

Lab 3: Creating ARM templates from scratch

[-> Lab 2: Work with an ARM template from an existing resource](#) | [Home](#) | [Lab 4: Failover between regions ->](#)

This lab will walk through some of the options for creating ARM templates.

Part 1: Azure QuickStart Templates

1. In a browser navigate to the [Azure QuickStart Templates](https://azure.microsoft.com/en-us/resources/templates/) (<https://azure.microsoft.com/en-us/resources/templates/>). This is a community resource of hundreds of examples of ARM Templates, from Microsoft and 3rd parties. If you have an infrastructure configuration in mind it's always worth searching to see if there are examples that would deliver, or at least give you a good start in creating, the ARM Templates required. We'll walk through finding an ARM template to match the manually created Windows Server with IIS Feature enabled in Lab 1.

The screenshot shows the Azure Quickstart Templates landing page. At the top, there's a blue header with the title "Azure Quickstart Templates" and a sub-instruction: "Deploy Azure resources through the Azure Resource Manager with community contributed templates to get more done. Deploy, learn, fork and contribute back." Below the header is a decorative graphic featuring clouds, a cube, and a person icon. The main content area has a light gray background. It starts with a section titled "What is Azure Resource Manager" which explains its purpose: "Azure Resource Manager allows you to provision your applications using a declarative template. In a single template, you can deploy multiple services along with their dependencies. You use the same template to repeatedly deploy your application during every stage of the application lifecycle." Below this is a "Learn more >" link. Further down, there's a search bar with a magnifying glass icon and a "See All" link. A message states "765 Quickstart templates are currently in the gallery." Underneath, there are two sections: "Most popular" and "See All".

2. In the Search field enter "IIS DSC", and select the "IIS Server using DSC extension on a Windows VM" option:

IIS DSC

x 

10 templates match your filter.

VM Scale Set Configuration managed by Azure Automation Deploy a VM Scale Set where virtual machines are deployed as registered nodes in the Azure Automation Desired State Configuration service, and node configura...  by Michael Greene, Last updated: 8/30/2018	Deploy Windows VM configure windows features SSL DSC This template allows you to deploy a Windows VM, configure windows features like IIS/Web Role, .Net, Custom login, windows auth, application initialization, do...  by Udy Sarma, Last updated: 8/22/2018	Create, configure and deploy Web Application to an Azure VM Create and configure a Windows VM with SQL Azure database, and deploy web application to the environment using PowerShell DSC  by Catherine Wang, Last updated: 8/15/2018
Simple DSC Pull Server This example allows to you deploy a powershell desired state configuration pull server.  by tcsatheesh[MSFT], Last updated: 8/1/2018	Create a DevTest environment with P2S VPN and IIS This template creates a simple DevTest environment with a Point-to-Site VPN and IIS on a Windows server which is a great way to get started.  by Adam Hurwitz, Last updated: 6/15/2018	IIS VMs & SQL Server 2014 VM. Create 1 or 2 IIS Windows 2012 R2 Web Servers and one back end SQL Server 2014 in VNET.  by Ali Baloch, Last updated: 5/22/2018
IIS Server using DSC extension on a Windows VM This template creates a Windows VM and sets up an IIS server using the DSC extension. NOTE: The DSC configuration module needs a SAS token to be passed in...  by Kay Singh, Last updated: 5/19/2018	App Gateway with WAF, SSL, IIS and HTTPS redirection This template deploys an Application Gateway with WAF, end to end SSL and HTTP to HTTPS redirect on the IIS servers.  by Jason Boeshart, Last updated: 4/28/2017	Deploy Windows VMSS configure windows features SSL DSC This template allows you to deploy two Windows VMSS, configure windows features like IIS/Web Role, .Net Framework 4.5, windows auth, application initialization, do...  by Udy Sarma, Last updated: 12/7/2016

3. Take a look at the information on the page, which includes details of the ARM template, the parameters to be supplied and examples of how to invoke the ARM template from the command line and powershell. Every ARM template in the QuickStart Gallery maintains the ARM template files in GitHub so that they are available to be reused. To take a look, once you've had a look at the information on the page, select Browse on GitHub:

[Templates](#) / IIS Server using DSC extension on a Windows VM

IIS Server using DSC extension on a Windows VM



by [Kay Singh](#)
Last updated: 5/19/2018

[Deploy to Azure](#)

[Browse on GitHub](#)

This template creates a Windows VM and sets up an IIS server using the DSC extension. NOTE: The DSC configuration module Storage. For DSC module link from Github (default in this template), this is not needed.

This Azure Resource Manager template was created by a member of the community and not by Microsoft. Each Resource Manager template is licensed to you under a for Resource Manager templates provided and licensed by community members and does not screen for security, compatibility, or performance. Community Resource program or service, and are made available AS IS without warranty of any kind.

Parameters

PARAMETER NAME	DESCRIPTION
diskType	Type of the Storage for disks
vmName	Name of the VM

4. You'll see a list of all the files in the GitHub repo. The most important are the .json files for the ARM Template itself, and the parameters file. Click on the `azuredeploy.json` file:

Azure / [azure-quickstart-templates](#)

Code Issues 547 Pull requests 39 Projects 0 Wiki Insights

Branch: master

[Create new file](#) [Find file](#) [History](#)

bmoore-msft and JulienFloris Changed Schema from http to https and 2014 preview to 2015 Latest commit e5cf7c1 on Nov 21, 2018

ContosoWebsite.ps1.zip Move DSC and Zookeeper over 4 years ago

README.md Add "Visualize" buttons to all template README.md files 3 years ago

azurereploy.json Hardcoding locations per request 10 months ago

azurereploy.parameters.json Changed Schema from http to https and 2014 preview to 2015 4 months ago

metadata.json added type property to metadata.json files 7 months ago

README.md

VM-DSC-Extension-IIS-Server

[Deploy to Azure](#) [Visualize](#)

This template allows you to create a VM with IIS Server and Management console setup. This is done using the DSC automation

5. You'll now see the contents of the ARM Template, most of which you should now be familiar with (parameters, resources etc.). Scroll down to the bottom of the template to see how DSC

is used to enable the IIS feature on the Windows VM. Note that the DSC script that is to be run is passed in as a parameter (modulesURL):

```

        ,
187    {
188        "type": "Microsoft.Compute/virtualMachines/extensions",
189        "name": "[concat(parameters('vmName'), '/', variables('vmExtensionName'))]",
190        "apiVersion": "2015-05-01-preview",
191        "location": "[parameters('location')]",
192        "dependsOn": [
193            "[concat('Microsoft.Compute/virtualMachines/', parameters('vmName'))]"
194        ],
195        "properties": {
196            "publisher": "Microsoft.PowerShell",
197            "type": "DSC",
198            "typeHandlerVersion": "2.19",
199            "autoUpgradeMinorVersion": true,
200            "settings": {
201                "ModulesUrl": "[parameters('modulesUrl')]", [
202                "ConfigurationFunction": "[parameters('configurationFunction')]", ]
203                "Properties": {
204                    "MachineName": "[parameters('vmName')]"
205                }
206            },
207            "protectedSettings": null
208        }
209    ]
210 }
211 }
```

