Project

Automating Workloads with ARM Templates

DESCRIPTION

Business Scenario

The Rand Enterprises Corporation wants to test ARM template to bring infrastructure as code into practice. They have decided to work on project RandEnt to verify the functionality.

The operations team at Rand decides to define entire networking architecture using ARM template, once that's in place they intended to create the storage account along with virtual machine housing their application.

As Rand Enterprises works extensively on delivering Image based content for their global audience, they are seeking to improve the performance on that aspect. To facilitate the same, they have decided to introduce Azure CDN of Standard_Akamai SKU.

The expectation of the operation team is to Rather than deploying resources in Azure independently, they should leverage Azure ARM templates to deploy and provision all resources in templatize format.

Overview

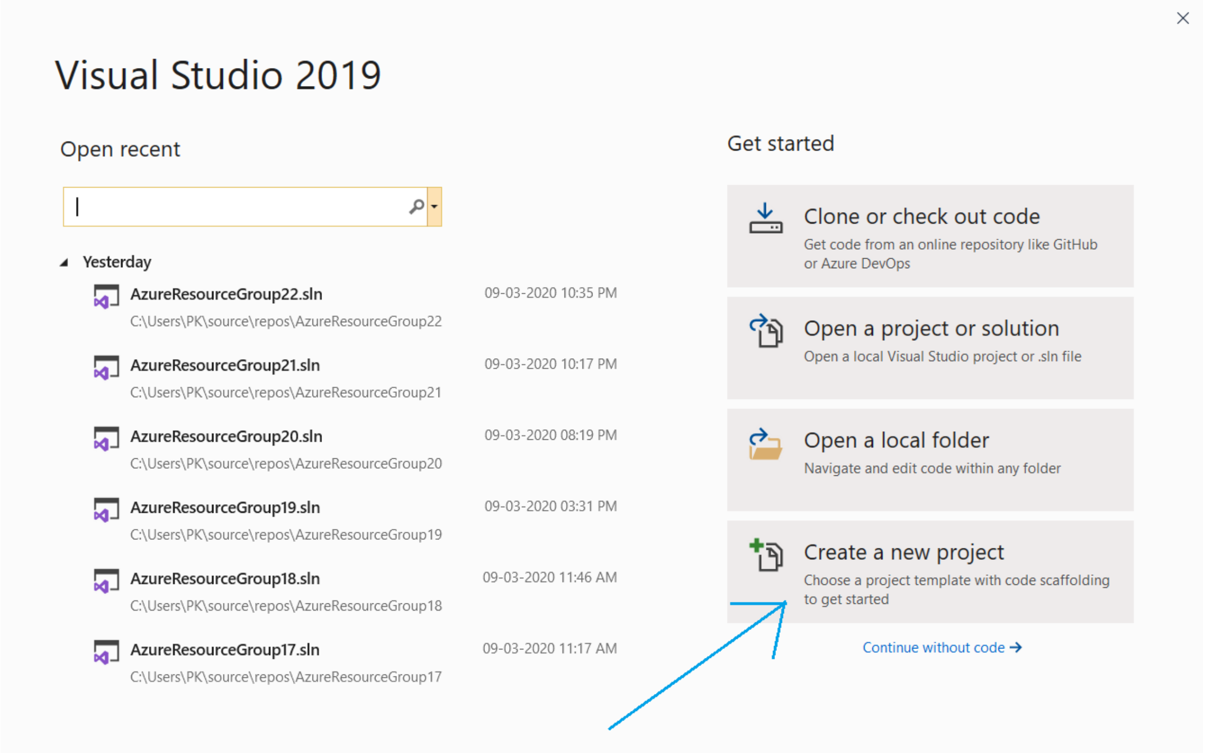
The main tasks for this exercise are as follows:

1. Define the network.
2. Extend that with Compute & Storage.
3. Create the Storage account for Images & implement CDN.

Note: Here, I am using Visual studio 2019 software to create ARM templates and to deploy it on Azure portal.

Task 1: Define network.

Step 1: Open visual studio 2019 and click on **“Create a new project”** as shown in the window below.



Visual Studio 2019

Open recent

Yesterday

AzureResourceGroup22.sln C:\Users\PK\source\repos\AzureResourceGroup22	09-03-2020 10:35 PM
AzureResourceGroup21.sln C:\Users\PK\source\repos\AzureResourceGroup21	09-03-2020 10:17 PM
AzureResourceGroup20.sln C:\Users\PK\source\repos\AzureResourceGroup20	09-03-2020 08:19 PM
AzureResourceGroup19.sln C:\Users\PK\source\repos\AzureResourceGroup19	09-03-2020 03:31 PM
AzureResourceGroup18.sln C:\Users\PK\source\repos\AzureResourceGroup18	09-03-2020 11:46 AM
AzureResourceGroup17.sln C:\Users\PK\source\repos\AzureResourceGroup17	09-03-2020 11:17 AM

Get started

Clone or check out code
Get code from an online repository like GitHub or Azure DevOps

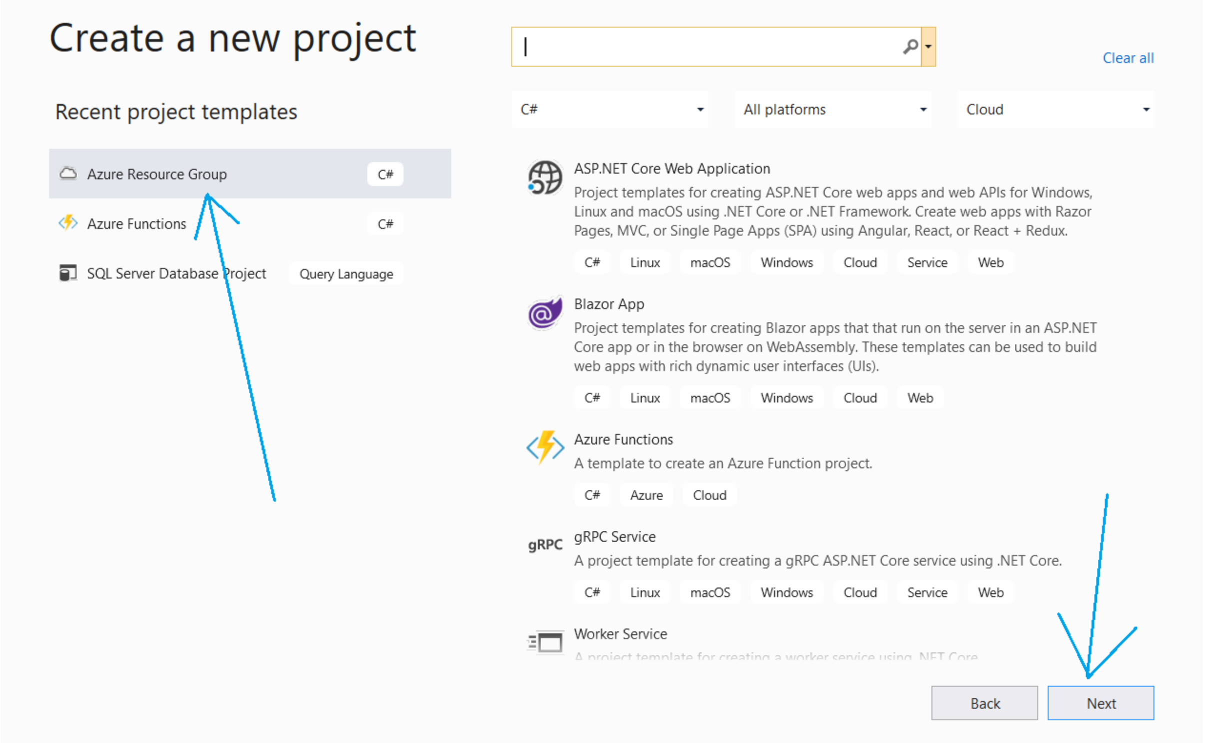
Open a project or solution
Open a local Visual Studio project or .sln file

Open a local folder
Navigate and edit code within any folder

Create a new project
Choose a project template with code scaffolding to get started

Continue without code →

Next, select **“Azure Resource Group”** and click on **“Next”** as shown in window below.



