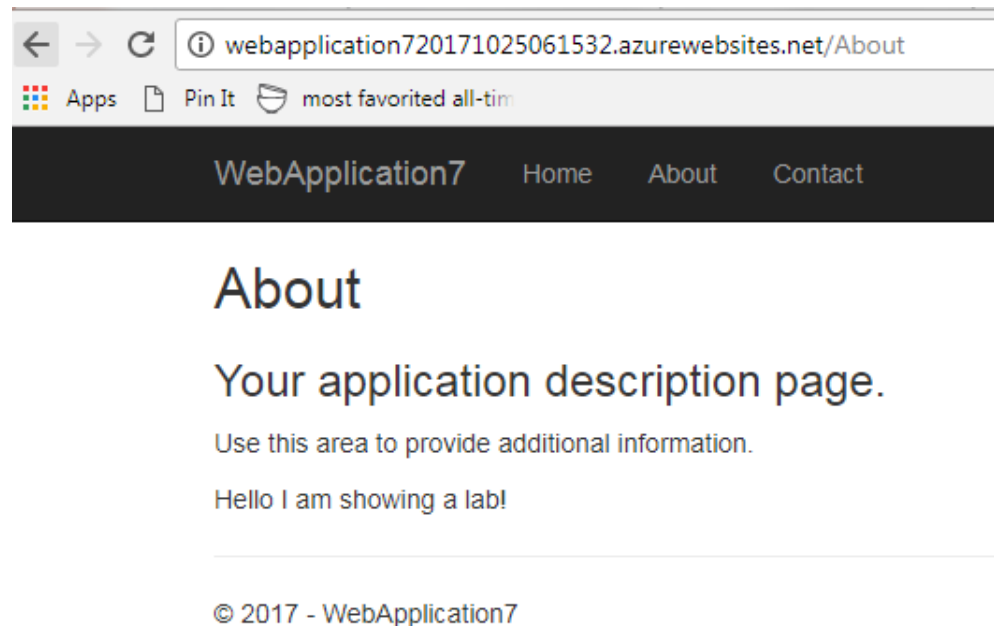


The screenshot shows a code editor with a dark theme. The top bar displays the file name 'About.cshtml' and the project name 'WebApplication7'. The code is as follows:

```
1 @page
2 @model AboutModel
3 @{
4     ViewData["Title"] = "About";
5 }
6 <h2>@ViewData["Title"]</h2>
7 <h3>@Model.Message</h3>
8
9 <p>Use this area to provide additional information.</p>
10 <p>Hello I am showing a lab!</p>
11
```

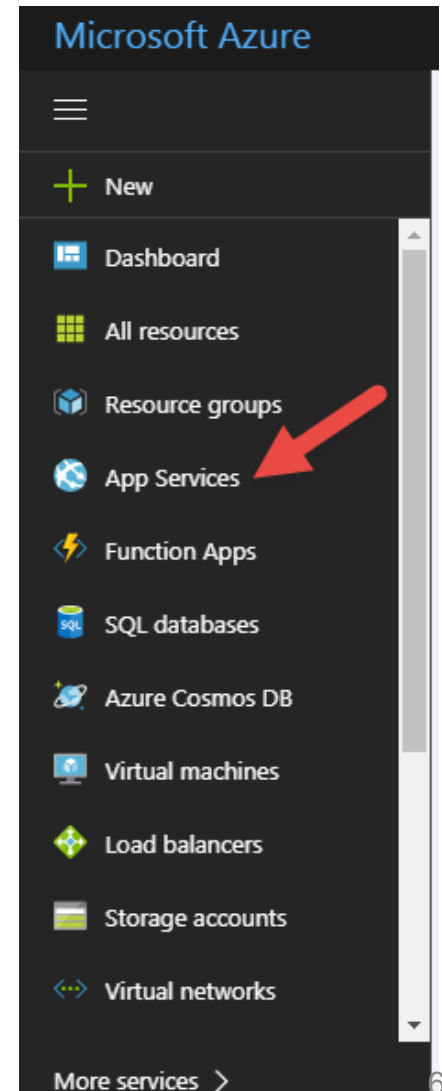
Updating the App

- Redeploy to Azure by right-clicking on the project in your Solution Explorer and selecting 'Publish'
- Note: Make sure that you close the browser from the last time you published or your publishing will fail



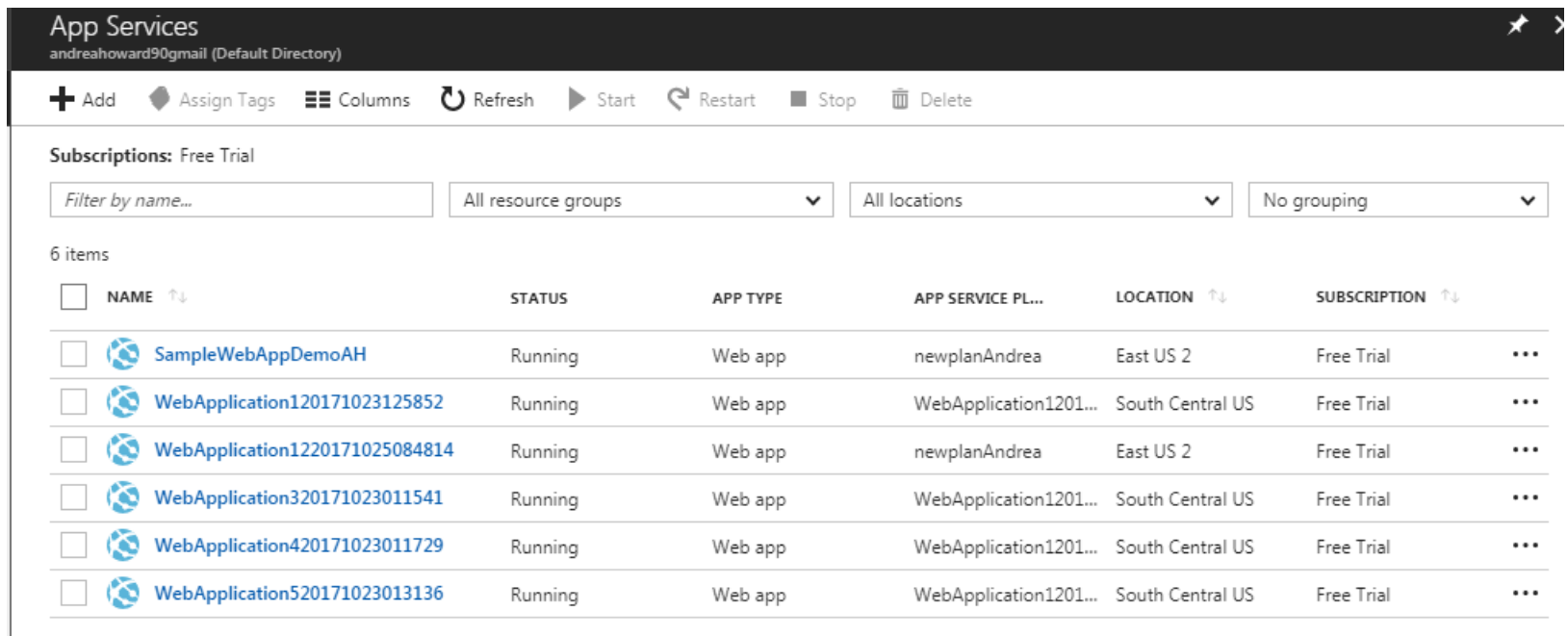
View Web App in Azure

- In Azure Select App Services









View Web App in Azure

- You can see a list of your App Services

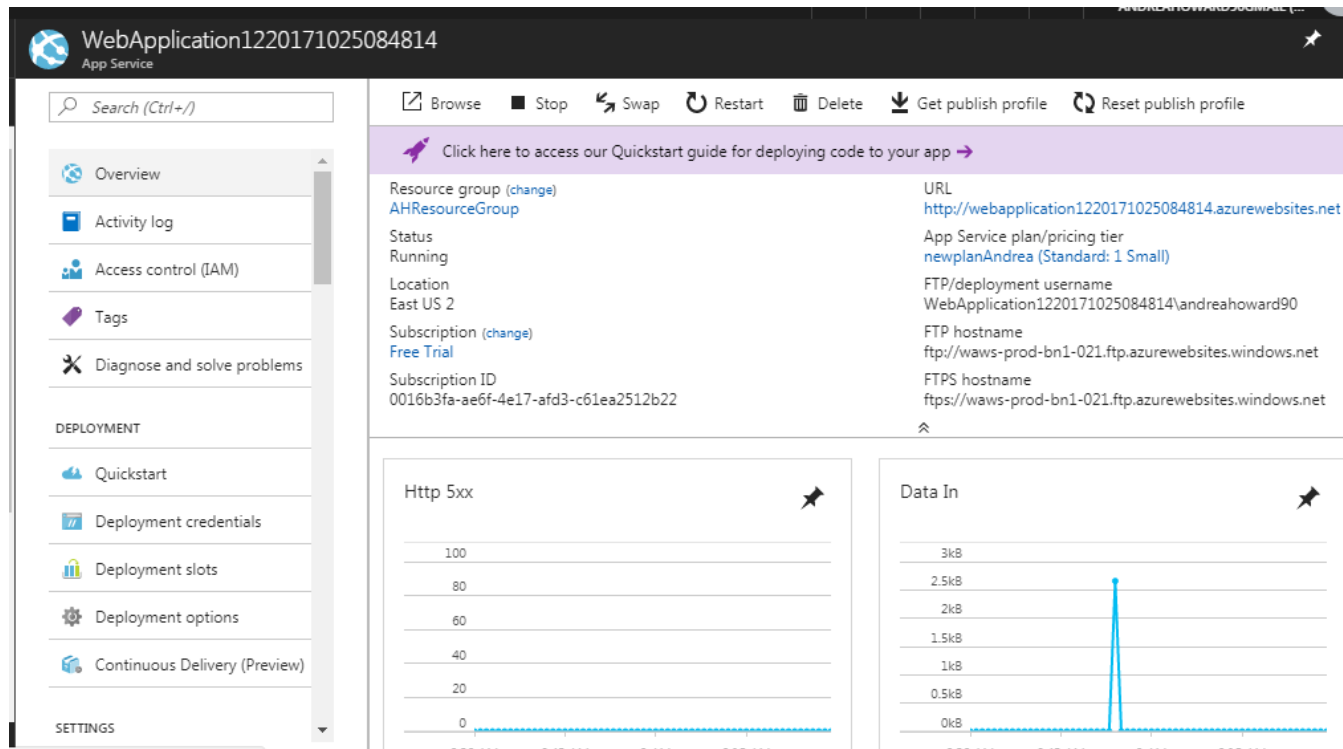


The screenshot displays the Azure App Services management interface. At the top, the header reads 'App Services' with the user 'andreahoward90gmail (Default Directory)'. Below the header is a toolbar with icons for '+ Add', 'Assign Tags', 'Columns', 'Refresh', 'Start', 'Restart', 'Stop', and 'Delete'. The main section is titled 'Subscriptions: Free Trial' and includes filter controls: 'Filter by name...', 'All resource groups', 'All locations', and 'No grouping'. Below these filters, it states '6 items' and presents a table of web applications.