6. To understand where the modulesURL is set and what it points to, go back in the browser to the previous contents page and select the azuredeploy.parameters.json file:

The screenshot shows a GitHub repository page for 'Azure / azure-quickstart-templates'. The repository has 589 stars and 7,086 forks. The 'Code' tab is selected, showing a list of files. The 'azurdeploy.parameters.json' file is highlighted with a red border. The file was last updated 4 months ago and contains the following commit message: 'Changed Schema from http to https and 2014 preview to 2015'.

File	Description	Last Commit
ContosoWebsite.ps1.zip	Move DSC and Zookeeper over	4 years ago
README.md	Add "Visualize" buttons to all template README.md files	3 years ago
azurdeploy.json	Hardcoding locations per request	10 months ago
azurdeploy.parameters.json	Changed Schema from http to https and 2014 preview to 2015	4 months ago
metadata.json	added type property to metadata.json files	7 months ago
README.md		

7. The parameters file sets all the default parameters to be passed into the main ARM template, unless they are overridden. Note that the default value for modulesURL is to a file in the same GitHub repository:

31 lines (30 sloc) | 869 Bytes

```

1  {
2      "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentParameters.json#",
3      "contentVersion": "1.0.0.0",
4      "parameters": {
5          "diskType": {
6              "value": "Standard_LRS"
7          },
8          "vmName": {
9              "value": "GEN-UNIQUE-8"
10         },
11         "vmSize": {
12             "value": "Standard_A2"
13         },
14         "imageSKU": {
15             "value": "2012-R2-Datacenter"
16         },
17         "adminUsername": {
18             "value": "GEN-UNIQUE"
19         },
20         "adminPassword": {
21             "value": "GEN-PASSWORD"
22         },
23         "modulesUrl": {
24             "value": "https://github.com/Azure/azure-quickstart-templates/raw/master/dsc-extension-iis-server-windows-vm/ContosoWebsite.ps1"
25         },
26         "configurationFunction": {
27             "value": "ContosoWebsite.ps1\\ContosoWebsite"
28         }
29     }
30 }

```

Copy the modulesURL value (<https://github.com/Azure/azure-quickstart-templates/raw/master/dsc-extension-iis-server-windows-vm/ContosoWebsite.ps1.zip>) to the clipboard or notepad for use in a minute.

8. Go back to the previous contents page again and the file that is being referenced in the parameter is the .ps1.zip file:

A screenshot of a GitHub repository page for 'Azure / azure-quickstart-templates'. The repository has 589 issues, 4,864 stars, and 7,086 forks. The 'Code' tab is selected. A dropdown menu shows 'Branch: master'. Below the navigation bar, it says 'azure-quickstart-templates / dsc-extension-iis-server-windows-vm /'. A list of files is shown, with 'ContosoWebsite.ps1.zip' highlighted by a red box. Other files listed include README.md, azuredeploy.json, azuredeploy.parameters.json, and metadata.json. The 'ContosoWebsite.ps1.zip' file was last modified 4 years ago.

File	Description	Last Modified
ContosoWebsite.ps1.zip	Move DSC and Zookeeper over	4 years ago
README.md	Add "Visualize" buttons to all template README.md files	3 years ago
azuredeploy.json	Hardcoding locations per request	10 months ago
azuredeploy.parameters.json	Changed Schema from http to https and 2014 preview to 2015	4 months ago
metadata.json	added type property to metadata.json files	7 months ago
README.md		

9. You could fork or download the files and then change them or reuse them as you see fit but let's return to the Azure QuickStart page (back in the browser or if needed search for the template again as above). Click on the Deploy to Azure button to deploy directly into your subscription:

[Templates](#) / IIS Server using DSC extension on a Windows VM

IIS Server using DSC extension on a Windows VM



by [Kay Singh](#)

Last updated: 5/19/2018

[Deploy to Azure](#)

[Browse on GitHub](#)

This template creates a Windows VM and sets up an IIS server using the DSC extension. NOTE: The DSC configuration module link from Github (default in this template), this is not needed.

This Azure Resource Manager template was created by a member of the community and not by Microsoft. Each Resource Manager template is licensed to you under the terms of the MIT license. This template is provided as-is without warranty or support. Microsoft and its partners are not responsible for the functionality of this template or the data it manages. This template is not supported by Microsoft.

Parameters

-
10. You will be taken into the Azure Portal and onto the template deployment page. Check all the fields but create a new resource group (again for easy deletion), vm name, admin user, password and copy in the modulesURL from above. Then agree to the terms and conditions and click Purchase:

[Home](#) > IIS Server using DSC extension on a Windows VM

IIS Server using DSC extension on a Windows VM

Azure quickstart template

* Subscription

Azure On Line Services

* Resource group

(New) Lab3P1-RG

[Create new](#)

* Location

UK South

SETTINGS

Disk Type ⓘ

Standard_LRS

* Vm Name ⓘ

lab3iisbox

Vm Size ⓘ

Standard_A2

Image SKU ⓘ

2012-R2-Datacenter

* Admin Username ⓘ

vmadmin

* Admin Password ⓘ

.....

* Modules Url ⓘ

t-templates/raw/master/dsc-extension-iis-server-windows-vm/ContosoWebsite.ps ...

Configuration Function ⓘ

ContosoWebsite.ps1\ContosoWebsite

Location ⓘ

[resourceGroup().location]

TERMS AND CONDITIONS

[Template information](#) | [Azure Marketplace Terms](#) | [Azure Marketplace](#)

By clicking "Purchase," I (a) agree to the applicable legal terms associated with the offering; (b) authorize Microsoft to charge or bill my current payment method for the fees associated the offering(s), including applicable taxes, with the same billing frequency as my Azure subscription, until I discontinue use of the offering(s); and (c) agree that, if the deployment involves 3rd party offerings, Microsoft may share my contact information and other details of such deployment with the publisher of that offering.

 I agree to the terms and conditions stated above[Purchase](#)

11. You should see the deployment gets underway:

 Delete  Cancel  Redeploy  Refresh

... Your deployment is underway

Check the status of your deployment, manage resources, or troubleshoot deployment issues. Pin this page to your dashboard to easily find it next time.