Create a new project

Recent project templates

Azure Resource Group	C#
Azure Functions	C#
SQL Server Database Project	Query Language

Clear all

C# | All platforms | Cloud

ASP.NET Core Web Application
Project templates for creating ASP.NET Core web apps and web APIs for Windows, Linux and macOS using .NET Core or .NET Framework. Create web apps with Razor Pages, MVC, or Single Page Apps (SPA) using Angular, React, or React + Redux.
C# Linux macOS Windows Cloud Service Web

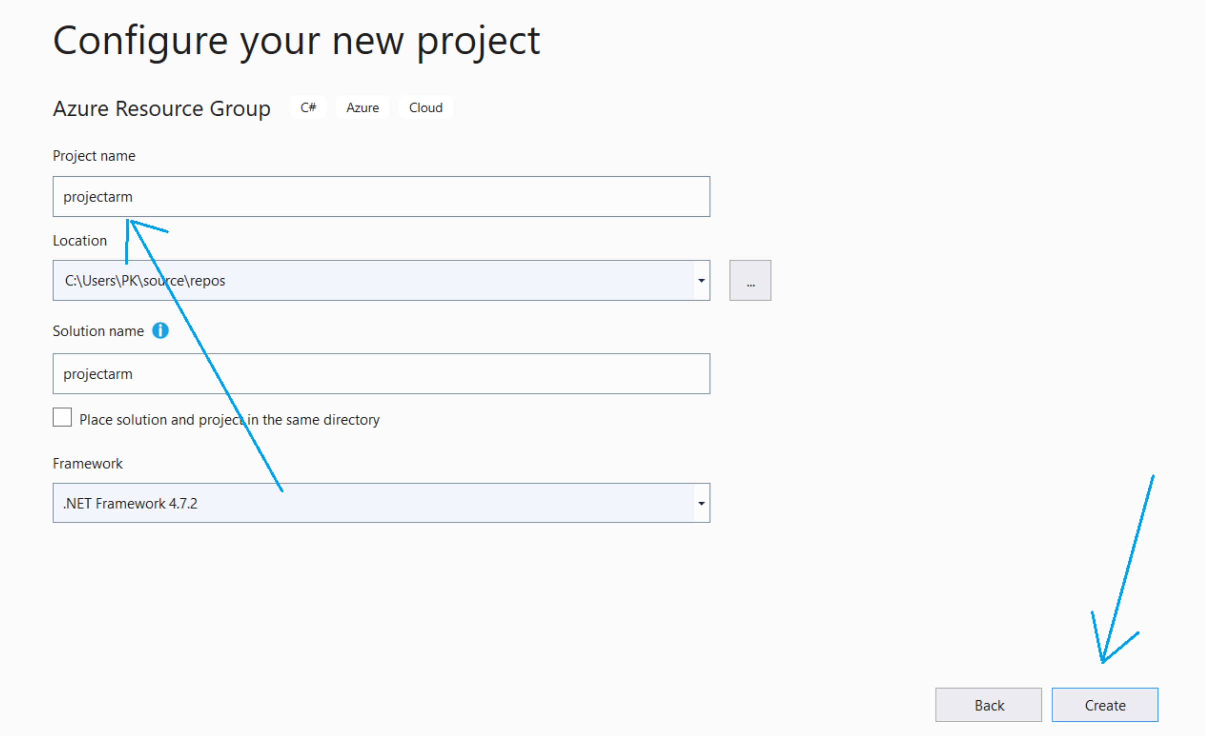
Blazor App
Project templates for creating Blazor apps that that run on the server in an ASP.NET Core app or in the browser on WebAssembly. These templates can be used to build web apps with rich dynamic user interfaces (UIs).
C# Linux macOS Windows Cloud Web

Azure Functions
A template to create an Azure Function project.
C# Azure Cloud

gRPC Service
A project template for creating a gRPC ASP.NET Core service using .NET Core.
C# Linux macOS Windows Cloud Service Web

Worker Service
A project template for creating a worker service using .NET Core

Now, enter the project name as **“projectarm”** as shown in the window below and click on **“Create”.**



Configure your new project

Azure Resource Group C# Azure Cloud

Project name: projectarm

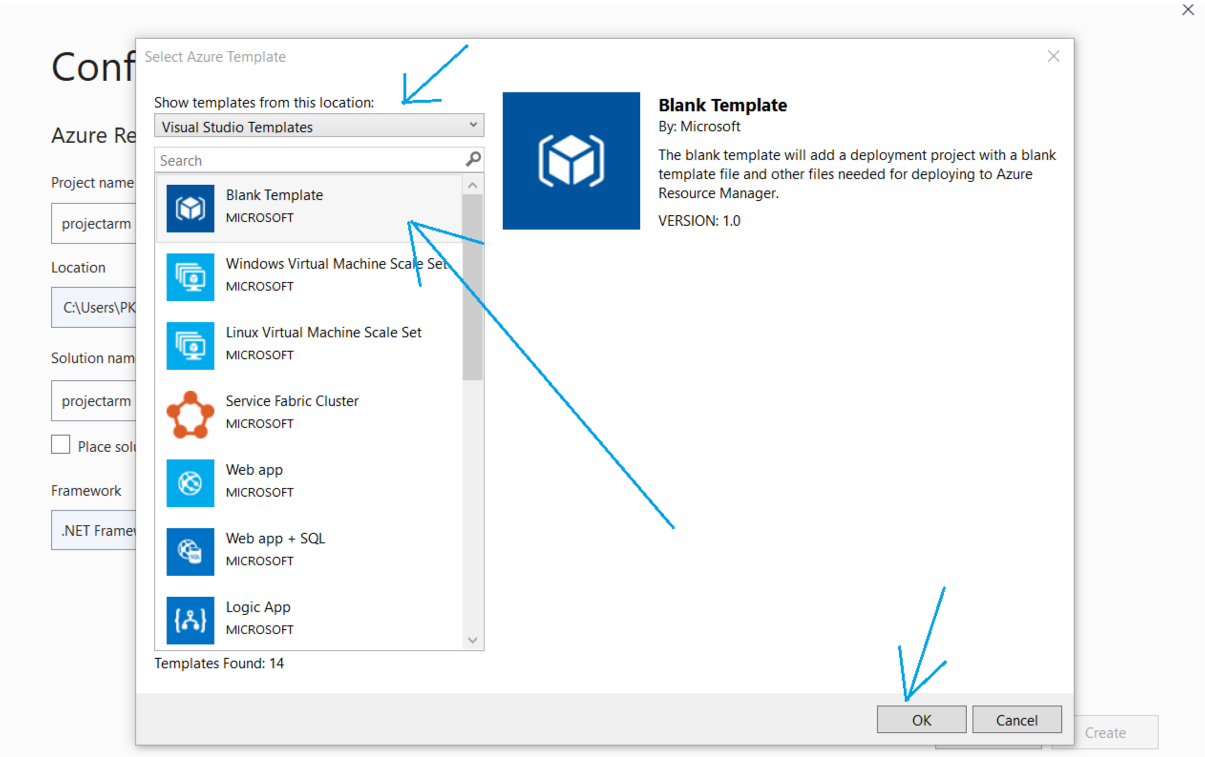
Location: C:\Users\PK\source\repos

Solution name: projectarm

□ Place solution and project in the same directory

Framework: .NET Framework 4.7.2

Step 2: Now select **“Visual Studio templates”** from drop down list and select **“Blank template”** as shown in the window below and click **“ok”.**