<input type="checkbox"/>	NAME ↑↓	STATUS	APP TYPE	APP SERVICE PL...	LOCATION ↑↓	SUBSCRIPTION ↑↓	
<input type="checkbox"/>	 SampleWebAppDemoAH	Running	Web app	newplanAndrea	East US 2	Free Trial	...
<input type="checkbox"/>	 WebApplication120171023125852	Running	Web app	WebApplication1201...	South Central US	Free Trial	...
<input type="checkbox"/>	 WebApplication1220171025084814	Running	Web app	newplanAndrea	East US 2	Free Trial	...
<input type="checkbox"/>	 WebApplication320171023011541	Running	Web app	WebApplication1201...	South Central US	Free Trial	...
<input type="checkbox"/>	 WebApplication420171023011729	Running	Web app	WebApplication1201...	South Central US	Free Trial	...
<input type="checkbox"/>	 WebApplication520171023013136	Running	Web app	WebApplication1201...	South Central US	Free Trial	...

Create an asp.net web app

- Select your web app and view any information about the app



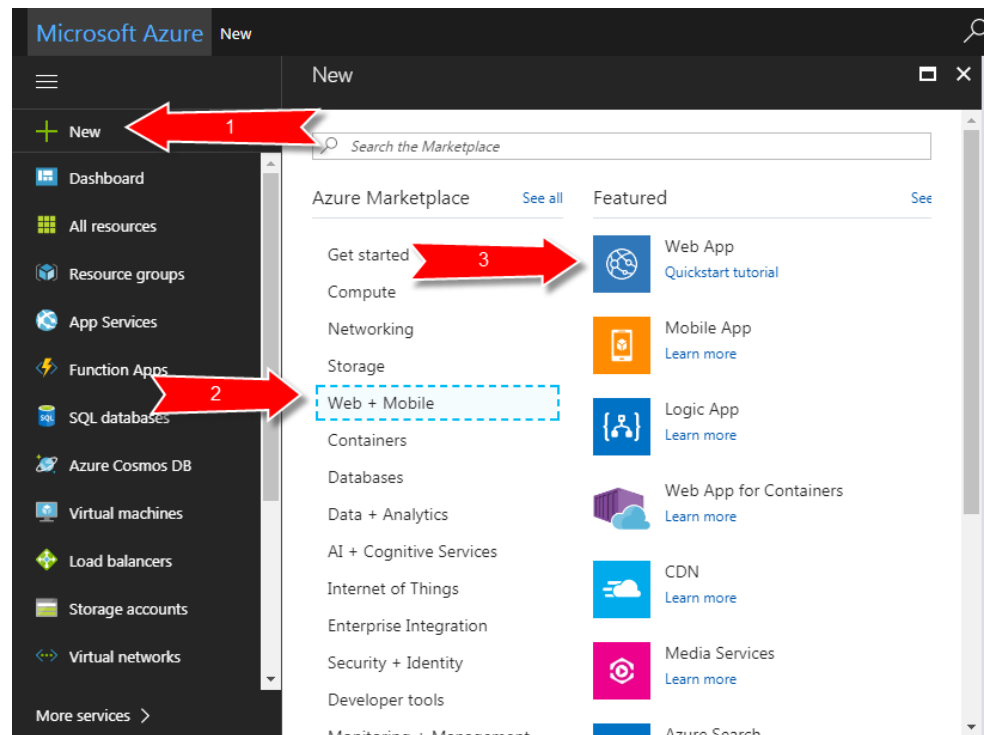
Delete Resources in Azure

- Finally close/delete your web apps and your resources so to ensure you don't get charged

4. Create a web app in Azure and Update the web app in VS

Create a web app in Azure

- Login to your Azure account and then Select 'New' then 'Web + Mobile' then 'Web App'



Create a web app in Azure

- Enter a name for your web app and then select 'Create'

Microsoft Azure New > Web App

Web App
Create

* App name
LabWebAppAH ✓
.azurewebsites.net

* Subscription
Free Trial

* Resource Group ⓘ
☒ Create new ☐ Use existing
LabWebAppAH ✓

* OS Windows Linux

* App Service plan/Location
ServicePlancef76fd3-a10f(South ... >

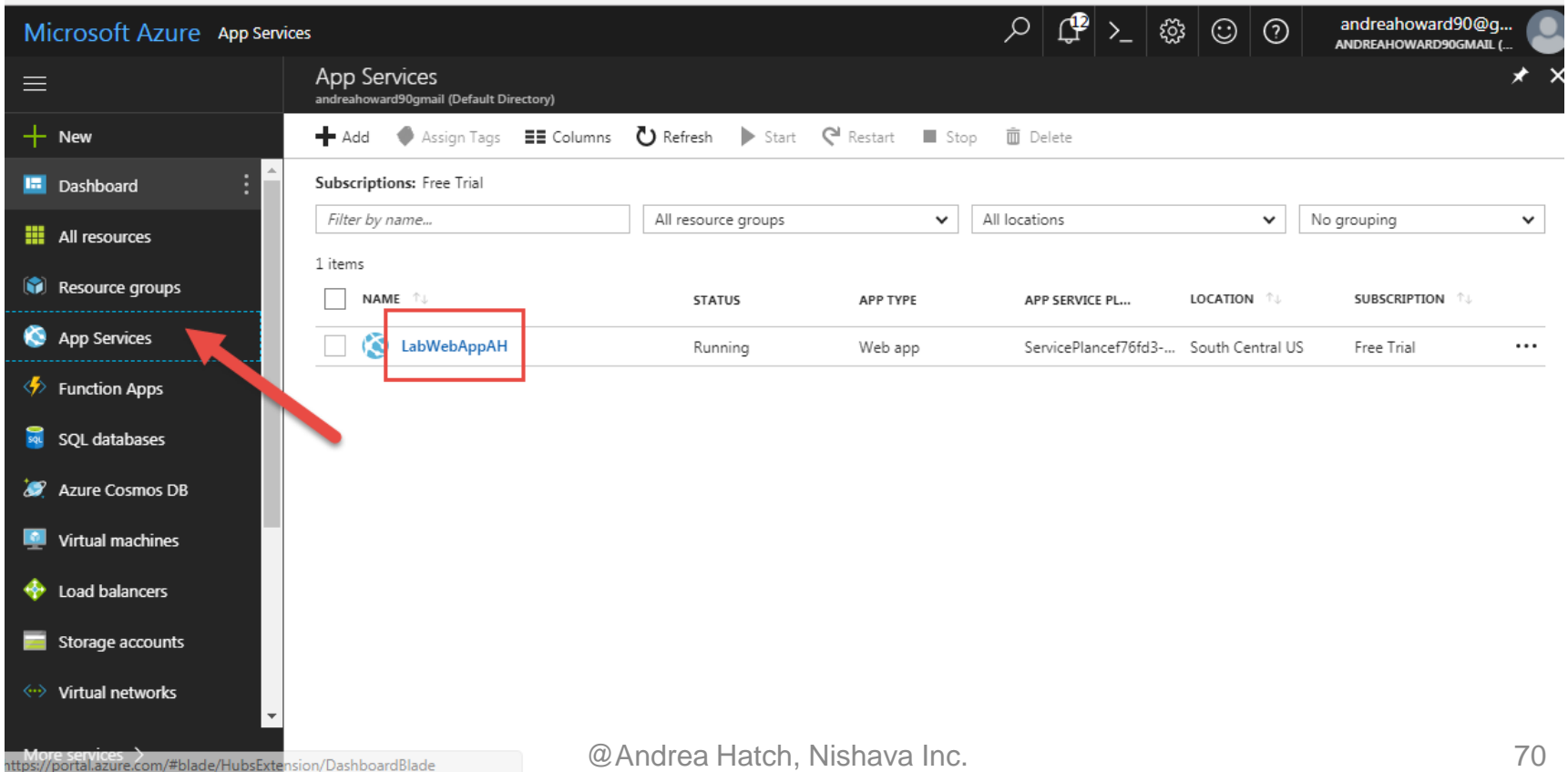
Application Insights ⓘ On Off

☐ Pin to dashboard

More services > Create Automation options

Create a web app in Azure

- You can see your Web App was created by clicking App Services

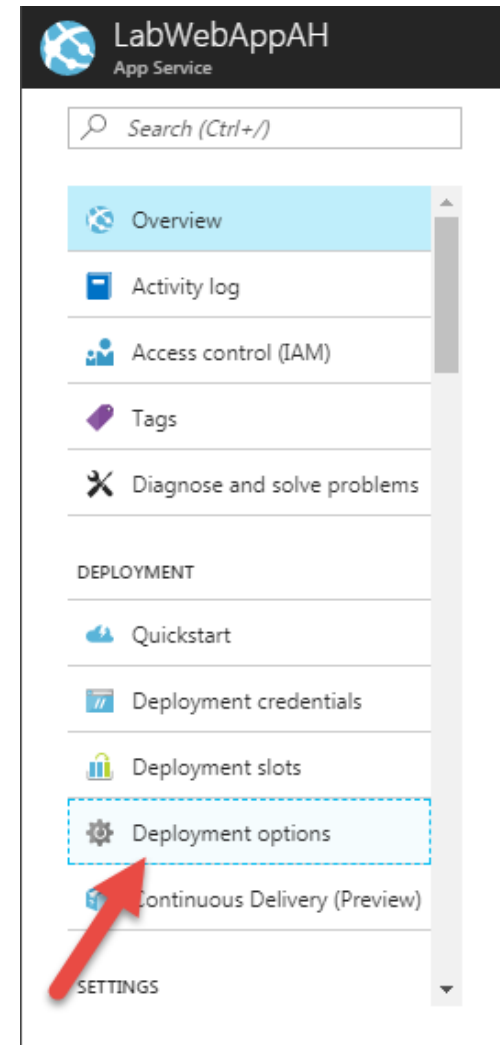


The screenshot shows the Microsoft Azure App Services dashboard. The left navigation pane lists various services, with 'App Services' highlighted and a red arrow pointing to it. The main content area displays a table of web apps under the 'Subscriptions: Free Trial' filter. The table has columns for NAME, STATUS, APP TYPE, APP SERVICE PL..., LOCATION, and SUBSCRIPTION. One item is listed: 'LabWebAppAH', which is highlighted with a red box. The status is 'Running', the app type is 'Web app', and the subscription is 'Free Trial'.