Deployment name: Microsoft.Template

Subscription: [Azure On Line Services](#)

Resource group: [Lab3P1-RG](#)

DEPLOYMENT DETAILS [\(Download\)](#)

Start time: 27/03/2019, 10:06:03

Duration: 1 minute 16 seconds

Correlation ID: 08c0d69a-4b28-425e-8172-a93016c3b68d

RESOURCE	TYPE	STATUS	OPERATION DETAILS
 lab3iisbox	Microsoft.Compute/...	Created	Operation details
 dscNIC	Microsoft.Network/n...	Created	Operation details
 dscVNET	Microsoft.Network/v...	OK	Operation details
 dscPubIP	Microsoft.Network/p...	OK	Operation details

12. Find the VM in the portal and once it shows as running select the extensions blade to see that DSC has been configured and is in the process of updating the VM:

lab3iisbox - Extensions

Virtual machine

Search (Ctrl+ /)

Add

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

- Networking
- Disk
- Size
- Security
- Extensions**
- Continuous delivery (Preview)
- Availability set
- Configuration
- Identity
- Properties
- Locks
- Export template

NAME	TYPE	VERSION	STATUS
dscExtension	Microsoft.PowerShell.DSC	2.*	Transitioning

13. Select the status details and, depending on timing, you'll see it's status change from transitioning to succeeded:

Home > Virtual machines > lab3iisbox - Extensions > dscExtension > dscExtension

dscExtension

lab3iisbox

Uninstall

Type: Microsoft.PowerShell.DSC

Version: 2.77.0.0

Status: Provisioning succeeded

Status level: Info

Status message: DSC configuration was applied successfully.

Detailed status: View detailed status

Handler status: Ready

Handler status level: Info

dscExtension

lab3iisbox

```

1  [
2   {
3     "code": "ComponentStatus/DscConfigurationLog/succeeded",
4     "level": "Info",
5     "displayStatus": "Provisioning succeeded",
6     "message": "[2019-03-27 10:19:54Z] [VERBOSE] [lab3iisbox]"
7   },
8   {
9     "code": "ComponentStatus/DscExtensionLog/succeeded",
10    "level": "Info",
11    "displayStatus": "Provisioning succeeded",
12    "message": "[2019-03-27 10:16:50Z] VMUUID is 7EA736FA-3DC0-4534-994F-0BF621138490;A"
13  },
14  {
15    "code": "ComponentStatus/Metadata/succeeded",
16    "level": "Info",
17    "displayStatus": "Provisioning succeeded",
18    "message": "VMUUID=7EA736FA-3DC0-4534-994F-0BF621138490;A"
19  }
20 ]

```

14. If you want to confirm, connect to the VM and you'll see that the IIS feature is enabled and running and/or open the browser and connect to <http://localhost> to see the default IIS page running.

The screenshot shows the Server Manager Dashboard. On the left, there's a navigation bar with 'Server Services' and a dropdown arrow. The main area has a 'WELCOME TO SERVER MANAGER' header. A vertical sidebar on the left contains 'QUICK START', 'WHAT'S NEW', and 'LEARN MORE' buttons. The 'QUICK START' button is highlighted in orange. Below this, under 'ROLES AND SERVER GROUPS', it says 'Roles: 2 | Server groups: 1 | Servers total: 1'. There are four items listed: 'File and Storage Services' (1 instance), 'IIS' (1 instance, highlighted with a red box), 'Local Server' (1 instance), and 'All Server Groups' (1 instance). Each item has a green horizontal bar above it and a list of metrics below: Manageability, Events, Performance, and BPA results.

This VM won't be needed any longer so delete the resource group whenever you've finished to ensure minimum cost.

Part 2: Creating an ARM template in the Portal

1. In the Azure Portal select Create a resource and search for Template Deployment:

The screenshot shows the Azure portal's 'New' blade. On the left is a sidebar with various service icons and names. The main area has a search bar at the top containing the text 'template'. Below the search bar is a list of results. The first result, 'Template deployment', is highlighted with a red box. Other results include 'PrestaShop Advanced Template', 'PrestaShop Performance Template', 'EvoStream Media Server for Template', and 'JFrog Artifactory Enterprise ARM Template'. Below the search results, there are categories and their corresponding icons and links:

Category	Icon	Description
Storage	[Icon]	Web App Quickstart tutorial
Web	[Icon]	SQL Database Quickstart tutorial
Mobile	[Icon]	Function App Quickstart tutorial
Containers	[Icon]	Azure Cosmos DB Quickstart tutorial
Databases	[Icon]	Mixed Reality
Analytics	[Icon]	Integration
AI + Machine Learning	[Icon]	Kubernetes Service Quickstart tutorial
Internet of Things	[Icon]	
Mixed Reality	[Icon]	
Integration	[Icon]	

2. Select Create on the next page:

[Home](#) > [New](#) > [Template deployment](#)

Template deployment

Microsoft

Applications running in Microsoft Azure usually rely on a combination of resources, like databases, servers, and web apps. Azure Resource Manager templates enable you to deploy and manage these resources as a group, using a JSON description of the resources and their deployment settings.

Edit your template with IntelliSense and deploy it to a new or existing resource group.

 Save for later

Publisher Microsoft

Useful Links [Documentation](#)

Select a software plan

Template deployment

Customize your template and build for the cloud



Create

3. You will see that there are many options including some common templates and all the Azure QuickStart templates from Part 1 (including the IIS DSC example you used, shown selected in the screenshot). Select build your own template in the editor:

[Home](#) > [New](#) > [Template deployment](#) > [Custom deployment](#)

Custom deployment

Deploy from a custom template

Learn about template deployment

 [Read the docs](#) 

 [Build your own template in the editor](#)

Common templates

 [Create a Linux virtual machine](#)

 [Create a Windows virtual machine](#)

 [Create a web app](#)

 [Create a SQL database](#)

Load a GitHub quickstart template

Select a template (disclaimer) 

dsc-extension-iis-server-windows-vm



This template creates a VM with IIS server using DSC extension

Author: singhkays

Last updated: 2018-05-19

[Learn more](#)

[Select template](#)

[Edit template](#)

4. You'll then have a blank ARM template. Note that you can add resources (e.g. a virtual machine, a virtual network etc.) piece by piece or choose again from the QuickStart Templates. Select the QuickStart Template:

Home > New > Template deployment > Custom deployment > Edit template

Edit template

Edit your Azure Resource Manager template

+ Add resource ↑ Quickstart template ⌂ Load file ⌂ Download

Parameters (0)
Variables (0)
Resources (0)

```
1 {  
2   "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",  
3   "contentVersion": "1.0.0.0",  
4   "parameters": {},  
5   "resources": []  
6 }
```