Select Azure Template

Show templates from this location: Visual Studio Templates

Blank Template

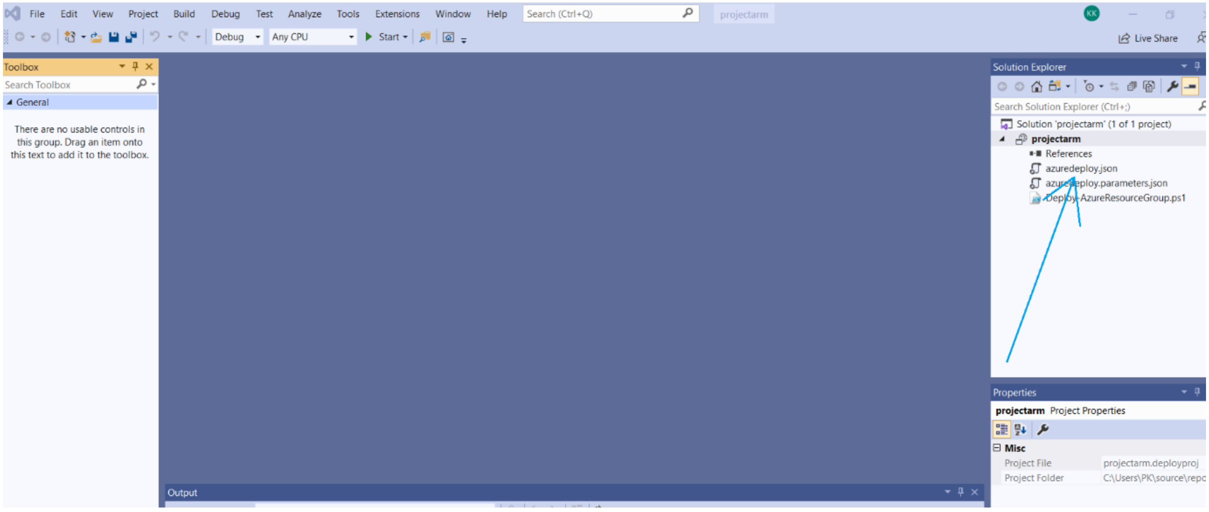
By: Microsoft

The blank template will add a deployment project with a blank template file and other files needed for deploying to Azure Resource Manager.

VERSION: 1.0

Templates Found: 14

Now click on **“azuredeploy.json”** file as shown in the window below.



There are no usable controls in this group. Drag an item onto this text to add it to the toolbox.

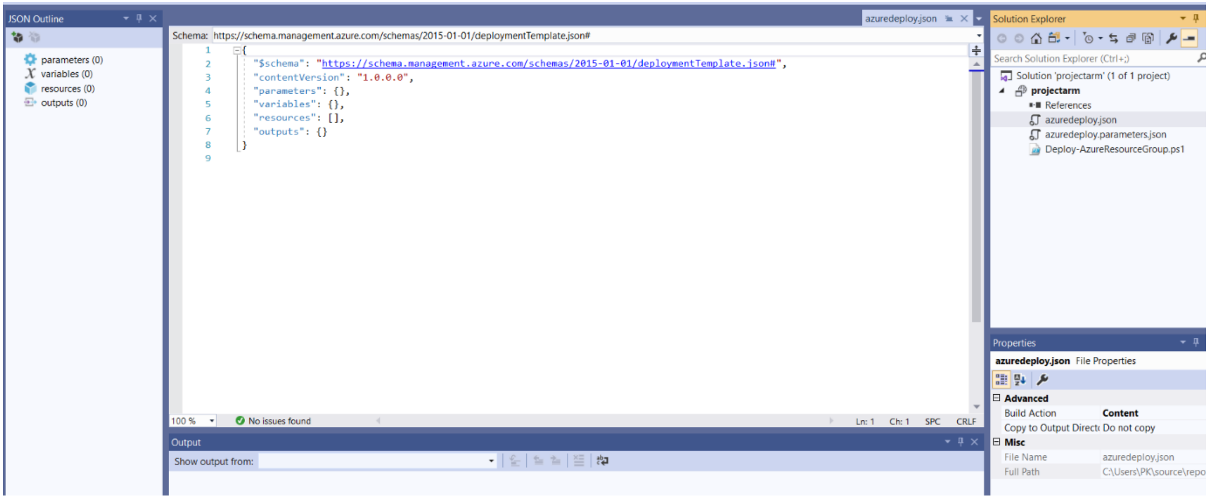
projectarm Project Properties

Misc

Project File projectarm.deployproj

Project Folder C:\Users\PK\source\repo

Below window shows the example of **“Blank template”.**



Schema: https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#

```
{
  "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {},
  "variables": {},
  "resources": [],
  "outputs": {}
}
```

azuredeploy.json File Properties

Advanced

Build Action Content

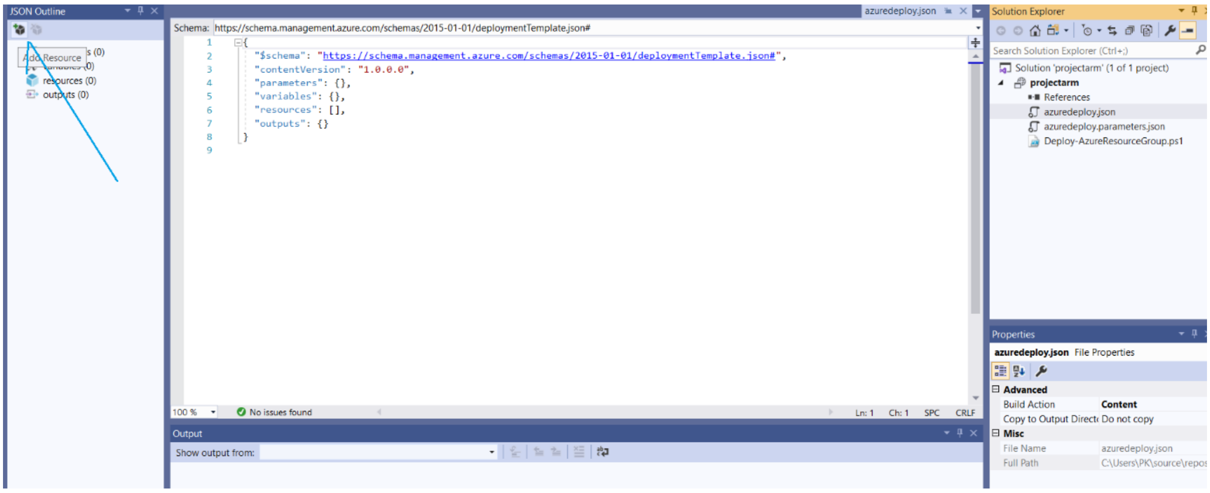
Copy to Output Directo Do not copy

Misc

File Name azuredeploy.json

Full Path C:\Users\PK\source\repo

Step 3: Now to create **“Virtual network”** click on **“Add resource”** as shown in the window below.