NAME	STATUS	APP TYPE	APP SERVICE PL...	LOCATION	SUBSCRIPTION
LabWebAppAH	Running	Web app	ServicePlancef76fd3-...	South Central US	Free Trial

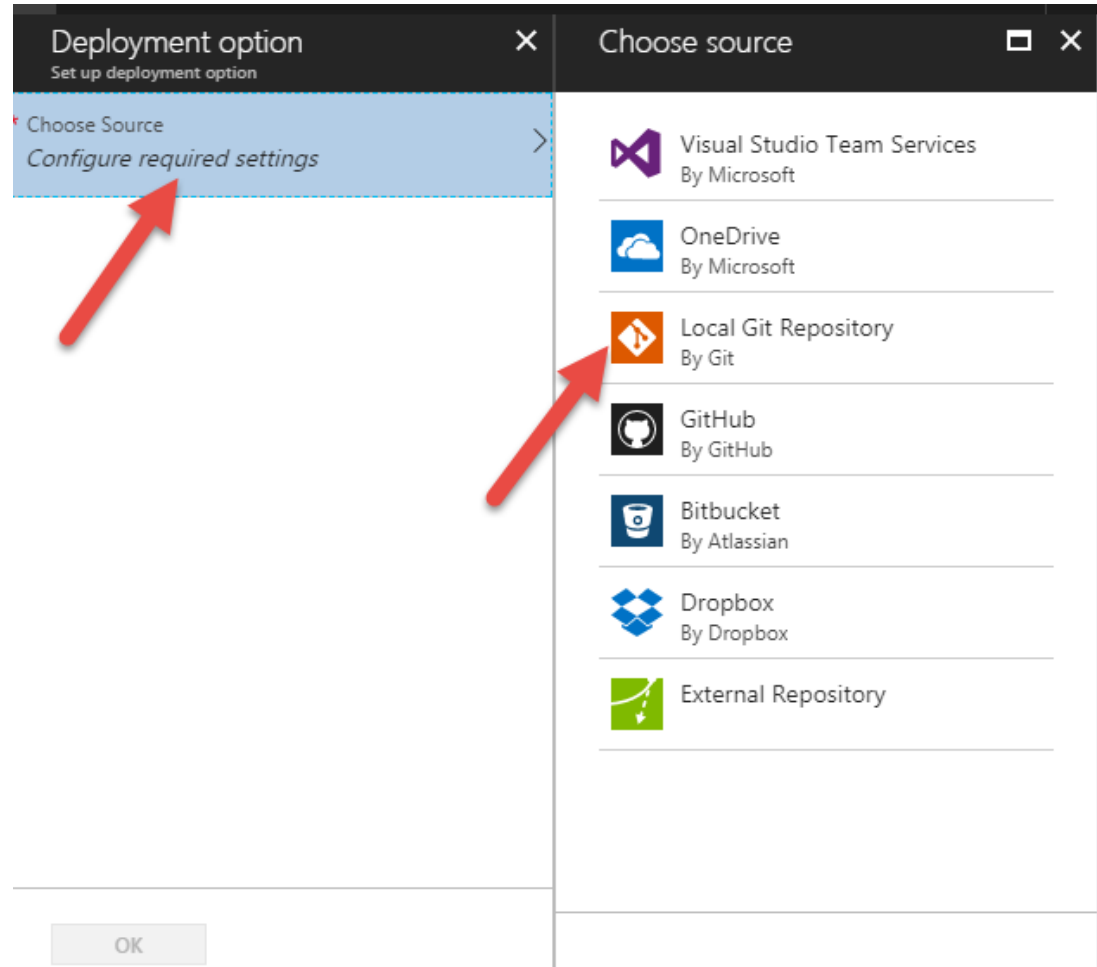
Create a web app in Azure

- Under the Deployment section of your web app select 'Deployment Options'



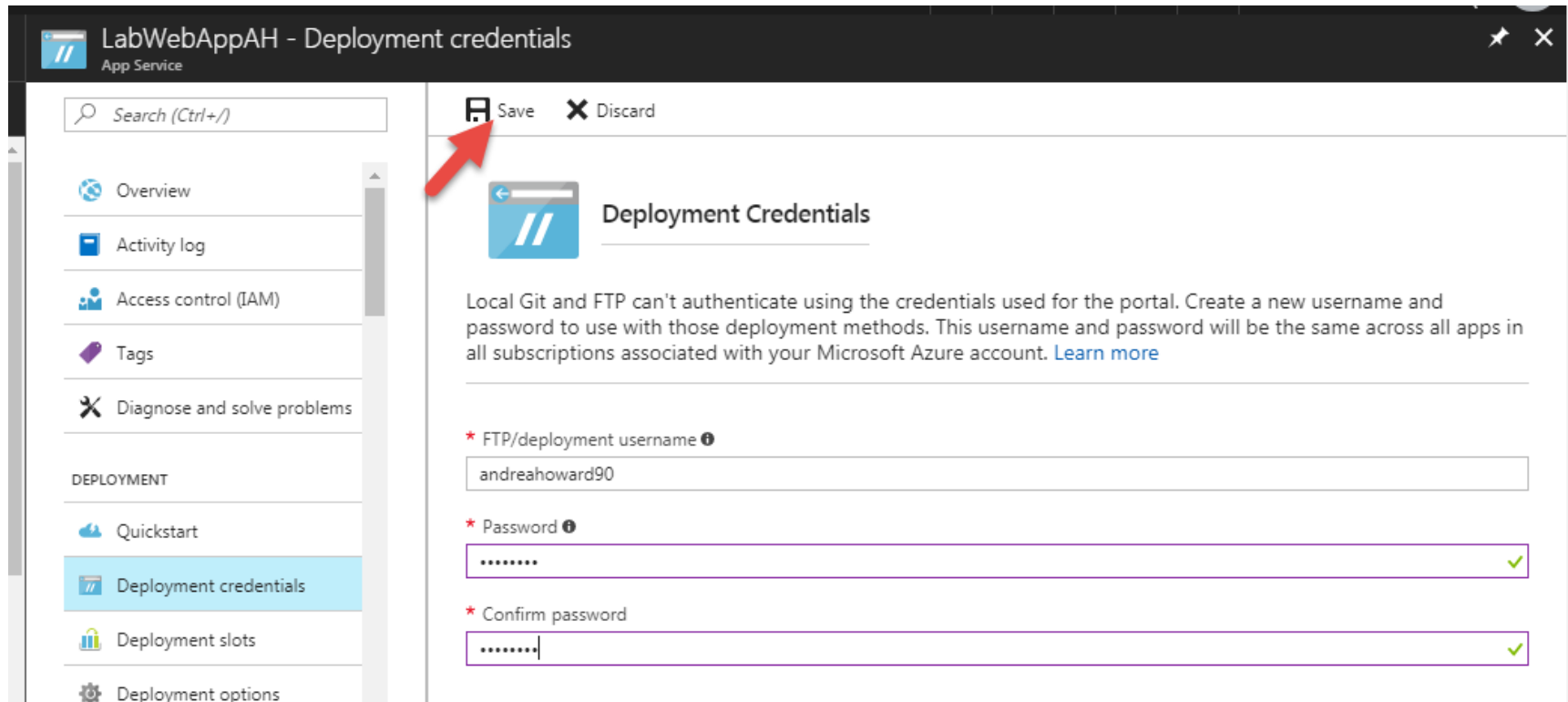
Create a web app in Azure

- Select 'Choose Source' and then select 'Local Git Repository' and then select 'OK'



Create a web app in Azure

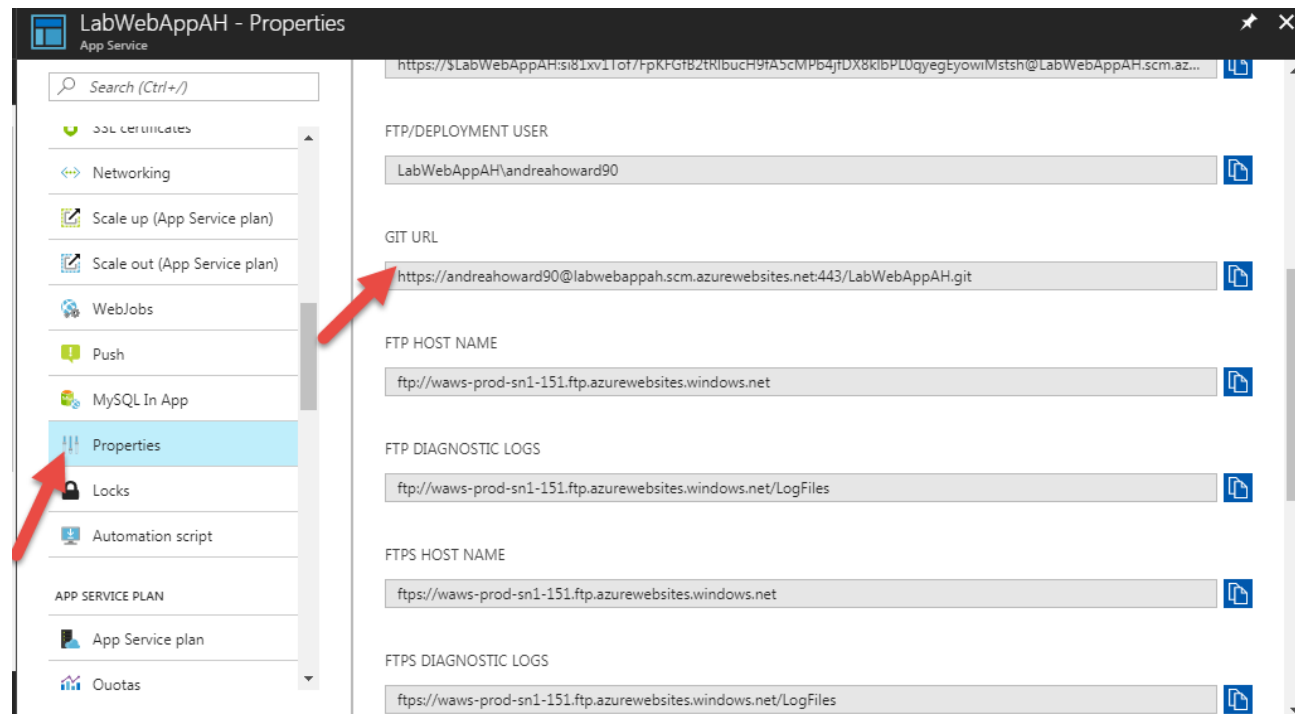
- Set up deployment credentials under the Deployment section and then select 'Save'



The screenshot shows the 'LabWebAppAH - Deployment credentials' page in the Azure App Service portal. The left sidebar contains navigation links: Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, and a 'DEPLOYMENT' section with Quickstart, Deployment credentials (highlighted), Deployment slots, and Deployment options. The main content area has a 'Save' button and a 'Discard' button at the top. Below them is the 'Deployment Credentials' section, which includes a description: 'Local Git and FTP can't authenticate using the credentials used for the portal. Create a new username and password to use with those deployment methods. This username and password will be the same across all apps in all subscriptions associated with your Microsoft Azure account. [Learn more](#)'. There are three required fields: 'FTP/deployment username' (containing 'andreahoward90'), 'Password' (masked with dots and a green checkmark), and 'Confirm password' (masked with dots and a green checkmark).