5. Search for the template used earlier by typing DSC into the field and selecting dsc-extension-iis-server-windows-vm:

Home > New > Template deployment > Custom deployment > Edit template

Edit template

Edit your Azure Resource Manager template

+ Add resource ↑ Quickstart template ⌂ Load file ⌂ Download

Load a quickstart template

Select a template (disclaimer) ⓘ

dsc

- 201-dsc-linux-azure-storage-on-ubuntu
- 201-dsc-linux-public-storage-on-ubuntu
- 201-vmss-automation-dsc
- 201-vmss-windows-webapp-dsc-autoscale
- 201-web-app-vm-dsc
- dsc-extension-azure-automation-pullserver
- dsc-extension-iis-server-windows-vm
- dsc-pullserver-to-win-server

6. Confirm by clicking OK:

Home > New > Template deployment > Custom deployment > Edit template

Edit template

Edit your Azure Resource Manager template

Add resource **Quickstart template** **Load file** **Download**

Load a quickstart template

Select a template (disclaimer) ⓘ

dsc-extension-iis-server-windows-vm

This template creates a VM with IIS server using DSC extension

Author: singhkays
Last updated: 2018-05-19
[Learn more](#)

OK **Cancel**

7. The blank ARM Template has now been replaced with the one you used in the previous part above:

Home > New > Template deployment > Custom deployment > Edit template

Edit template

Edit your Azure Resource Manager template

Add resource Quickstart template Load file Download

Parameters (9) Variables (12) Resources (5)

```
18
19     "description": "Name of the VM"
20   },
21 }
22 "vmSize": {
23   "type": "string",
24   "defaultValue": "Standard_DS1_v2",
25   "metadata": {
26     "description": "Size of the VM"
27   }
28 },
29 "imageSKU": {
30   "type": "string",
31   "defaultValue": "2012-R2-Datacenter",
32   "allowedValues": [
33     "2008-R2-SP1",
34     "2012-Datacenter",
35     "2012-R2-Datacenter",
36     "2016-Datacenter"
37   ],
38   "metadata": {
39     "description": "Image SKU"
40   }
41 },
42 "adminUsername": {
43   "type": "string",
44   "metadata": {
45     "description": "Admin username"
46   }
}
```

8. Edit the template to change the default VM size to "Standard_DS1_v2" and add an additional allowed value for the image SKU of "2016-Datacenter". Note that, again, the preceding allowed value needs a comma added:

Home > New > Template deployment > Custom deployment > Edit template

Edit template

Edit your Azure Resource Manager template

Add resource Quickstart template Load file Download

Parameters (9) Variables (12) Resources (5)

```
18
19     "description": "Name of the VM"
20   },
21 },
22 "vmSize": {
23   "type": "string",
24   "defaultValue": "Standard_DS1_v2",
25   "metadata": {
26     "description": "Size of the VM"
27   }
28 },
29 "imageSKU": {
30   "type": "string",
31   "defaultValue": "2012-R2-Datacenter",
32   "allowedValues": [
33     "2008-R2-SP1",
34     "2012-Datacenter",
35     "2012-R2-Datacenter",
36     "2016-Datacenter"
37   ],
38   "metadata": {
39     "description": "Image SKU"
40   }
41 },
42 "adminUsername": {
43   "type": "string",
44   "metadata": {
45     "description": "Admin username"
46   }
}
```

9. Click Save and then you will be presented with the deployment screen. Note that your changes are now showing:

Home > New > Template deployment > Custom deployment

Custom deployment

Deploy from a custom template

* Subscription	Azure On Line Services
* Resource group	Select a resource group Create new
* Location	UK South

SETTINGS

Disk Type ⓘ	Standard_LRS
* Vm Name ⓘ	
Vm Size ⓘ	Standard_DS1_v2
Image SKU ⓘ	2012-R2-Datacenter
* Admin Username ⓘ	2008-R2-SP1
* Admin Password ⓘ	2012-Datacenter
* Modules Url ⓘ	2012-R2-Datacenter
Configuration Function ⓘ	ContosoWebsite.ps1\ContosoWebsite
Location ⓘ	[resourceGroup().location]

TERMS AND CONDITIONS

[Purchase](#)

10. If you want to then test out your changes by completing the fields and deploying a new VM. Again, the recommendation is to create a new resource group and then to delete the resource group when you are finished with it. If you select the deployment notification and click on Deployment:

Deployment in progress... Running 

Deployment to resource group 'Lab3P2-RG' is in progress.

3 minutes ago

you can also select the template and add it into the library for reuse:

```

1  {
2      "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
3      "contentVersion": "1.0.0.0",
4      "parameters": {
5          "diskType": {
6              "defaultValue": "Standard_LRS",
7              "allowedValues": [
8                  "Standard_LRS",
9                  "Premium_LRS"
10             ],
11             "type": "String",
12             "metadata": {
13                 "description": "Type of the Storage for disks"
14             }
15         },
16         "vmName": {
17             "type": "String",
18             "metadata": {
19                 "description": "Name of the VM"
20             }
21         },
22         "vmSize": {
23     }
24     }
25 }

```

Part 3: Creating an ARM Template in Visual Studio

1. The Portal editor is good, but it can be preferable to work offline and/or with more editing options and other capabilities, such as adding ARM Templates to source control. Visual Studio provides good ARM Template editing support. Open Visual Studio and select File | New | Project and select Cloud | Azure Resource Group. Save the solution name and location as you wish and then click OK:

New Project

Recent

Installed

Visual C#

- Get Started
- Windows Universal
- Windows Desktop
- Web

 - Previous Versions
 - .NET Core
 - .NET Standard
 - Cloud**
 - Net Standard
 - Test
 - WCF