Schema: https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#

```
{
  "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {},
  "variables": {},
  "resources": [],
  "outputs": {}
}
```

azuredeploy.json File Properties

Advanced

Build Action Content

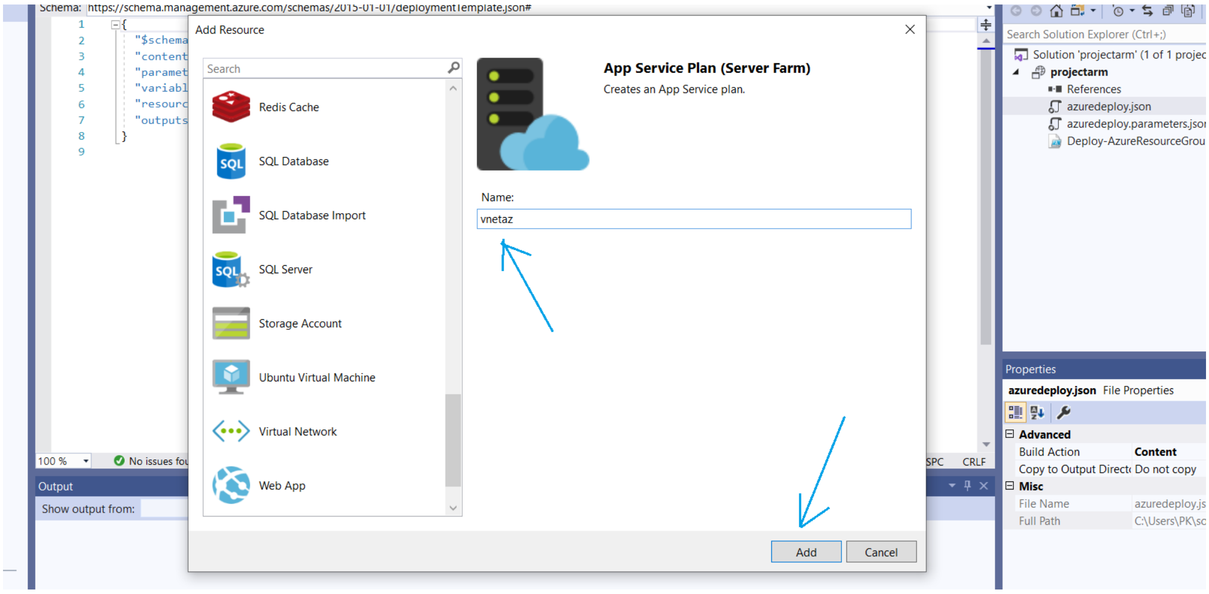
Copy to Output Directo Do not copy

Misc

File Name azuredeploy.json

Full Path C:\Users\PK\source\repos

Now, select the resource as **“Virtual Network”** and give it a name as **“vnetaz”** as shown in the window below and click on **“Add”**.

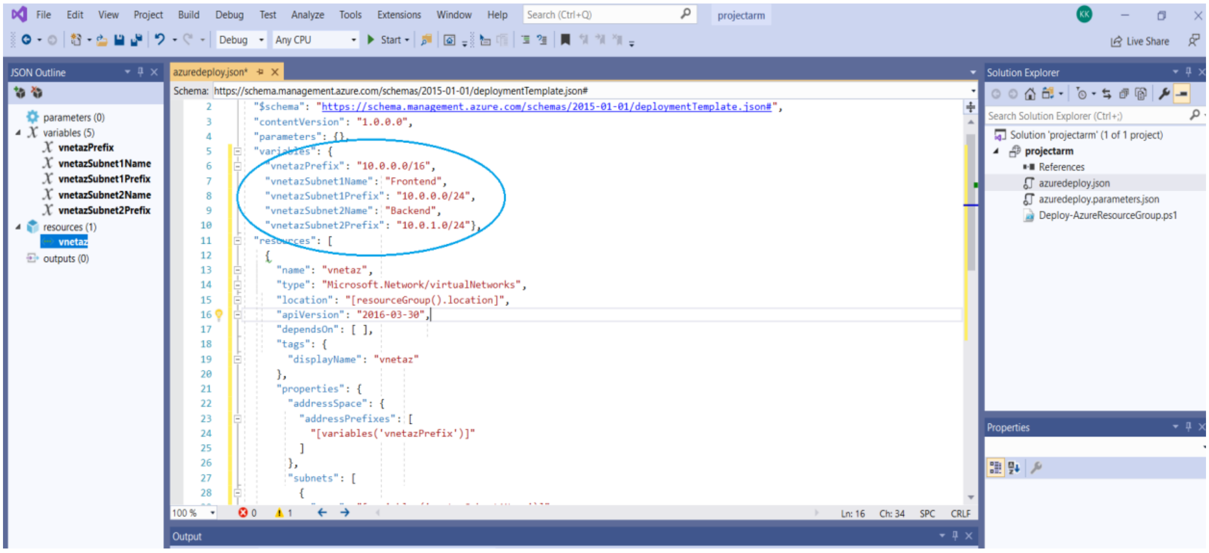


App Service Plan (Server Farm)

Creates an App Service plan.

Name: vnetaz

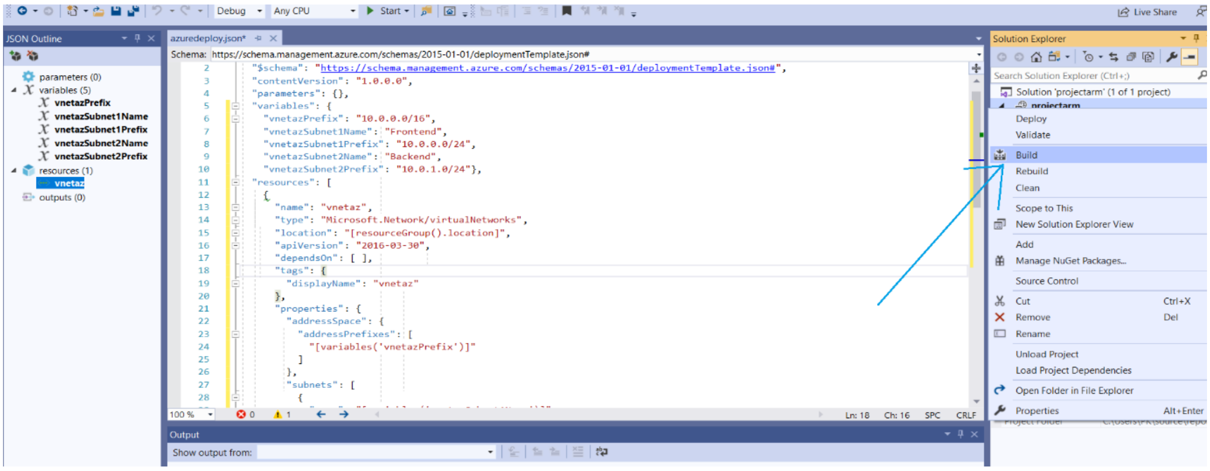
Now rename the subnet1 name as **“Frontend”** and subnet2 name as **“Backend”** as shown in the window below.



Schema: https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#

```
  "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {},
  "variables": {
    "vnetazPrefix": "10.0.0.0/16",
    "vnetazSubnet1Name": "Frontend",
    "vnetazSubnet1Prefix": "10.0.0.0/24",
    "vnetazSubnet2Name": "Backend",
    "vnetazSubnet2Prefix": "10.0.1.0/24"},
  "resources": [
    {
      "name": "vnetaz",
      "type": "Microsoft.Network/virtualNetworks",
      "location": "[resourceGroup().location]",
      "apiVersion": "2016-03-30",
      "dependsOn": [ ],
      "tags": {
        "displayName": "vnetaz"
      },
      "properties": {
        "addressSpace": {
          "addressPrefixes": [
            "[variables('vnetazPrefix')]"
          ]
        },
        "subnets": [
          {
```

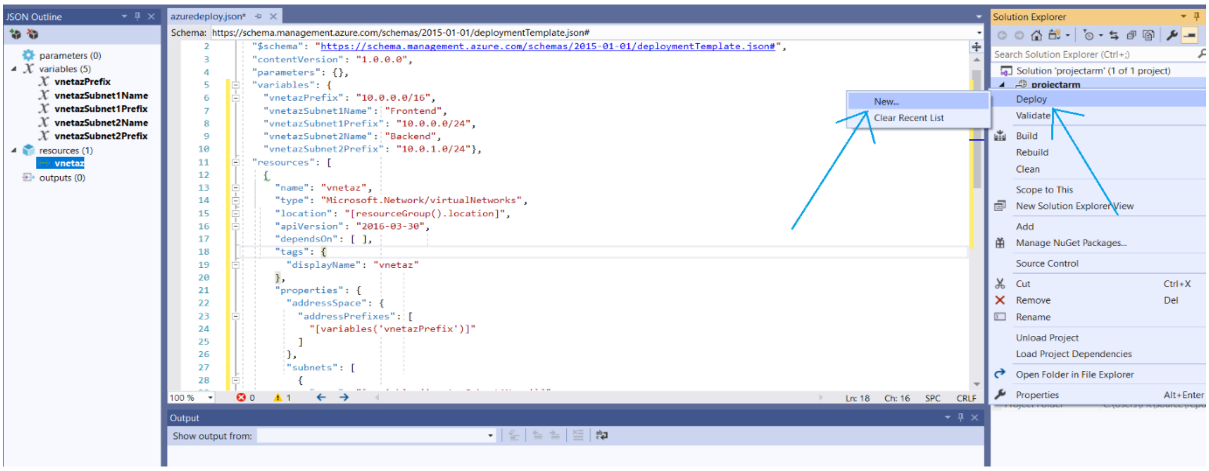
Step 4: Now right click on project **“projectarm”** and select **“Build”** as shown in the window below.