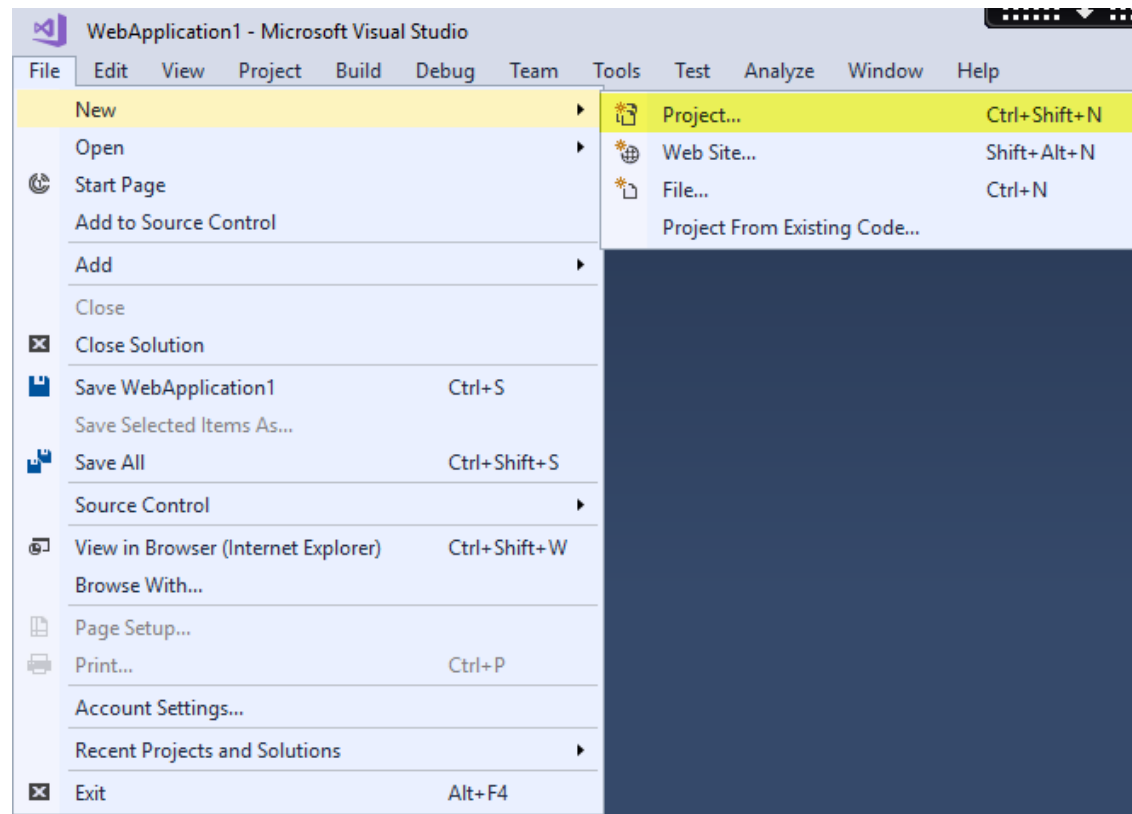
Create a web app in Azure

- Under Settings Properties you will be able to find information about your web app. Copy the GIT URL as we will need this to connect to your app in VS.



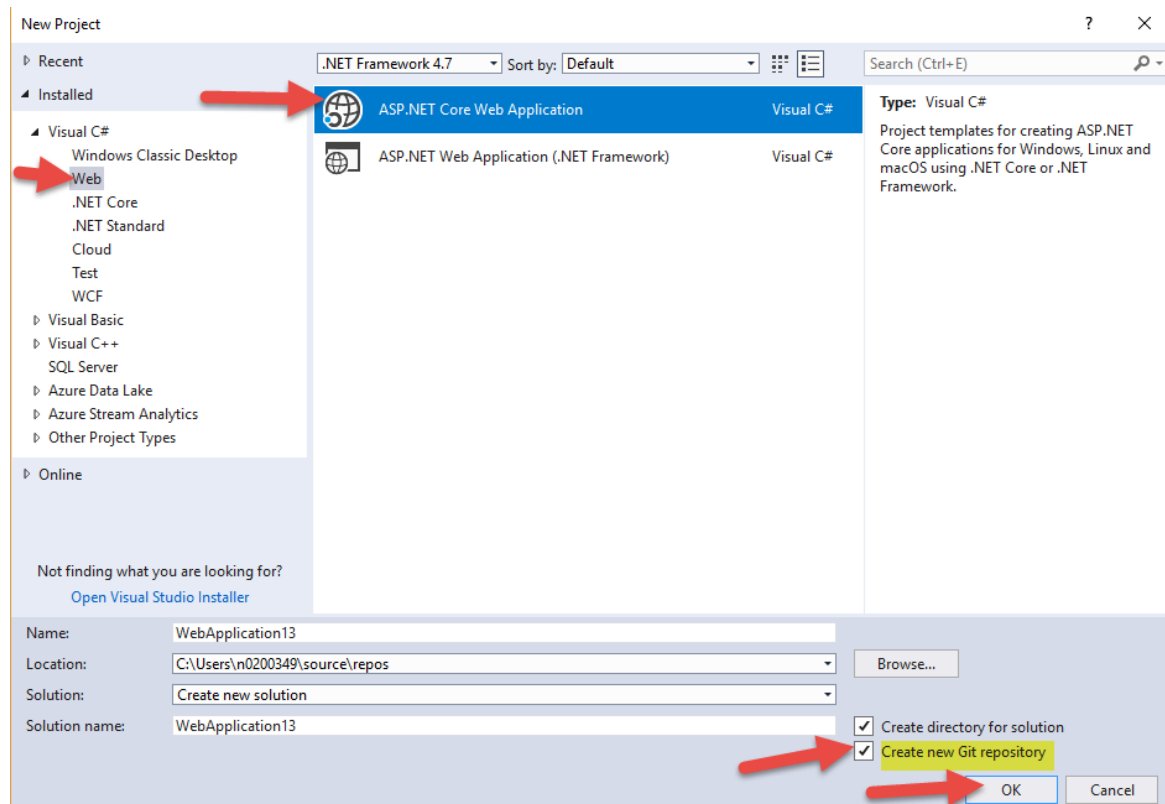
Create a project in VS

- In VS, select File -> New -> Project



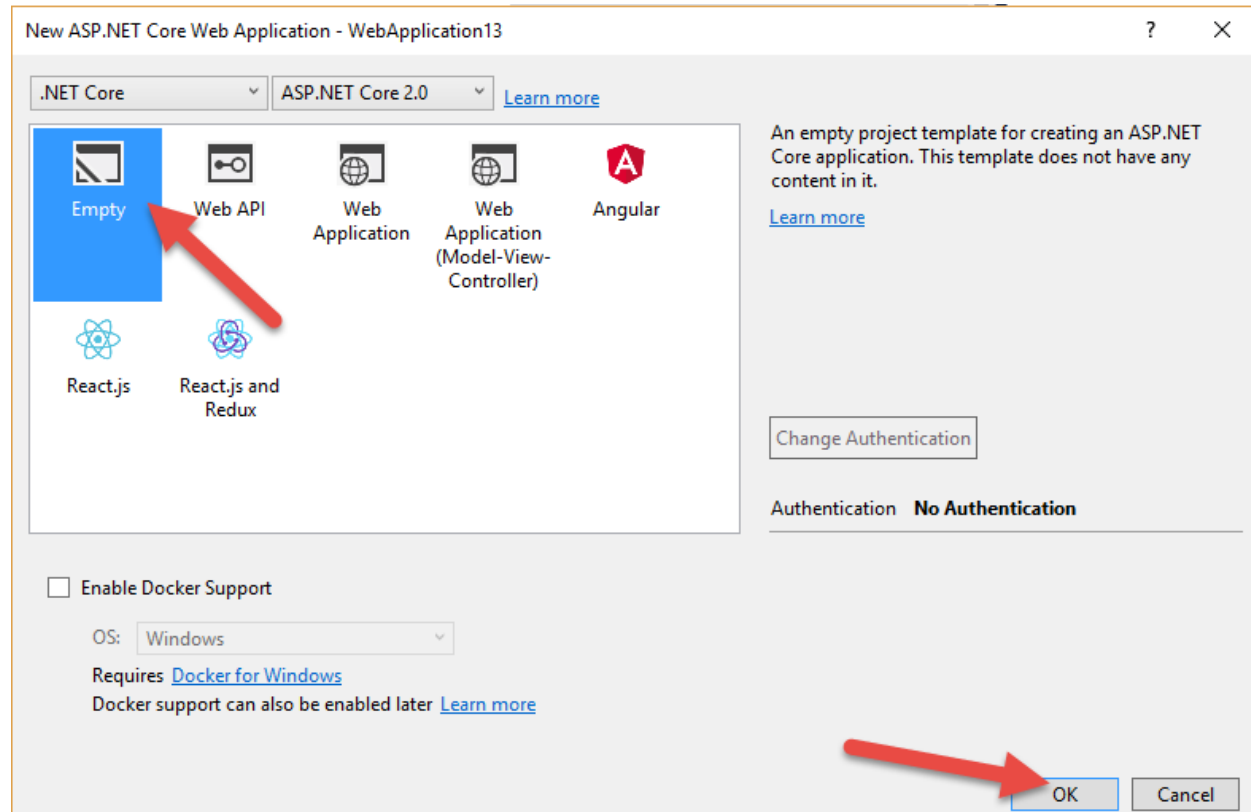
Create a project in VS

- Under Web, select the ASP.NET Core Web App, give the app a Name and Make sure Create new Git Repository is selected then select 'OK'



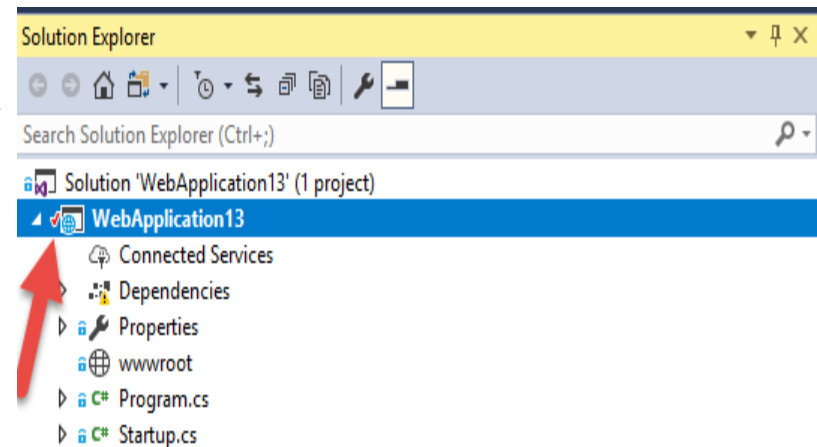
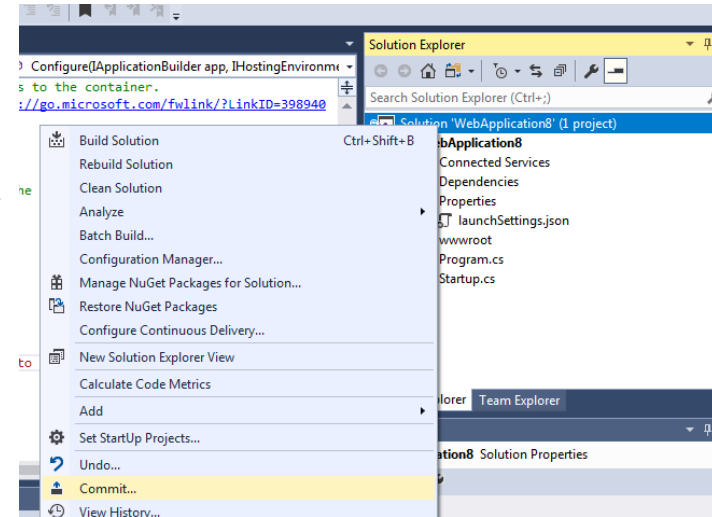
Create a project in VS

- Select 'Empty' then select 'OK'