Azure Data Lake

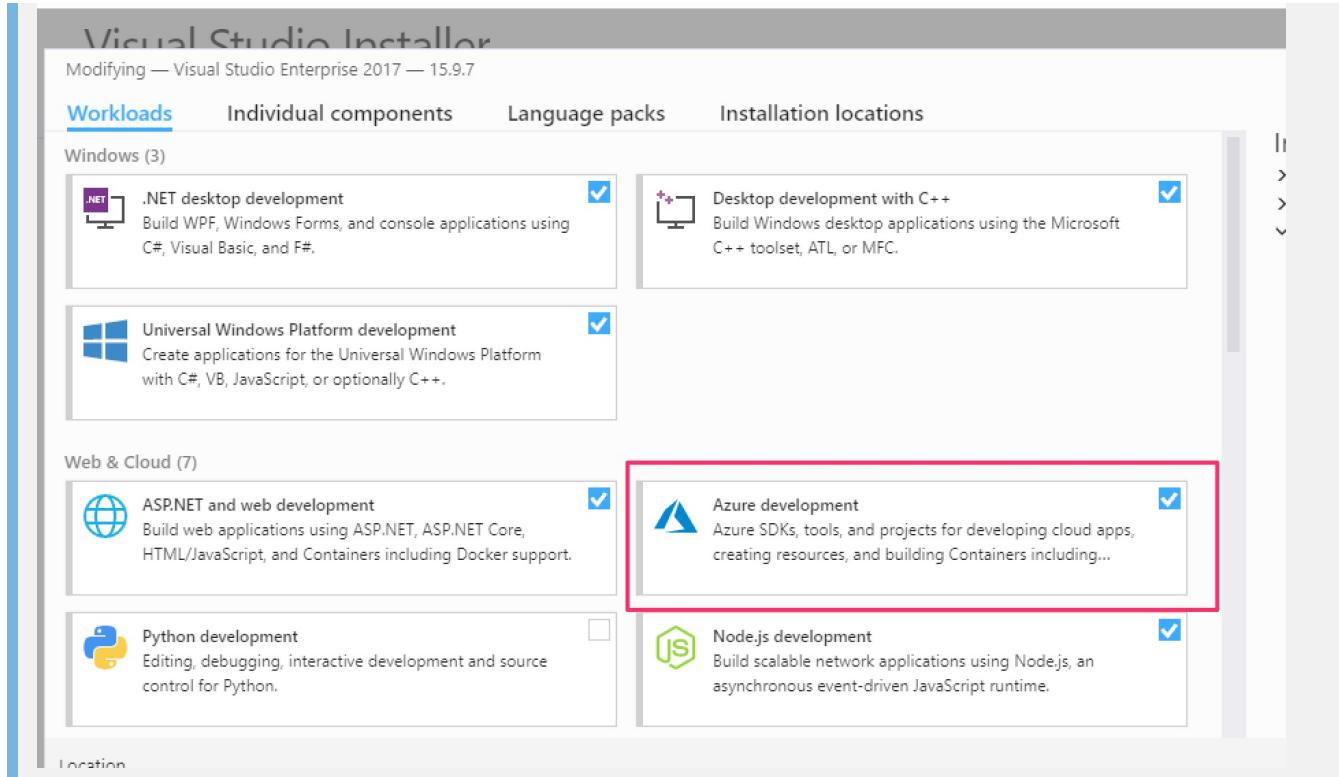
Not finding what you are looking for?

Open Visual Studio Installer

Name:	AzureResourceGroup5	
Location:	C:\Users\giles\Documents\Visual Studio 2017\Projects	Browse...
Solution name:	AzureResourceGroup5	<input checked="" type="checkbox"/> Create directory for solution
Framework:	.NET Framework 4.6	<input type="checkbox"/> Create new Git repository

OK Cancel

If you don't see the Cloud option, then you need to make sure that Azure Development is enabled in Visual Studio. Open the Visual Studio Installer, select modify and check the Azure Development section:



2. You will then be presented with a dialog offering ARM templates to choose from. Note that you can select from some basic templates or from the Azure QuickStart gallery templates you used earlier:

Select Azure Template X

Show templates from this location: Visual Studio Templates

Blank Template
By: Microsoft
This template will add a deployment project file and other files needed to Azure Resource Manager.

Visual Studio Templates

- Azure QuickStart - Featured (github.com/Azure/azure-quickstart-templates)
- Azure QuickStart (github.com/Azure/azure-quickstart-templates)
- Azure Stack QuickStart (github.com/AzureStack-QuickStart-Templates)

Windows Virtual Machine
Scale Set
MICROSOFT

Linux Virtual Machine Scale Set
MICROSOFT

Service Fabric Cluster
MICROSOFT

Web app
MICROSOFT

Web app + SQL
MICROSOFT

Templates Found: 14

OK Cancel

3. Select the Visual Studio Templates and then choose Windows Virtual Machine. Click OK:

Select Azure Template X

Show templates from this location: Visual Studio Templates

Windows Virtual Machine
By: Microsoft
This template takes a minimum amount of parameters and deploys a Windows VM, using the latest patched version.

VERSION: 2015-04-28

Windows Server Virtual Machines with Load Balancer
MICROSOFT

This template allows you to deploy a simple Windows VM using a few different options for the Windows version, using the latest patched version. This will deploy in West US on a D1 VM Size.