Schema: https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#

```
  "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {},
  "variables": {
    "vnetazPrefix": "10.0.0.0/16",
    "vnetazSubnet1Name": "Frontend",
    "vnetazSubnet1Prefix": "10.0.0.0/24",
    "vnetazSubnet2Name": "Backend",
    "vnetazSubnet2Prefix": "10.0.1.0/24"},
  "resources": [
    {
      "name": "vnetaz",
      "type": "Microsoft.Network/virtualNetworks",
      "location": "[resourceGroup().location]",
      "apiVersion": "2016-03-30",
      "dependsOn": [ ],
      "tags": {
        "displayName": "vnetaz"
      },
      "properties": {
        "addressSpace": {
          "addressPrefixes": [
            "[variables('vnetazPrefix')]"
          ]
        },
        "subnets": [
          {
```

Again right click on project **“projectarm”** and select **“Deploy”** and **“New”** as shown in the window below.

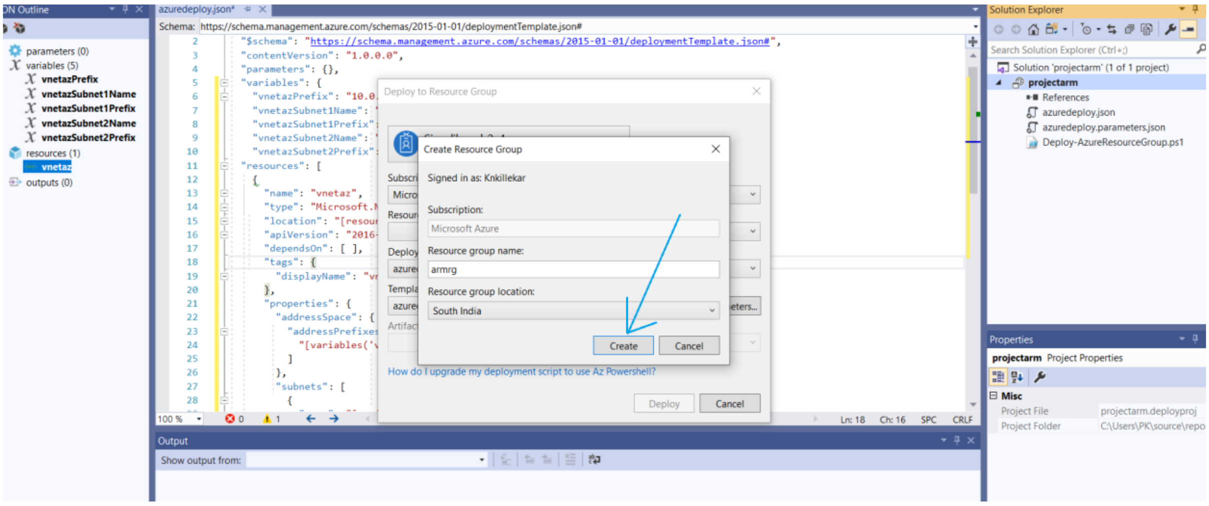


Schema: https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#

```
  "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {},
  "variables": {
    "vnetazPrefix": "10.0.0.0/16",
    "vnetazSubnet1Name": "Frontend",
    "vnetazSubnet1Prefix": "10.0.0.0/24",
    "vnetazSubnet2Name": "Backend",
    "vnetazSubnet2Prefix": "10.0.1.0/24"},
  "resources": [
    {
      "name": "vnetaz",
      "type": "Microsoft.Network/virtualNetworks",
      "location": "[resourceGroup().location]",
      "apiVersion": "2016-03-30",
      "dependsOn": [ ],
      "tags": {
        "displayName": "vnetaz"
      },
      "properties": {
        "addressSpace": {
          "addressPrefixes": [
            "[variables('vnetazPrefix')]"
          ]
        },
        "subnets": [
          {
```

Next select **“Resource group”** and click **“Create New”** and enter the following details as shown in the window below and click on **“Create”.**

- Resource group name: **armrg.**
- Resource group location : **South India**.



Deploy to Resource Group

Create Resource Group

Signed in as: Knkillekar

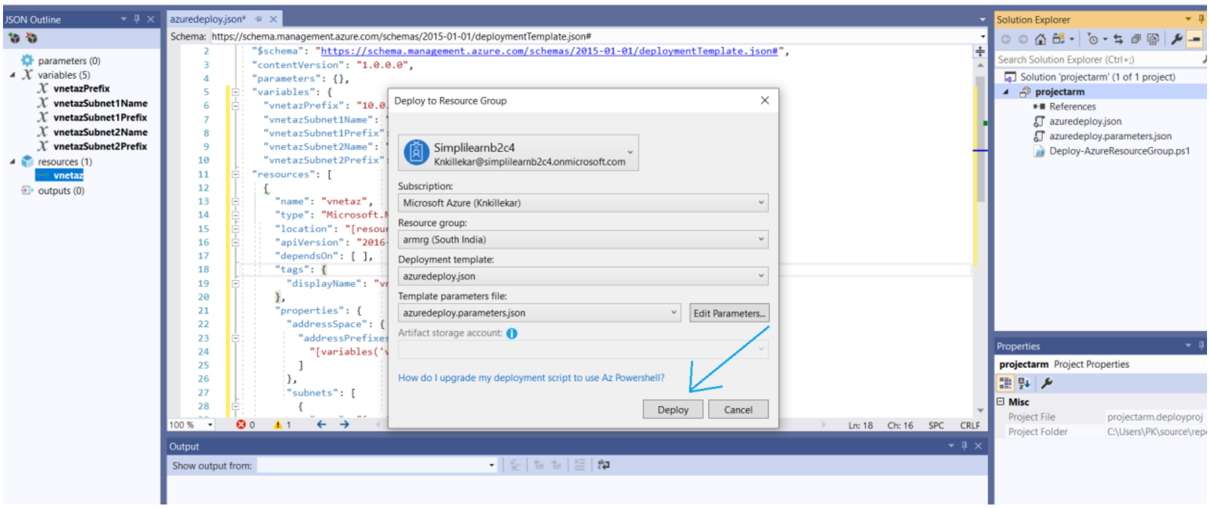
Subscription: Microsoft Azure

Resource group name: armrg

Resource group location: South India

Create
Cancel
How do I upgrade my deployment script to use Az Powershell?
Deploy
Cancel
Solution Explorer
Solution 'projectarm' (1 of 1 project)
projectarm
References
azuredeploy.json
azuredeploy.parameters.json
Deploy-AzureResourceGroup.ps1
Properties
projectarm Project Properties
Misc
Project File projectarm.deployproj
Project Folder C:\Users\PK\source\repo

Finally, click on **“Deploy”** to deploy the virtual network as shown in the window below.