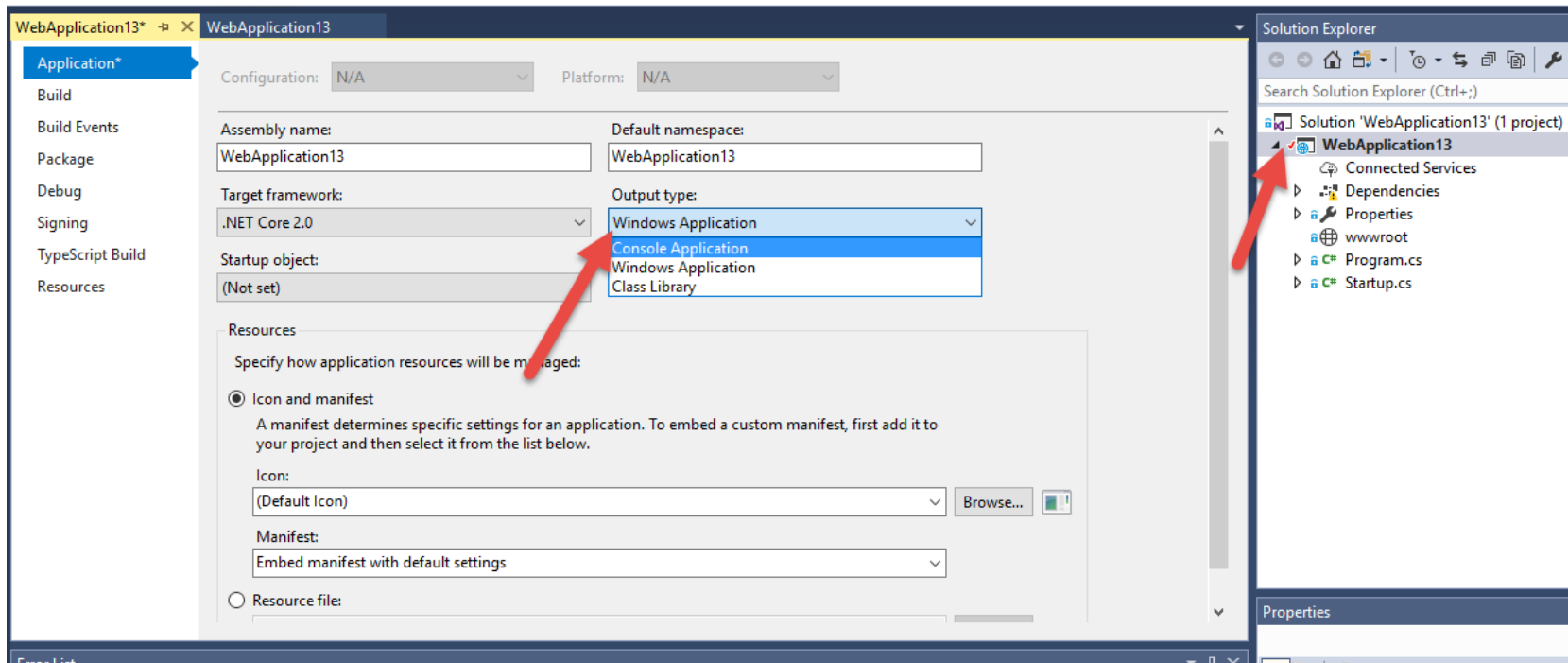
Update your web app in VS

- In VS find the Solution Explorer and right click on the Solution project. →
- Check to see if you have the option to Commit. → You will know if you can commit your Web App if there is a red check mark next the Web App name.



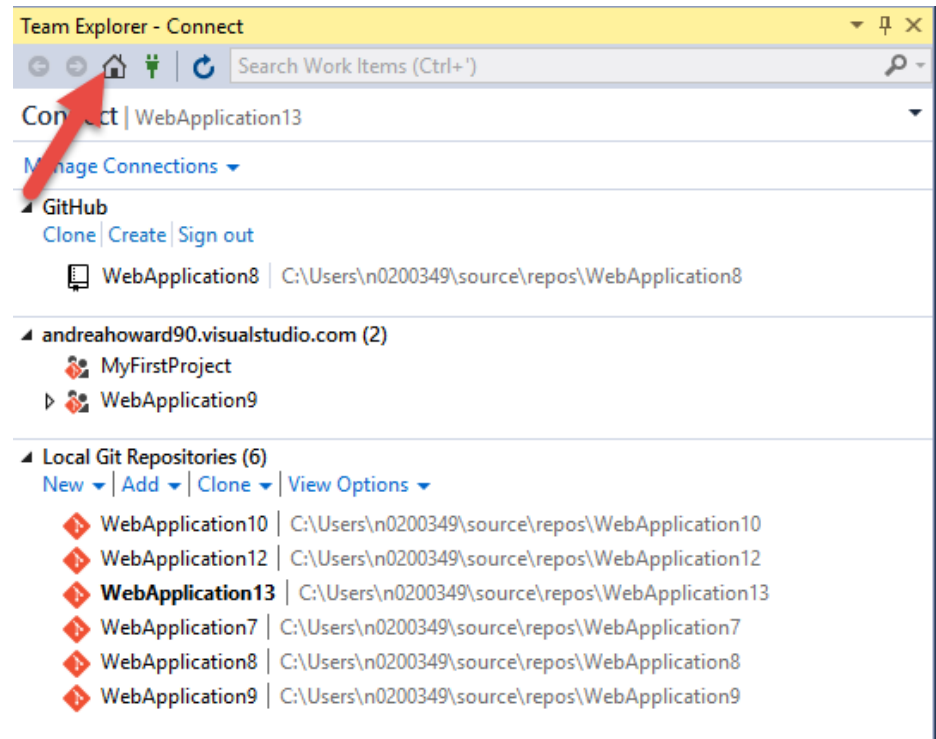
Update your web app in VS

- If you do not have a red check mark you need to make any change to your web app so that it can be committed. For example select the web application and then go under Project -> Properties and change the Output type and this will allow you to commit your changes.



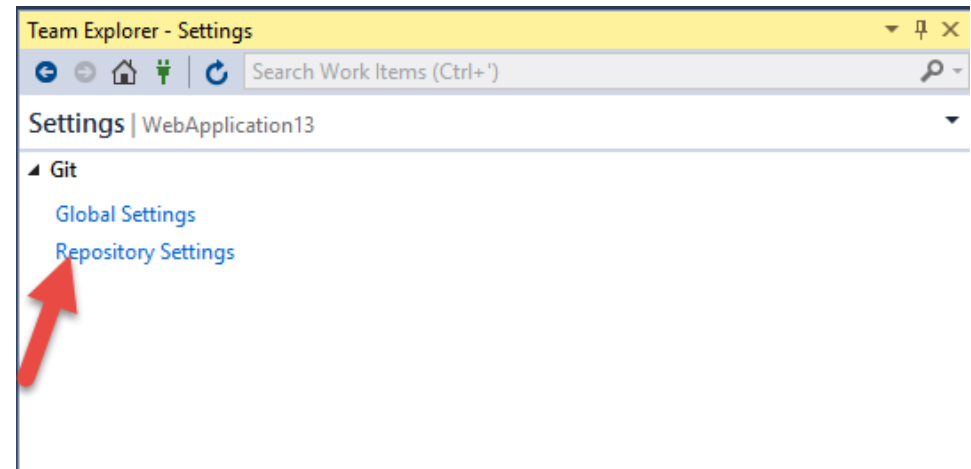
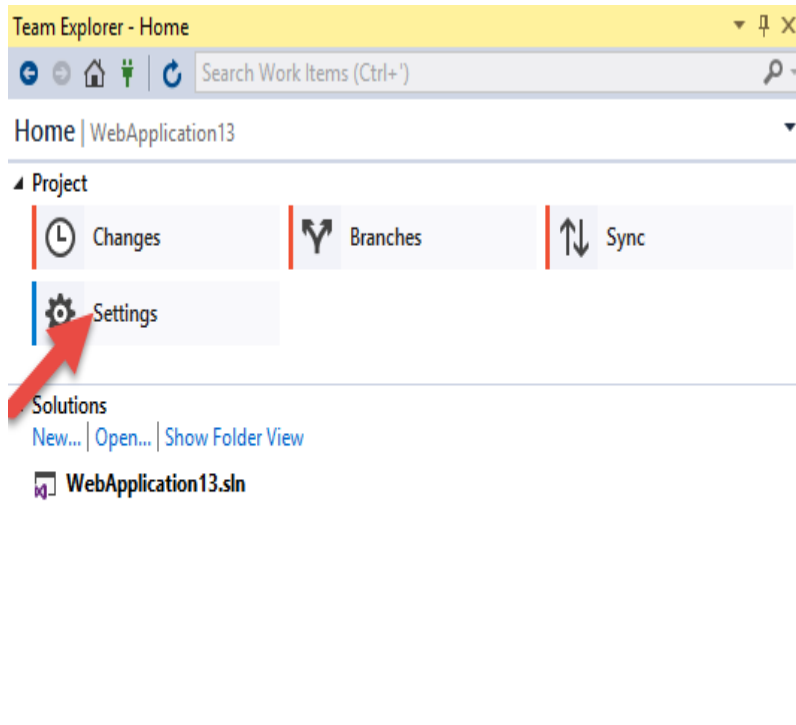
Update your web app in VS

- Once your changes are committed you will be brought to the Team Explorer and you will need to select the home icon