Docker on Ubuntu Server
CANONICAL + MS OPEN TECH

Windows Virtual Machine
MICROSOFT

Ubuntu Server
MICROSOFT

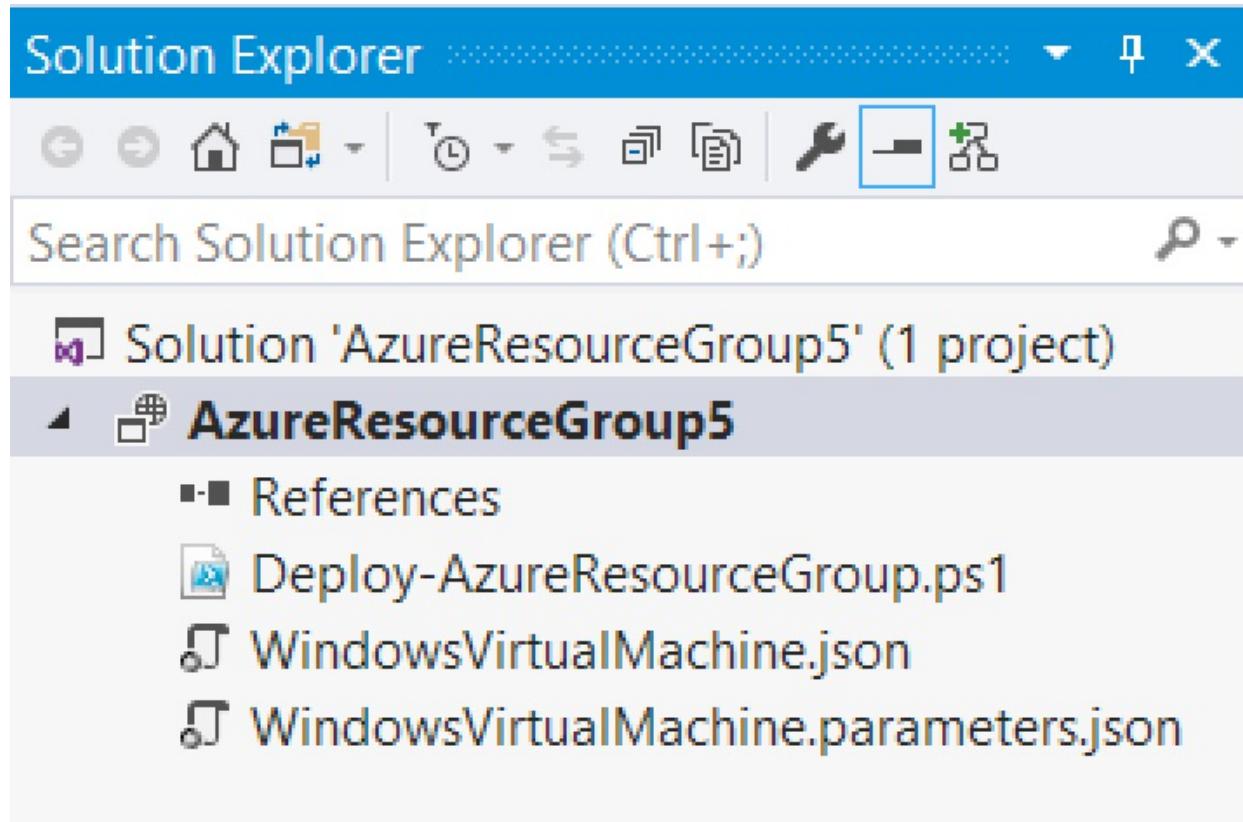
Web app + CDN
MICROSOFT

Templates Found: 14

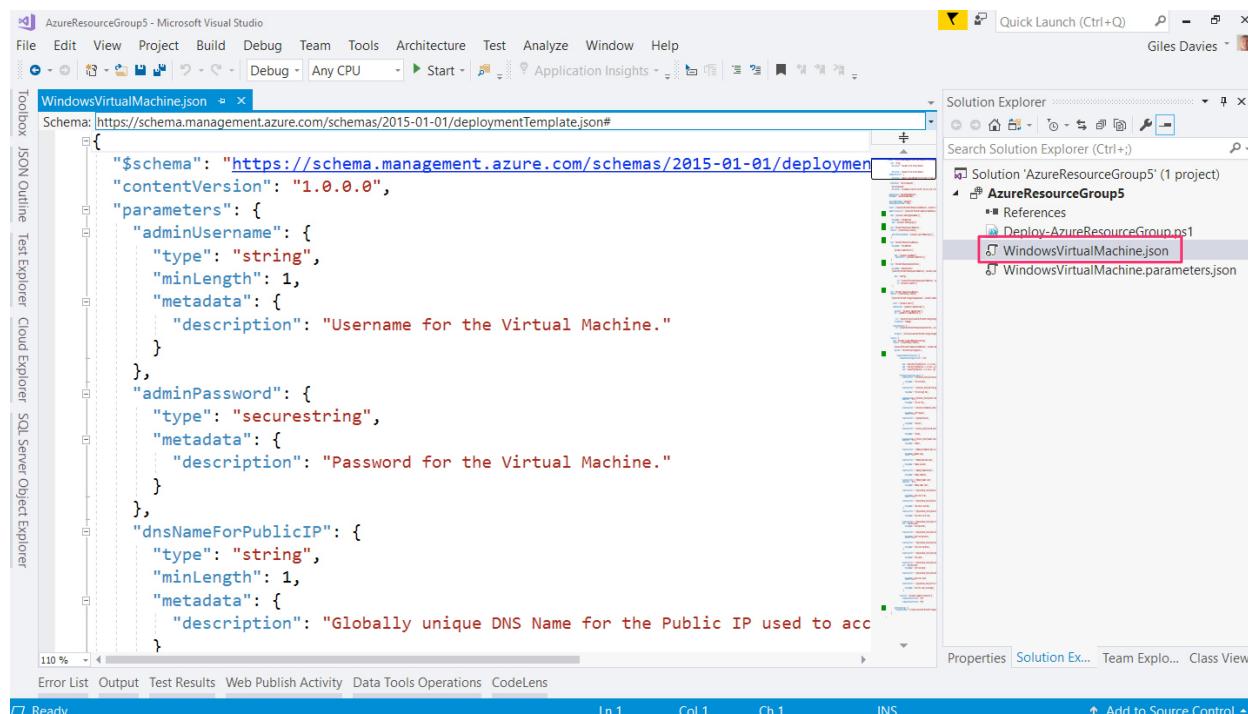
OK Cancel

4. You will now have a new solution with 3 files in it:

- Deploy-AzureResourceGroup.ps1: A powershell script to deploy the ARM Template. If you are not going to deploy via Powershell then you can ignore or delete this file.
- WindowsVirtualMachine.json: The ARM Template itself.
- WindowsVirtualMachine.parameters.json: The ARM Template parameters file.



5. Open the WindowsVirtualMachine.json file to see the JSON file in the editor:



6. Find the WindowsOSVersion parameter and add an additional line to allowedvalues e.g. 2016-Datacenter but DON'T add the preceding comma in order to see how the editor will