Deploy to Resource Group

Simplilearnb2c4
Knkillekar@simplilearnb2c4.onmicrosoft.com

Subscription: Microsoft Azure (Knkillekar)

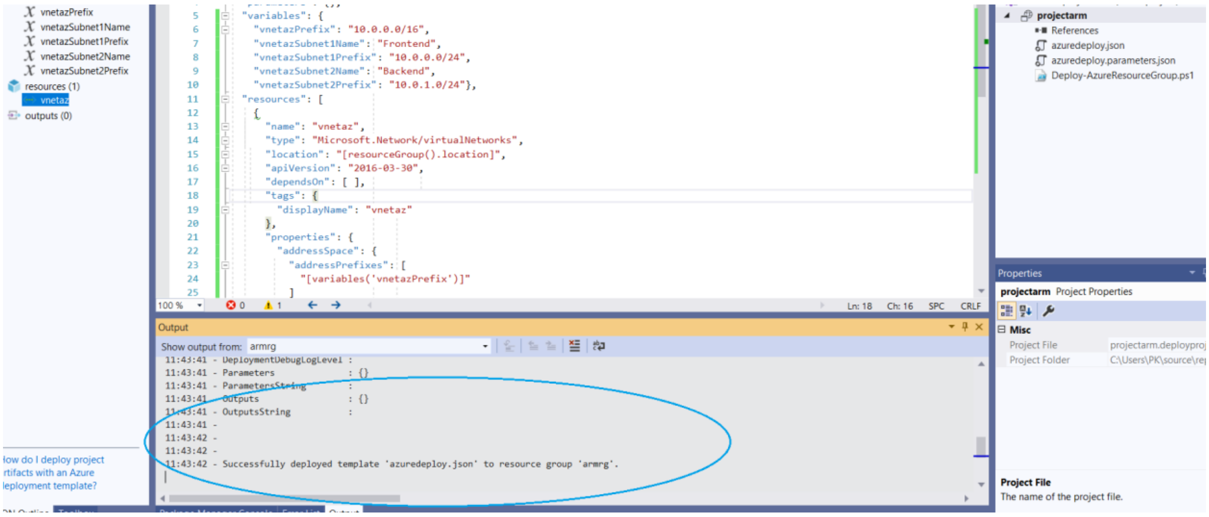
Resource group: armrg (South India)

Deployment template: azuredeploy.json

Template parameters file: azuredeploy.parameters.json

JSON Outline
parameters (0)
variables (5)
vnetazPrefix
vnetazSubnet1Name
vnetazSubnet1Prefix
vnetazSubnet2Name
vnetazSubnet2Prefix
resources (1)
vnetaz
outputs (0)
azuredeploy.json*
Schema: https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#
Edit Parameters...
Artifact storage account:
How do I upgrade my deployment script to use Az Powershell?
Deploy
Cancel
Output
Show output from:
Solution Explorer
Solution 'projectarm' (1 of 1 project)
projectarm
References
azuredeploy.json
azuredeploy.parameters.json
Deploy-AzureResourceGroup.ps1
Properties
projectarm Project Properties
Misc
Project File projectarm.deployproj
Project Folder C:\Users\PK\source\rep

The template is deployed successfully as shown in the window below.



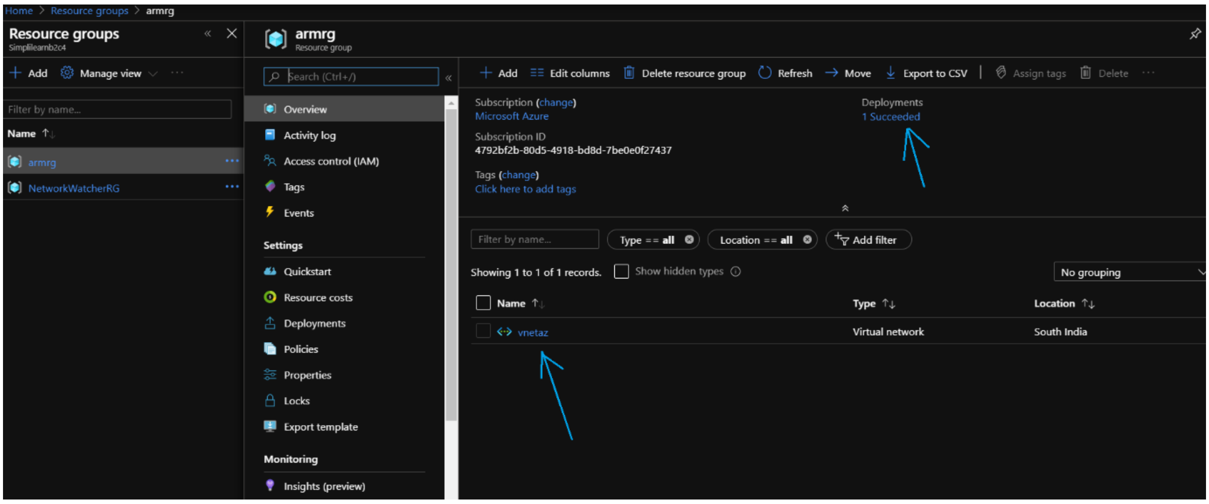
```
"variables": {
  "vnetazPrefix": "10.0.0.0/16",
  "vnetazSubnet1Name": "Frontend",
  "vnetazSubnet1Prefix": "10.0.0.0/24",
  "vnetazSubnet2Name": "Backend",
  "vnetazSubnet2Prefix": "10.0.1.0/24"},
"resources": [
  {
    "name": "vnetaz",
    "type": "Microsoft.Network/virtualNetworks",
    "location": "[resourceGroup().location]",
    "apiVersion": "2016-03-30",
    "dependsOn": [ ],
    "tags": {
      "displayName": "vnetaz"
    },
    "properties": {
      "addressSpace": {
        "addressPrefixes": [
          "[variables('vnetazPrefix')]"
        ]
```

Show output from: armrg

```
11:43:41 - DeploymentDebugLogLevel :
11:43:41 - Parameters              : {}
11:43:41 - ParametersString        :
11:43:41 - Outputs                 : {}
11:43:41 - OutputsString           :
11:43:41 -
11:43:42 -
11:43:42 -
11:43:42 - Successfully deployed template 'azuredeploy.json' to resource group 'armrg'.
```

Output
Ln: 18
Ch: 16
SPC
CRLF
Properties
projectarm Project Properties
Project File
The name of the project file.
How do I deploy project artifacts with an Azure deployment template?

The same can be verified in the **“Azure Portal”** as shown in the window below,the **“Virtual Network”** is deployed successfully.



Subscription (change)
Microsoft Azure

Subscription ID
4792bf2b-80d5-4918-bd8d-7be0e0f27437

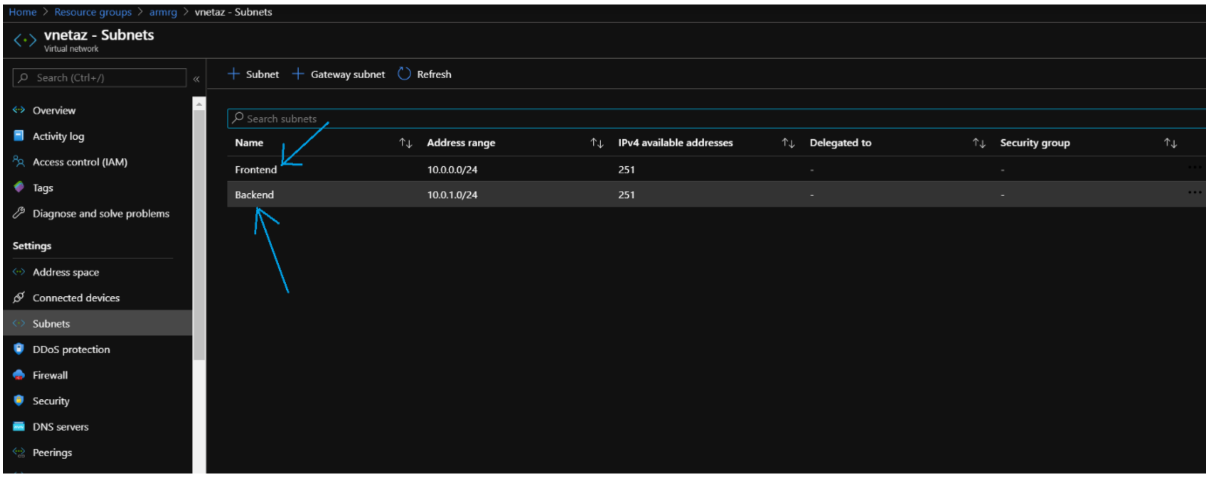
Tags (change)
Click here to add tags

Deployments
1 Succeeded

Showing 1 to 1 of 1 records.