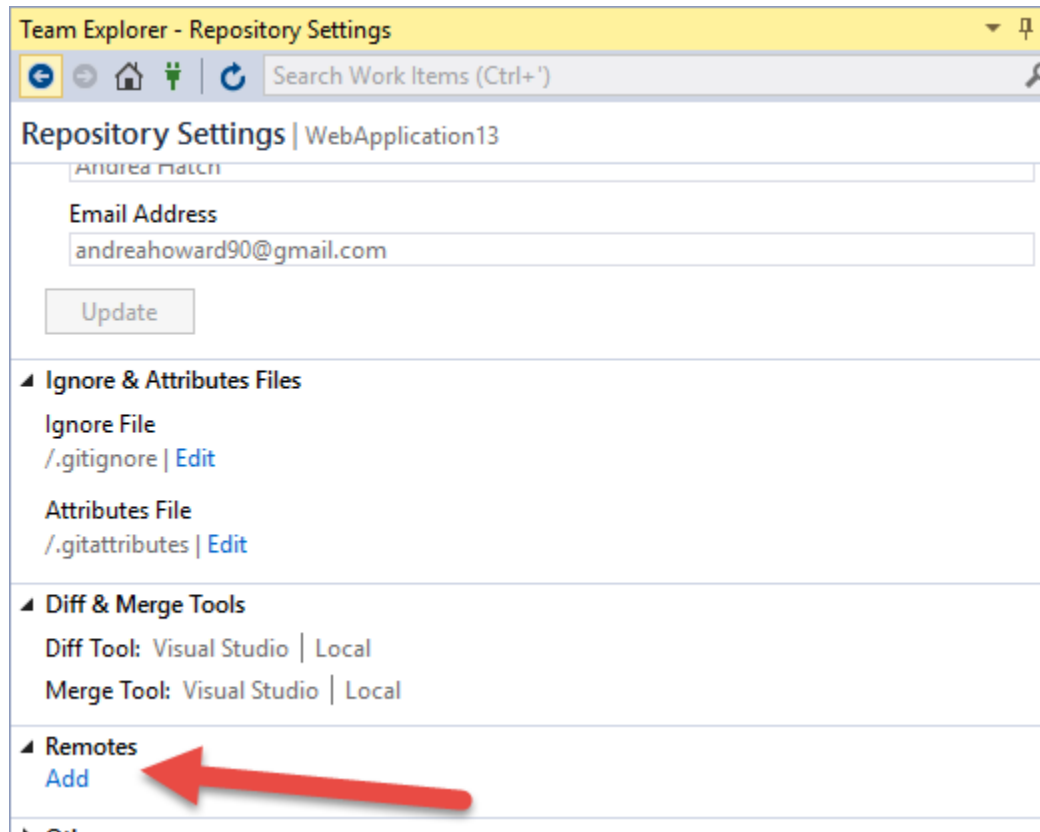
Update your web app in VS

- Select Settings -> Repository Settings



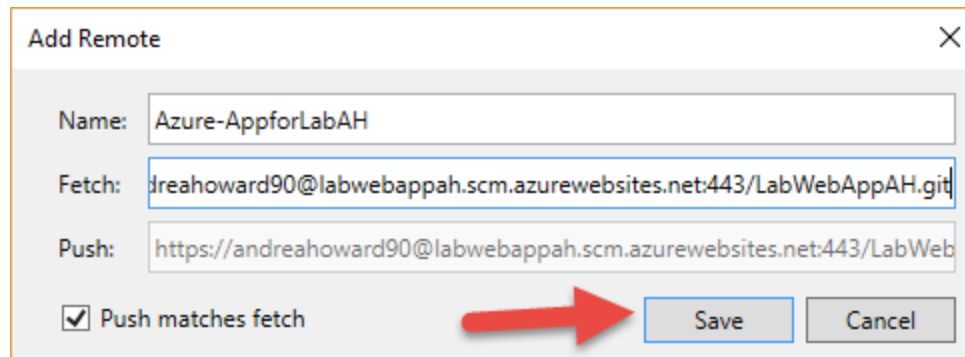
Update your web app in VS

- Under Remotes select 'Add'



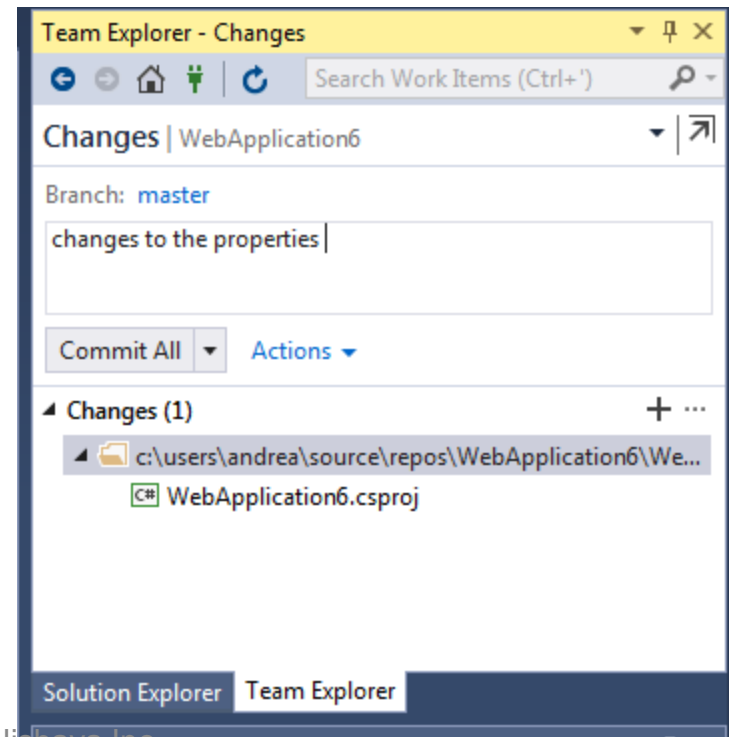
Update your web app in VS

- Give your Remote a name and then paste in your git URL from Azure to the Fetch section and then select 'Save'



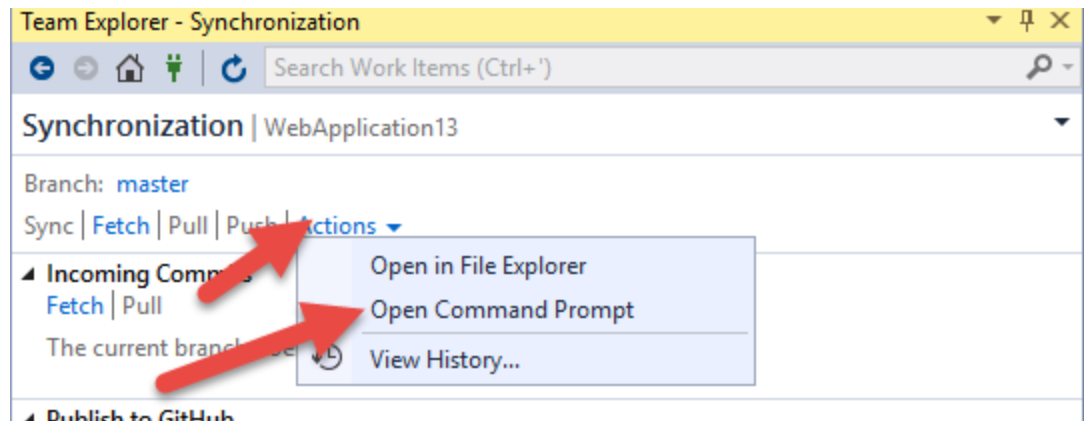
Update your web app in VS

- Select the home icon again and this time select 'Changes' so that we can commit the changes. Enter in a commit message and select 'Commit All'



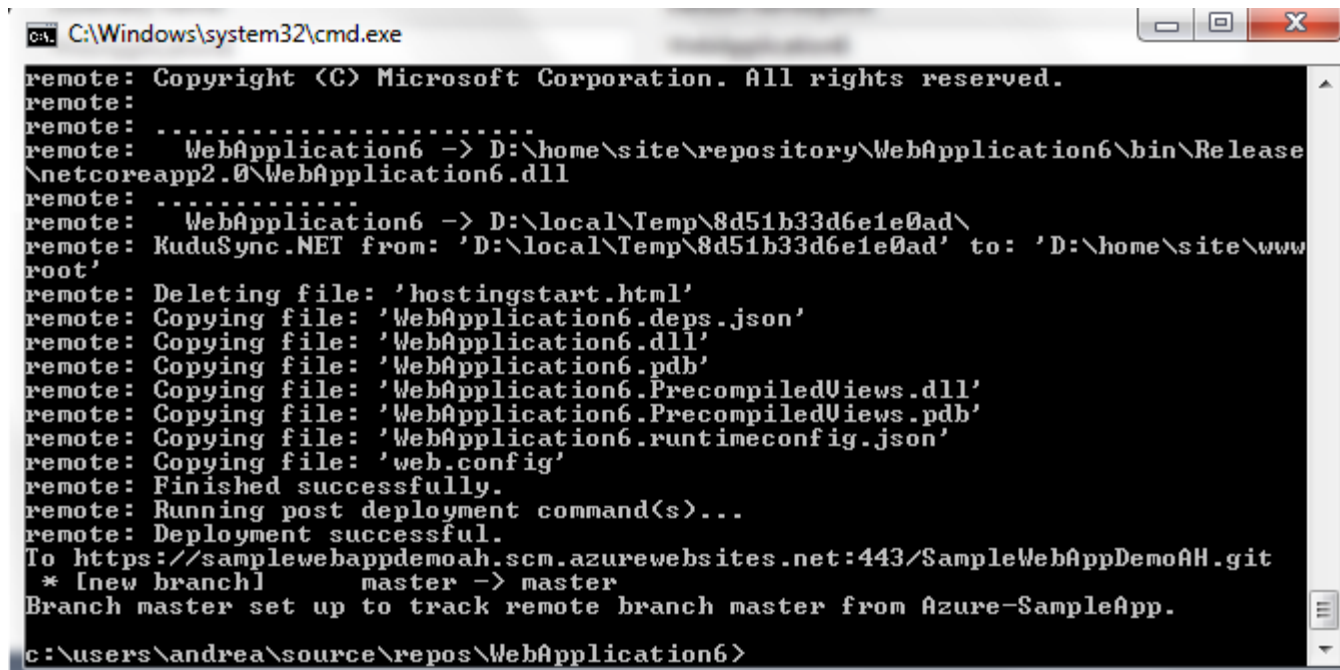
Update your web app in VS

- Open a command prompt to sync your changes with your Web App in Azure by going into Team Explorer and selecting the home icon -> Sync -> Actions -> Open Command Prompt



Update your web app in VS

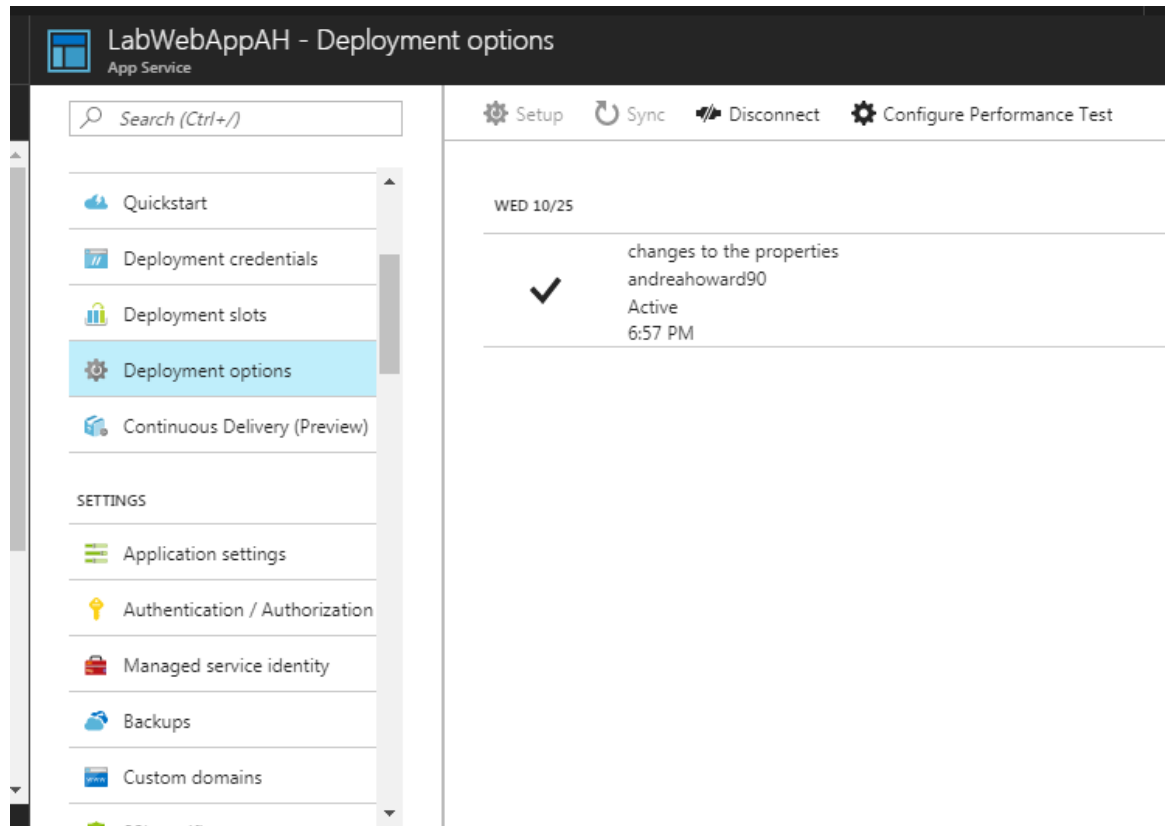
- Enter in: `git push -u (name of Remote) master`. Your app will start syncing and once done will say 'Deployment Successful'
 - Ex: `git push -u Azure-AppforLabAH master`



```
C:\Windows\system32\cmd.exe
remote: Copyright (C) Microsoft Corporation. All rights reserved.
remote:
remote: .....
remote: WebApplication6 -> D:\home\site\repository\WebApplication6\bin\Release
\netcoreapp2.0\WebApplication6.dll
remote: .....
remote: WebApplication6 -> D:\local\Temp\8d51b33d6e1e0ad\
remote: KuduSync.NET from: 'D:\local\Temp\8d51b33d6e1e0ad' to: 'D:\home\site\www
root'
remote: Deleting file: 'hostingstart.html'
remote: Copying file: 'WebApplication6.deps.json'
remote: Copying file: 'WebApplication6.dll'
remote: Copying file: 'WebApplication6.pdb'
remote: Copying file: 'WebApplication6.PrecompiledViews.dll'
remote: Copying file: 'WebApplication6.PrecompiledViews.pdb'
remote: Copying file: 'WebApplication6.runtimeconfig.json'
remote: Copying file: 'web.config'
remote: Finished successfully.
remote: Running post deployment command(s)...
remote: Deployment successful.
To https://samplewebappdemoah.scm.azurewebsites.net:443/SampleWebAppDemoAH.git
 * [new branch]      master -> master
Branch master set up to track remote branch master from Azure-SampleApp.
c:\users\andrea\source\repos\WebApplication6>
```