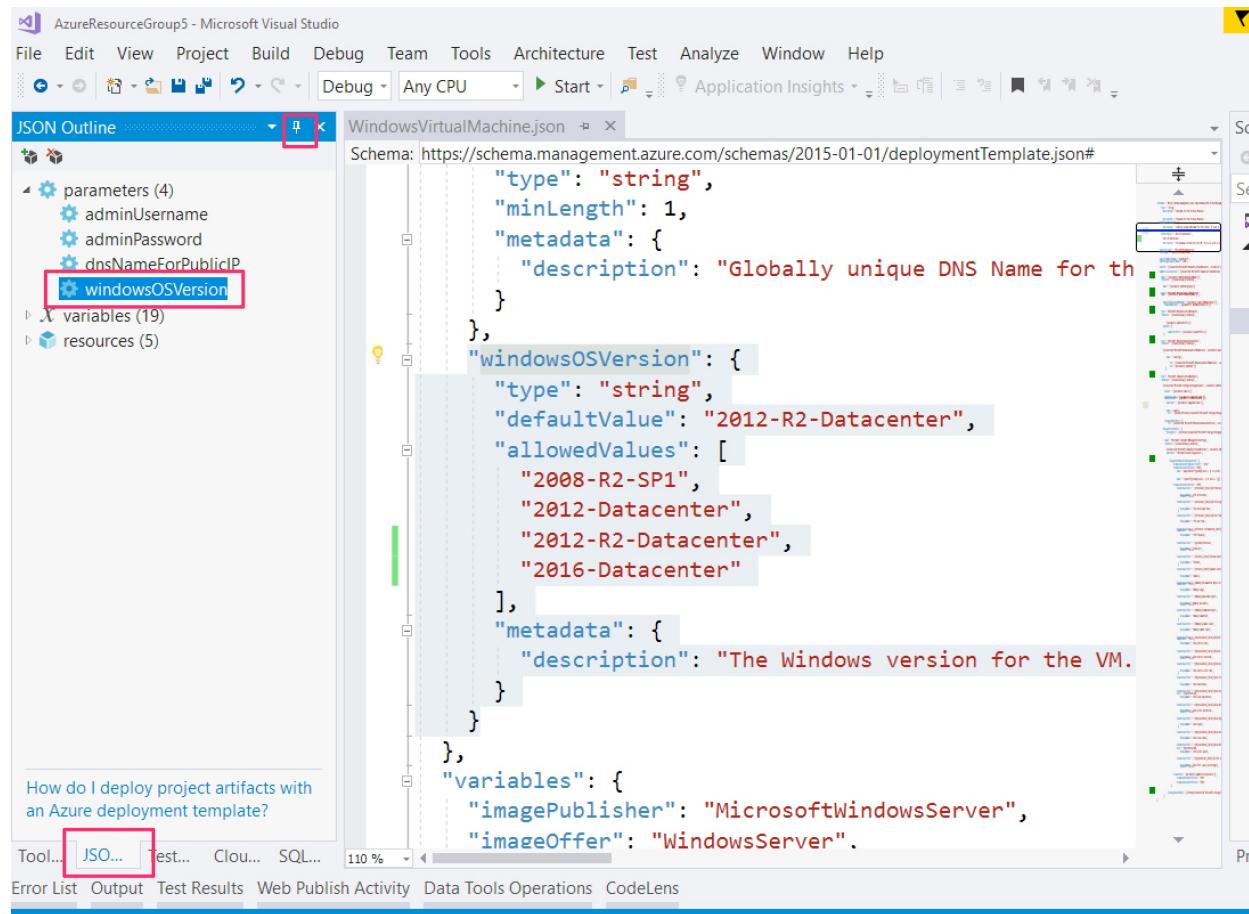
help you. Notice that Visual Studio displays a green squiggly line to indicate an issue. If you hover over it, it will say that a comma is missing:

```
    },
    "windowsOSVersion": {
        "type": "string",
        "defaultValue": "2012-R2-Datacenter",
        "allowedValues": [
            "2008-R2-SP1",
            "2012-Datacenter",
            "2012-R2-Datacenter"
            "2016-Datacenter" ⚠ Missing a comma after an array element.
        ],
        "metadata": {
            "description": "The Windows version for the VM. Tr"
        }
    }
```

7. Add in the missing comma and save the file:

```
    },
    "windowsOSVersion": {
        "type": "string",
        "defaultValue": "2012-R2-Datacenter",
        "allowedValues": [
            "2008-R2-SP1",
            "2012-Datacenter",
            "2012-R2-Datacenter",
            "2016-Datacenter"
        ],
        "metadata": {
            "description": "The Windows version for the"
        }
    }
```

8. Open the JSON Outline view (usually on the left hand side of Visual Studio). You may want to pin it open for convenience. This allows you to see the elements in the ARM Template and if you click on an element (e.g. a parameter) then it selects that element in the ARM Template. This can make understanding and navigating around the ARM Template much easier:



The screenshot shows the Microsoft Visual Studio interface with the 'WindowsVirtualMachine.json' file open in the JSON Outline view. The 'parameters' section is expanded, and the 'windowsOSVersion' parameter is selected. The 'variables' section is also visible. A red box highlights the 'JSO...' tab in the toolbar.

```
parameters (4)
  adminUsername
  adminPassword
  dnsNameForPublicIP
  windowsOSVersion
variables (19)
resources (5)
```

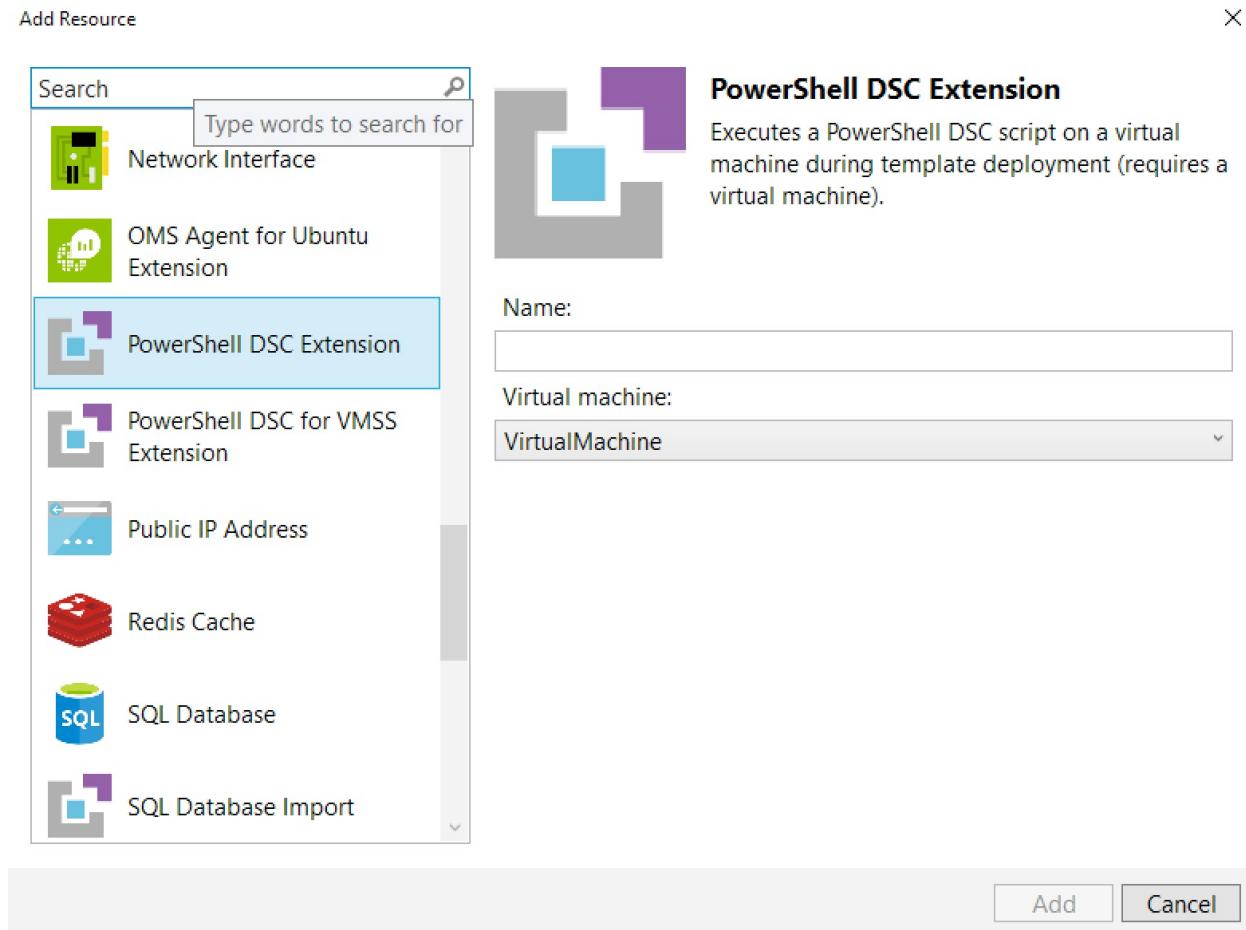
```
Schema: https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#
{
  "type": "string",
  "minLength": 1,
  "metadata": {
    "description": "Globally unique DNS Name for the VM"
  },
  "windowsOSVersion": {
    "type": "string",
    "defaultValue": "2012-R2-Datacenter",
    "allowedValues": [
      "2008-R2-SP1",
      "2012-Datacenter",
      "2012-R2-Datacenter",
      "2016-Datacenter"
    ],
    "metadata": {
      "description": "The Windows version for the VM."
    }
  },
  "variables": {
    "imagePublisher": "MicrosoftWindowsServer",
    "imageOffer": "WindowsServer"
  }
}
```