Name	Type	Location
vnetaz	Virtual network	South India

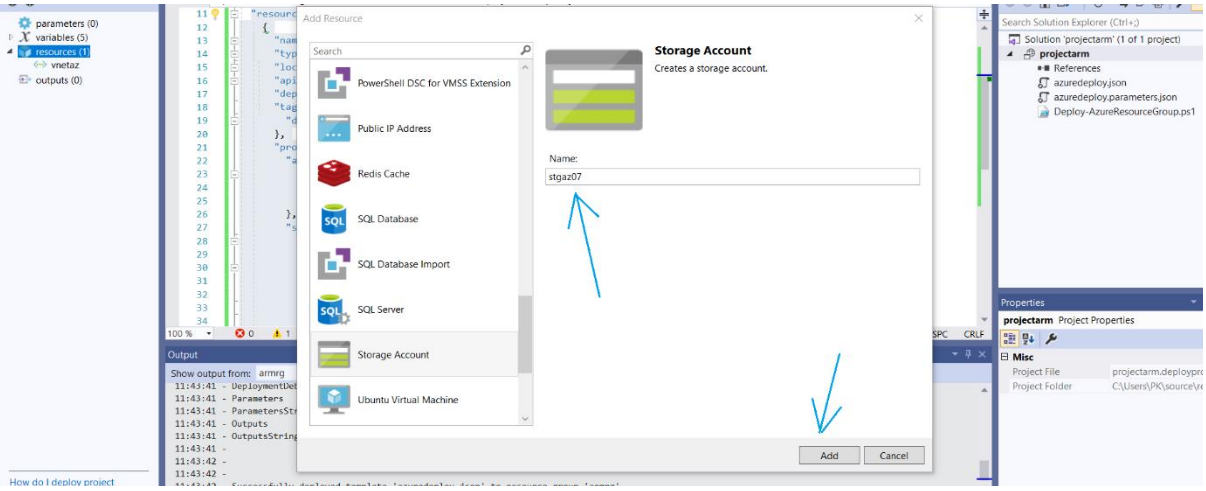
The two subnets **“Frontend”** and **“Backend”** are also created as shown in the window below.



Name	Address range	IPv4 available addresses	Delegated to	Security group
Frontend	10.0.0.0/24	251	-	-
Backend	10.0.1.0/24	251	-	-

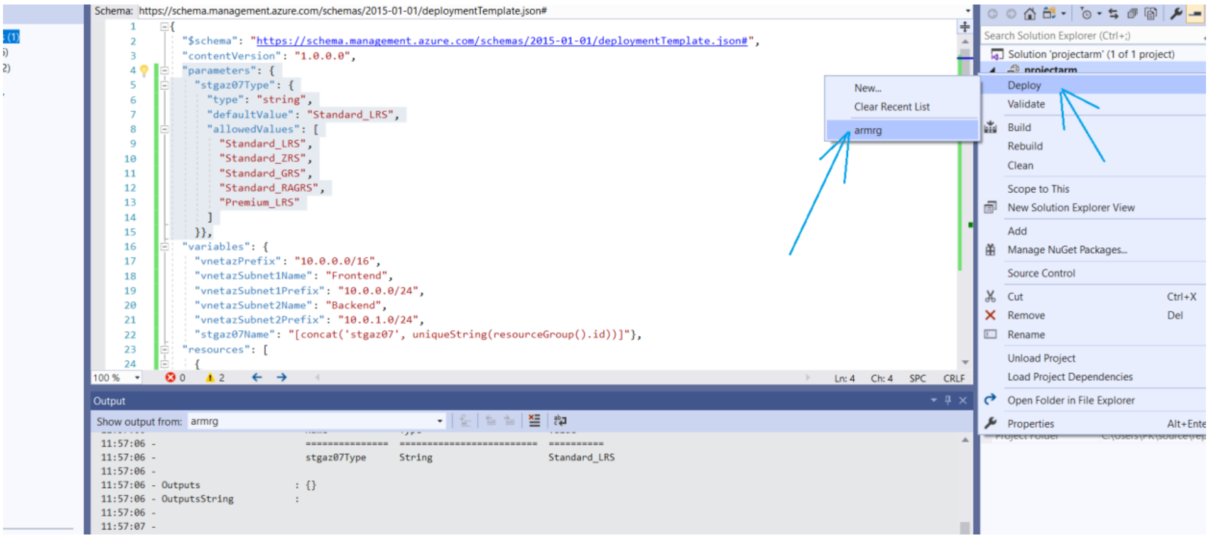
Task 2: Extend that with Compute & Storage.

Step 1: Now add resource as **“Storage Account”** and give it a name as **“stgaz07”** as shown in the window below and click on **“Add”.**



Name: stgaz07

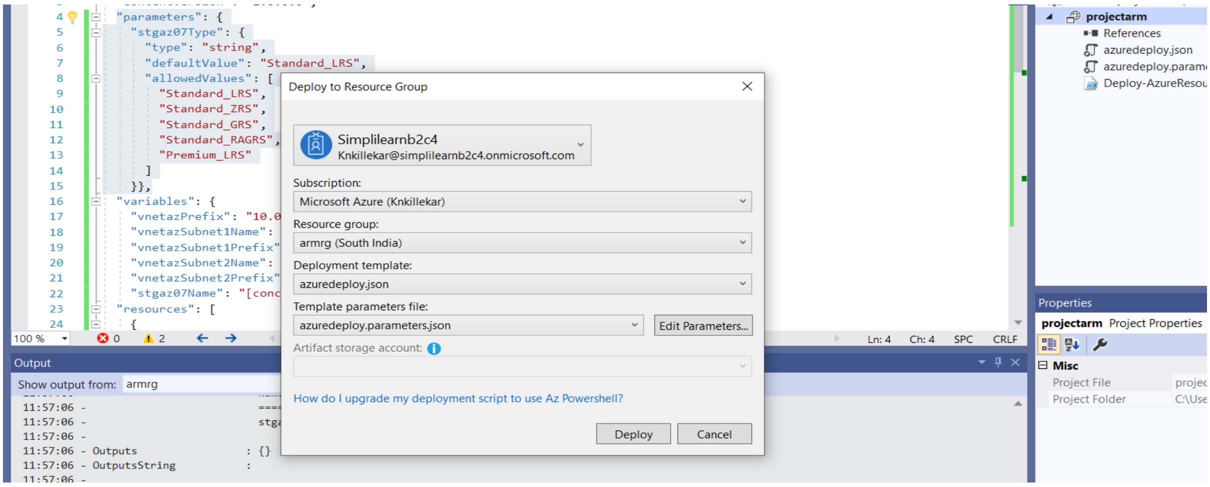
Now, right click on project **“projectarm”** and select **“Deploy”** and select the resource group as **“armrg”** as shown in the window below.