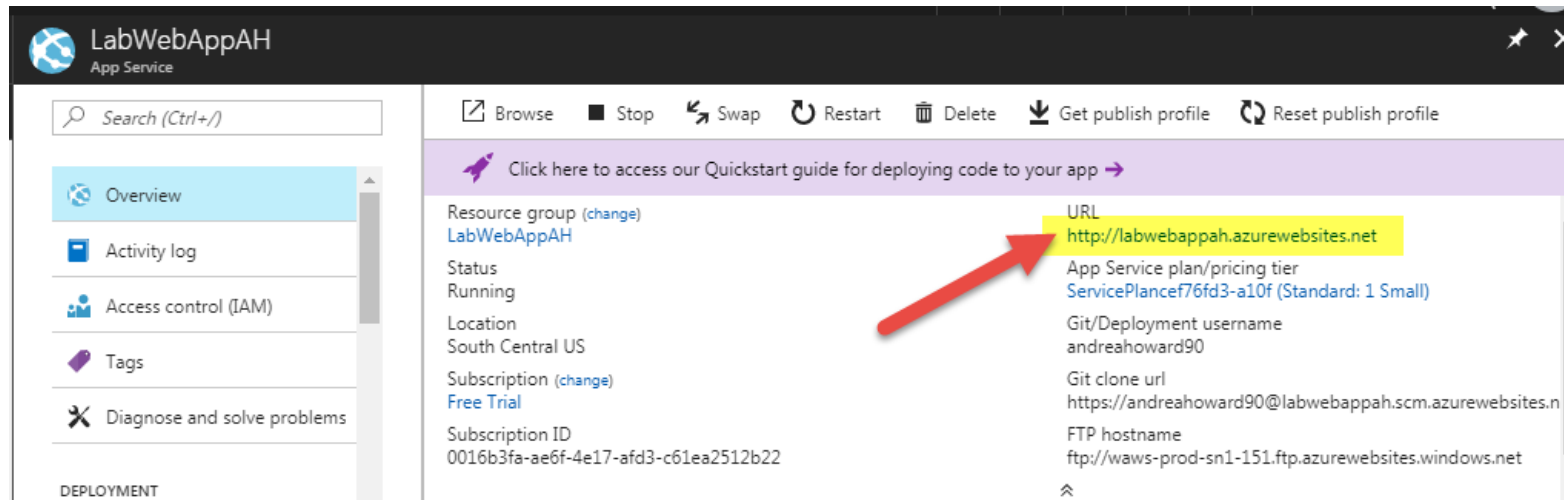
Update your web app in VS

- Verify your deployment in Azure by selecting your Web App and then Deployment options



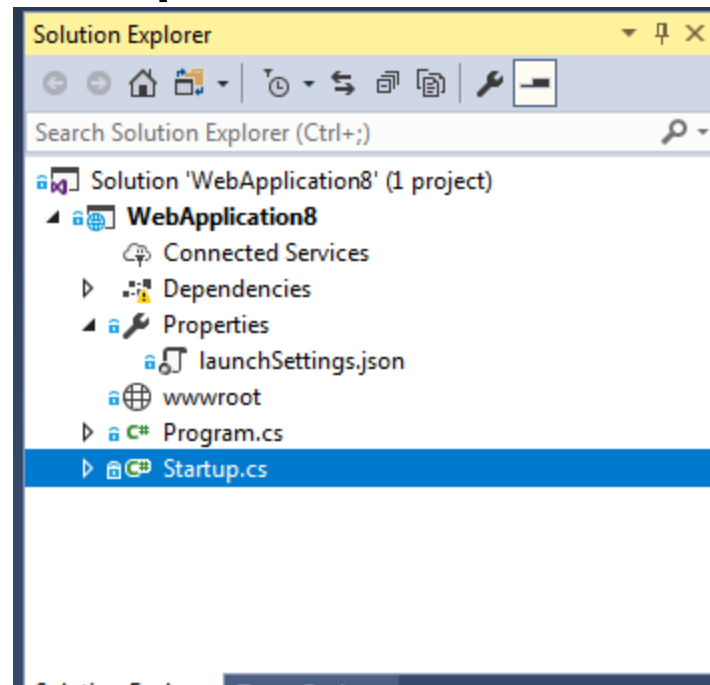
Update your web app in VS

- To run the app you can select Overview and see the URL to your app.



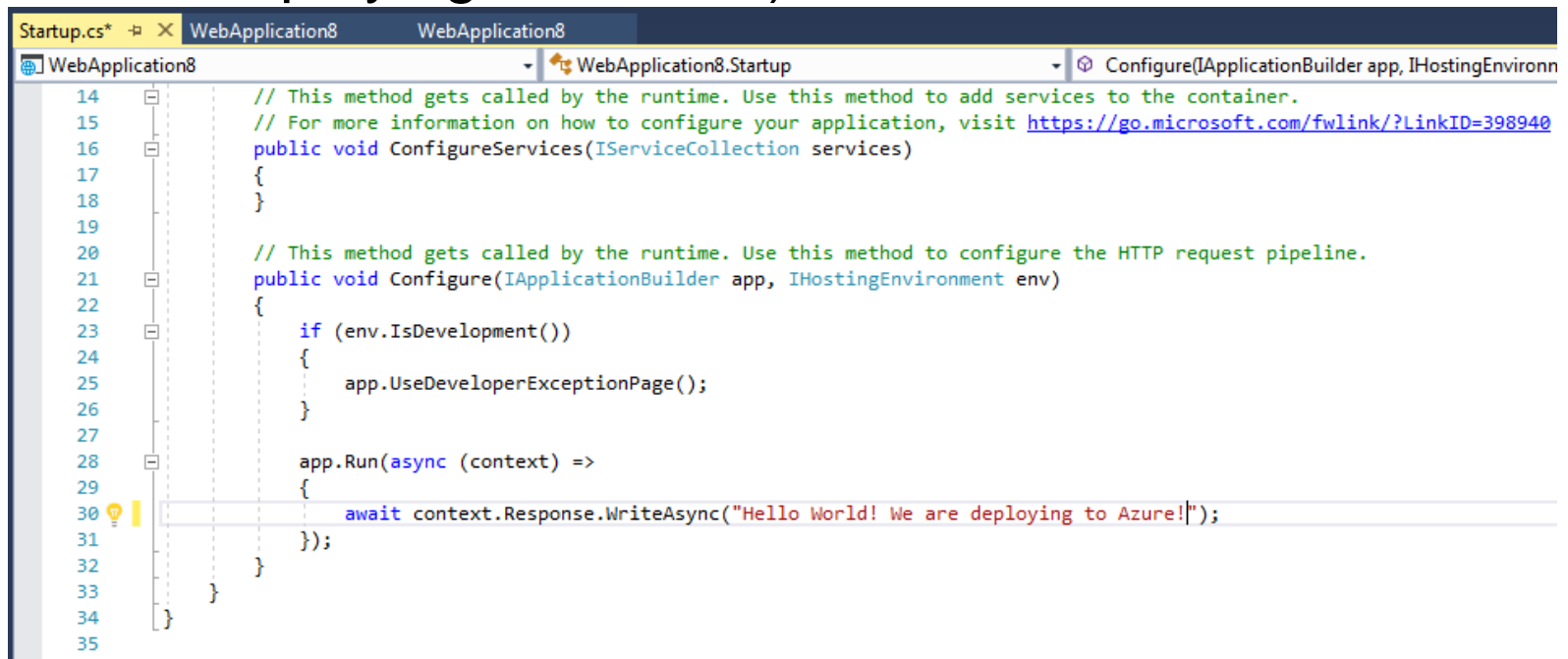
Update your web app in VS

- You can update your app again in VS by going to the Solution Explorer and opening up the Startup.cs file



Update your web app in VS

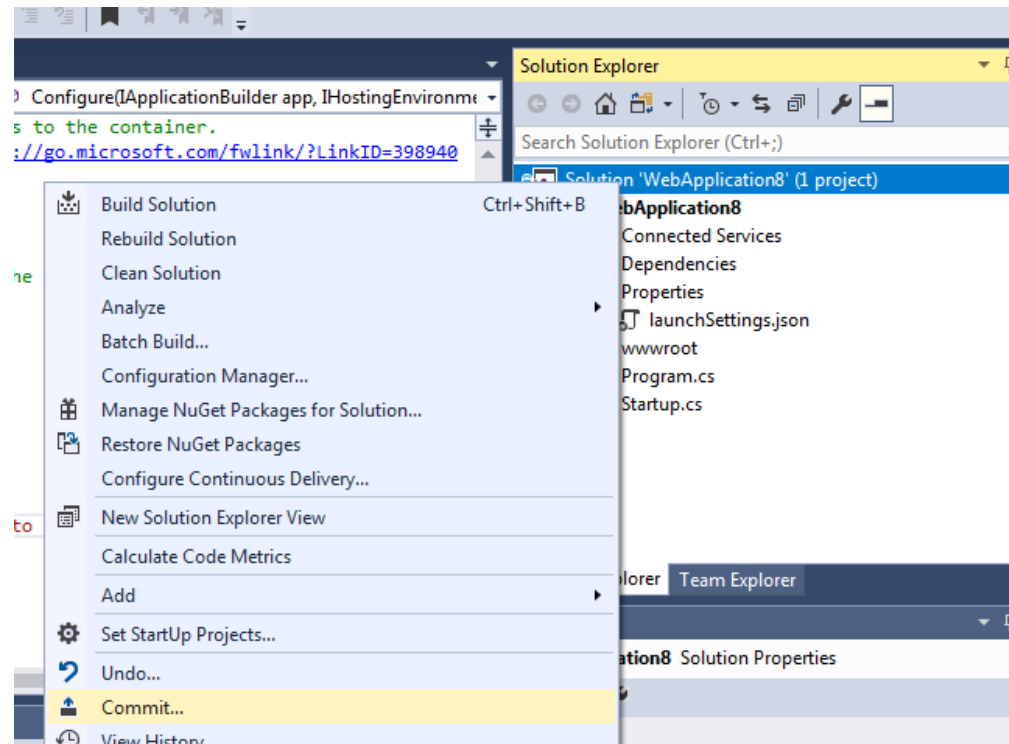
- In the Configure method modify Response.WriteAsync and type a message and then select Save:
 - Ex: `await context.Response.WriteAsync("Hello World! We are deploying to Azure.");`