Try selecting different parameters, variables and resources.

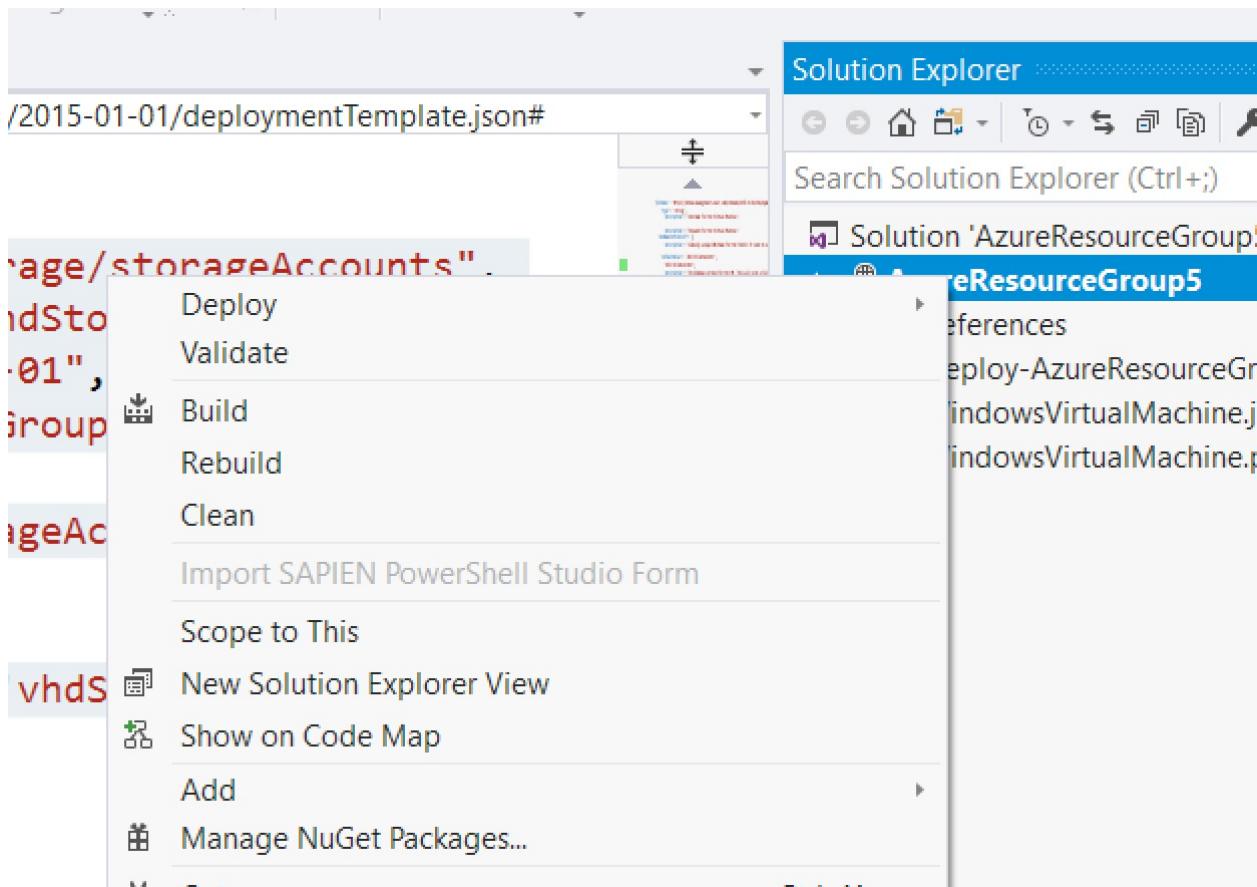
9. You can add in additional elements in the JSON outline view. Right-click on the Resources section and select Add new resource:

The screenshot shows two tabs side-by-side. The left tab is titled "JSON Outline" and displays a hierarchical structure of parameters, variables, and resources. Under "resources (5)", there is a context menu with options: "Add New Resource" (highlighted with a red box), "Delete", "VirtualNetwork", "NetworkInterface", and "VirtualMachine". The right tab is titled "WindowsVirtualMachine" and shows a partial JSON schema with placeholder values like "resourceType": "Microsoft.Compute/virtualMachines", "name": "vmName", etc.

10. Take a look at the options in the resource list. To replicate what we did earlier for example you could add in a DSC extension. For this lab there is no need to actually add them in, just be aware:



11. To deploy an ARM Template from Azure right-click on the solution and select Deploy | New:



12. Then complete the fields in the dialog, including creating a new resource group (if desired):

Deploy to Resource Group X

 Microsoft
gidavies@microsoft.com

Subscription:
Azure On Line Services (Giles Davies)

Resource group:
Lab3P3-RG (UK South)

Deployment template:
windowsvirtualmachine.json

Template parameters file:
windowsvirtualmachine.parameters.json Edit Parameters...

Artifact storage account: i

How do I deploy project artifacts with an Azure deployment template

Deploy Cancel

Feel free to deploy (and later delete) anything you've created but for this lab it's not necessary, understanding that you can deploy directly from Visual Studio is the aim.

[<- Lab 2: Work with an ARM template from an existing resource](#) | [Home](#) | [Lab 4: Failover between regions ->](#)