A screenshot of the Visual Studio IDE showing the 'Startup.cs' file for a project named 'WebApplication8'. The file is open in the 'WebApplication8.Startup' tab. The code is in C# and shows the 'Configure' method being edited. The method signature is 'public void Configure(IApplicationBuilder app, IHostingEnvironment env)'. Inside the method, there is a conditional block for development mode. The line being edited is 'await context.Response.WriteAsync("Hello World! We are deploying to Azure!");'. A lightbulb icon is visible next to this line, indicating a suggestion or tip. The line numbers 14 through 35 are visible on the left side of the editor.

```
14 // This method gets called by the runtime. Use this method to add services to the container.  
15 // For more information on how to configure your application, visit https://go.microsoft.com/fwlink/?LinkID=398940  
16 public void ConfigureServices(IServiceCollection services)  
17 {  
18 }  
19  
20 // This method gets called by the runtime. Use this method to configure the HTTP request pipeline.  
21 public void Configure(IApplicationBuilder app, IHostingEnvironment env)  
22 {  
23     if (env.IsDevelopment())  
24     {  
25         app.UseDeveloperExceptionPage();  
26     }  
27  
28     app.Run(async (context) =>  
29     {  
30         await context.Response.WriteAsync("Hello World! We are deploying to Azure!");  
31     });  
32 }  
33  
34 }  
35
```

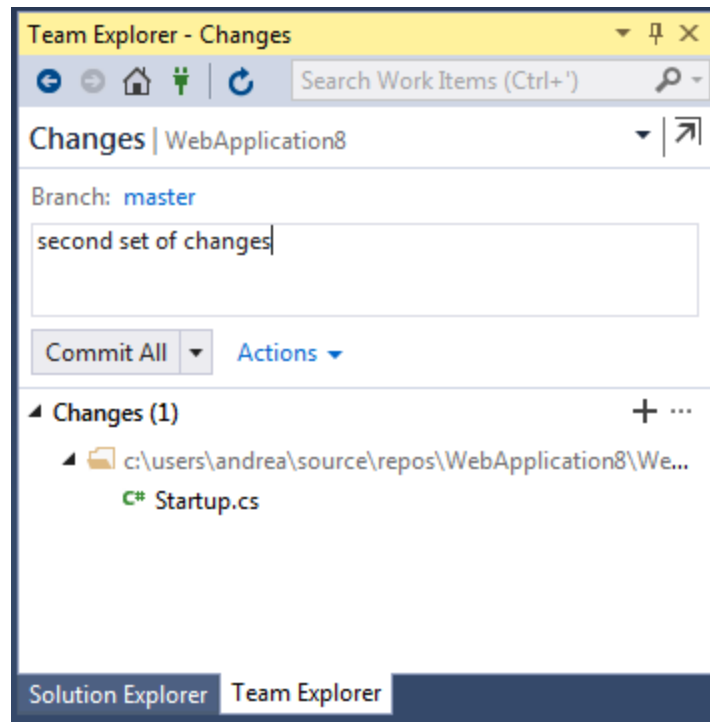
Update your web app in VS

- Make sure to save your changes and then right click again on the Solution Web App name in VS and select 'Commit'. Enter your commit message and then press the Commit button.



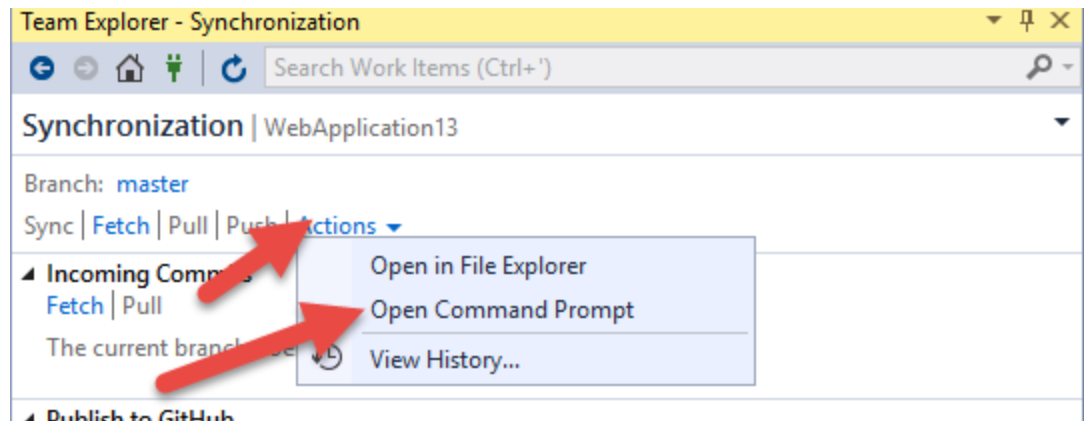
Update your web app in VS

- In the Team explorer click on the home icon -> changes then type a comment and Commit



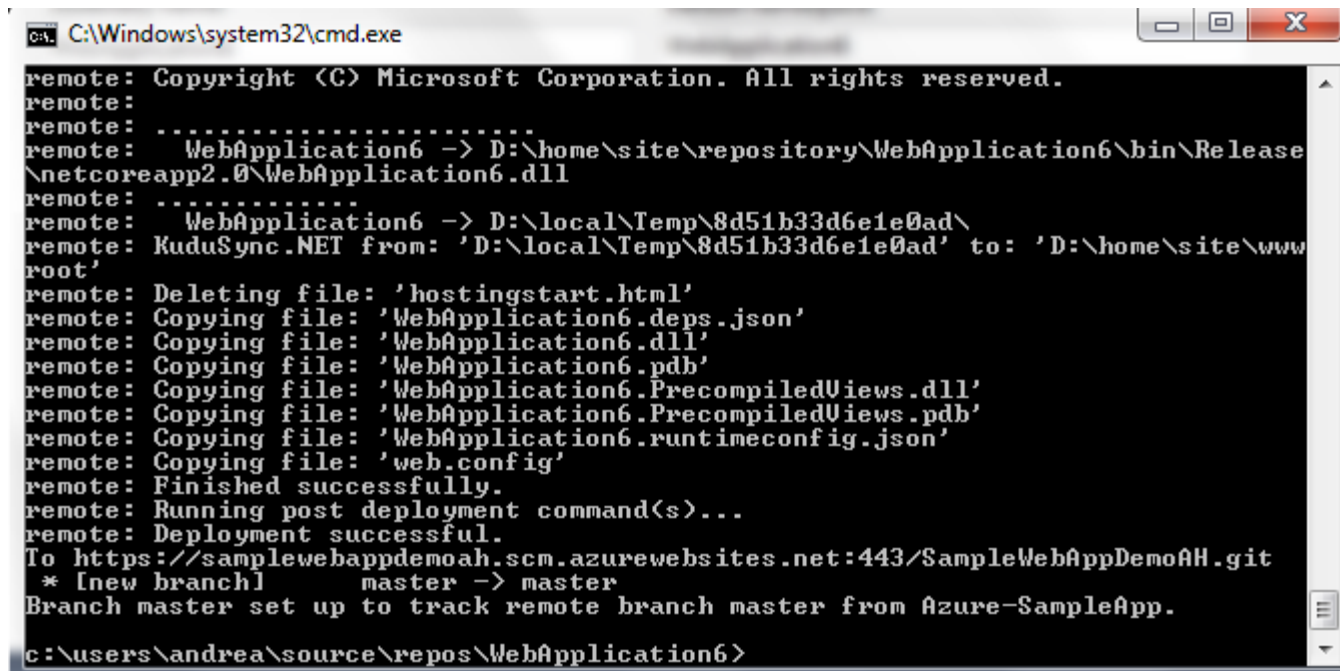
Update your web app in VS

- Open a command prompt to sync your changes with your Web App in Azure by going into Team Explorer and selecting the home icon -> Sync -> Actions -> Open Command Prompt



Update your web app in VS

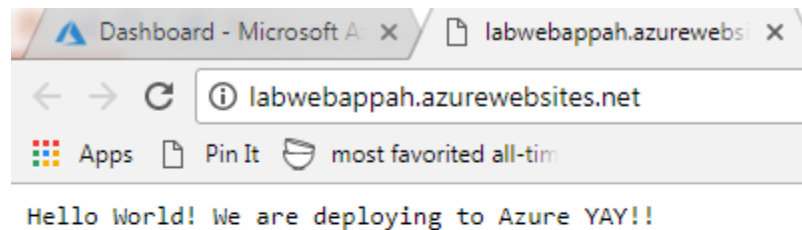
- Enter in: `git push -u (name of Remote) master`. Your app will start syncing and once done will say 'Deployment Successful'
 - Ex: `git push -u Azure-AppforLabAH master`



```
C:\Windows\system32\cmd.exe
remote: Copyright (C) Microsoft Corporation. All rights reserved.
remote:
remote: .....
remote: WebApplication6 -> D:\home\site\repository\WebApplication6\bin\Release
\netcoreapp2.0\WebApplication6.dll
remote: .....
remote: WebApplication6 -> D:\local\Temp\8d51b33d6e1e0ad\
remote: KuduSync.NET from: 'D:\local\Temp\8d51b33d6e1e0ad' to: 'D:\home\site\www
root'
remote: Deleting file: 'hostingstart.html'
remote: Copying file: 'WebApplication6.deps.json'
remote: Copying file: 'WebApplication6.dll'
remote: Copying file: 'WebApplication6.pdb'
remote: Copying file: 'WebApplication6.PrecompiledViews.dll'
remote: Copying file: 'WebApplication6.PrecompiledViews.pdb'
remote: Copying file: 'WebApplication6.runtimeconfig.json'
remote: Copying file: 'web.config'
remote: Finished successfully.
remote: Running post deployment command(s)...
remote: Deployment successful.
To https://samplewebappdemoah.scm.azurewebsites.net:443/SampleWebAppDemoAH.git
 * [new branch]      master -> master
Branch master set up to track remote branch master from Azure-SampleApp.
c:\users\andrea\source\repos\WebApplication6>
```

Update your web app in VS

- Finally, open a browser and enter the URL to your web app to view your update.



Update your web app in VS

- You can also view your webpage on your local host by selecting Debug -> Start without Debugging in VS