Schema: https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#
New...
Clear Recent List
armrg
Search Solution Explorer (Ctrl+;)
Solution 'projectarm' (1 of 1 project)
projectarm
Deploy
Validate
Build
Rebuild
Clean
Scope to This
New Solution Explorer View
Add
Manage NuGet Packages...
Source Control
Cut
Ctrl+X
Remove
Del
Rename
Unload Project
Load Project Dependencies
Open Folder in File Explorer
Properties
Alt+Ente
100 %
Ln: 4
Ch: 4
SPC
CRLF
Output
Show output from: armrg

```
{
  "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {
    "stgaz07Type": {
      "type": "string",
      "defaultValue": "Standard_LRS",
      "allowedValues": [
        "Standard_LRS",
        "Standard_ZRS",
        "Standard_GRS",
        "Standard_RAGRS",
        "Premium_LRS"
      ]
    }},
  "variables": {
    "vnetazPrefix": "10.0.0.0/16",
    "vnetazSubnet1Name": "Frontend",
    "vnetazSubnet1Prefix": "10.0.0.0/24",
    "vnetazSubnet2Name": "Backend",
    "vnetazSubnet2Prefix": "10.0.1.0/24",
    "stgaz07Name": "[concat('stgaz07', uniqueString(resourceGroup().id))]"},
  "resources": [
    {
```

```
11:57:06 -                  ===============  ========================  ==========
11:57:06 -                  stgaz07Type      String                    Standard_LRS
11:57:06 -
11:57:06 - Outputs         : {}
11:57:06 - OutputsString   :
11:57:06 -
11:57:07 -
```

Next, keep the settings as shown in the window below and click on **"Deploy".**



```
  "parameters": {
    "stgaz07Type": {
      "type": "string",
      "defaultValue": "Standard_LRS",
      "allowedValues": [
        "Standard_LRS",
        "Standard_ZRS",
        "Standard_GRS",
        "Standard_RAGRS",
        "Premium_LRS"
      ]
    }},
  "variables": {
    "vnetazPrefix": "10.0
    "vnetazSubnet1Name":
    "vnetazSubnet1Prefix"
    "vnetazSubnet2Name":
    "vnetazSubnet2Prefix"
    "stgaz07Name": "[conc
  "resources": [
    {
```

Deploy to Resource Group

Simplilearnb2c4
Knkillekar@simplilearnb2c4.onmicrosoft.com

Subscription: Microsoft Azure (Knkillekar)

Resource group: armrg (South India)

Deployment template: azuredeploy.json

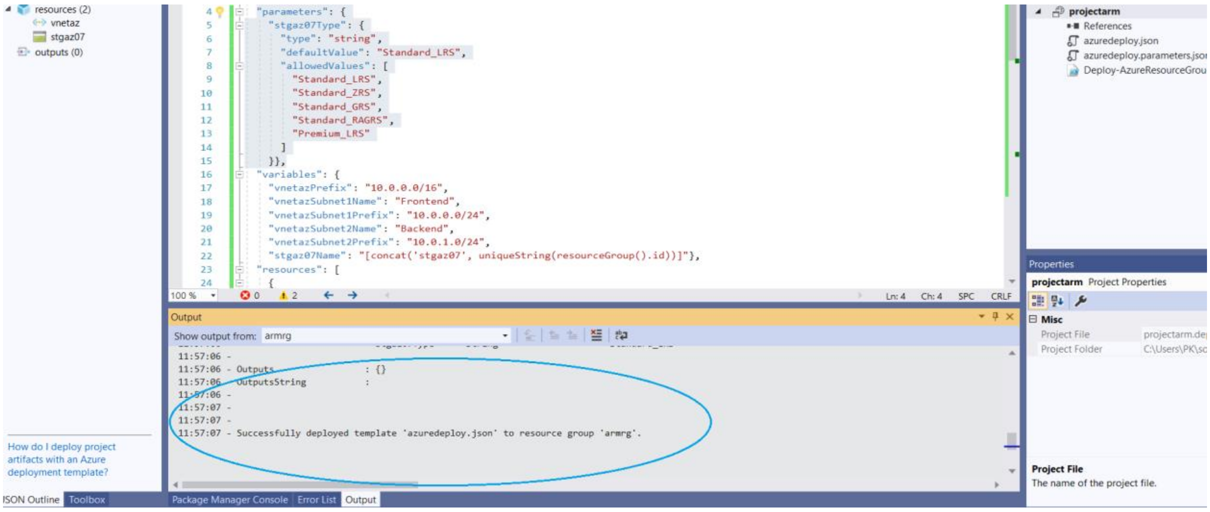
Template parameters file: azuredeploy.parameters.json

Artifact storage account:

How do I upgrade my deployment script to use Az Powershell?

```
11:57:06 -
11:57:06 -
11:57:06 -
11:57:06 - Outputs                    : {}
11:57:06 - OutputsString              :
11:57:06 -
```

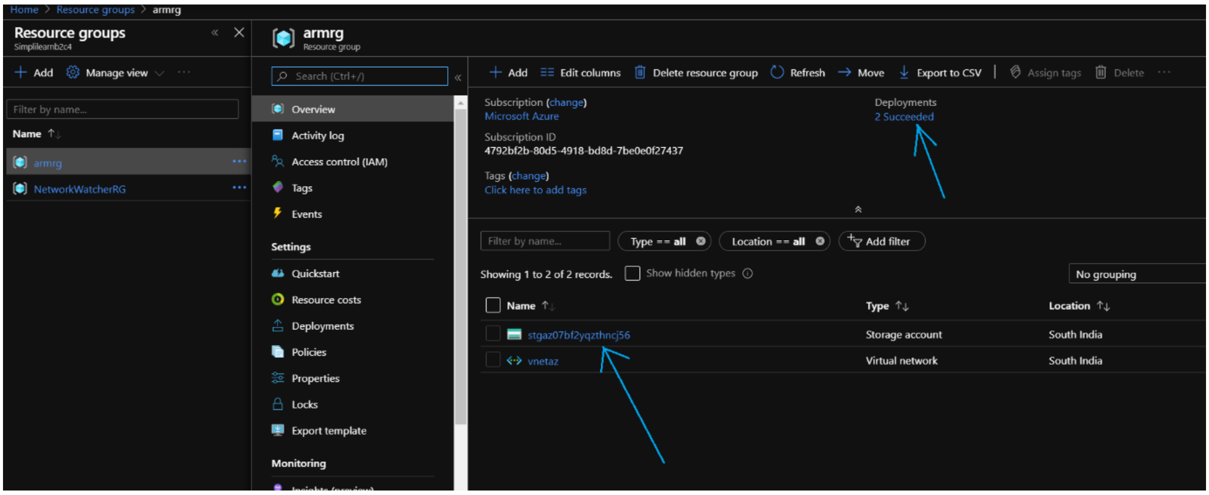
The template is deployed successfully as shown in the window below.



```
  "parameters": {
    "stgaz07Type": {
      "type": "string",
      "defaultValue": "Standard_LRS",
      "allowedValues": [
        "Standard_LRS",
        "Standard_ZRS",
        "Standard_GRS",
        "Standard_RAGRS",
        "Premium_LRS"
      ]
    }},
  "variables": {
    "vnetazPrefix": "10.0.0.0/16",
    "vnetazSubnet1Name": "Frontend",
    "vnetazSubnet1Prefix": "10.0.0.0/24",
    "vnetazSubnet2Name": "Backend",
    "vnetazSubnet2Prefix": "10.0.1.0/24",
    "stgaz07Name": "[concat('stgaz07', uniqueString(resourceGroup().id))]"},
  "resources": [
    {
```

```
11:57:06 -
11:57:06 - Outputs                    : {}
11:57:06 - OutputsString              :
11:57:06 -
11:57:07 -
11:57:07 -
11:57:07 - Successfully deployed template 'azuredeploy.json' to resource group 'armrg'.
```

The same can be verified in the **"Azure Portal"** as shown in the window below, the **"Storage Account"** is deployed successfully.



Subscription (change)
Microsoft Azure

Subscription ID
4792bf2b-80d5-4918-bd8d-7be0e0f27437

Tags (change)
Click here to add tags

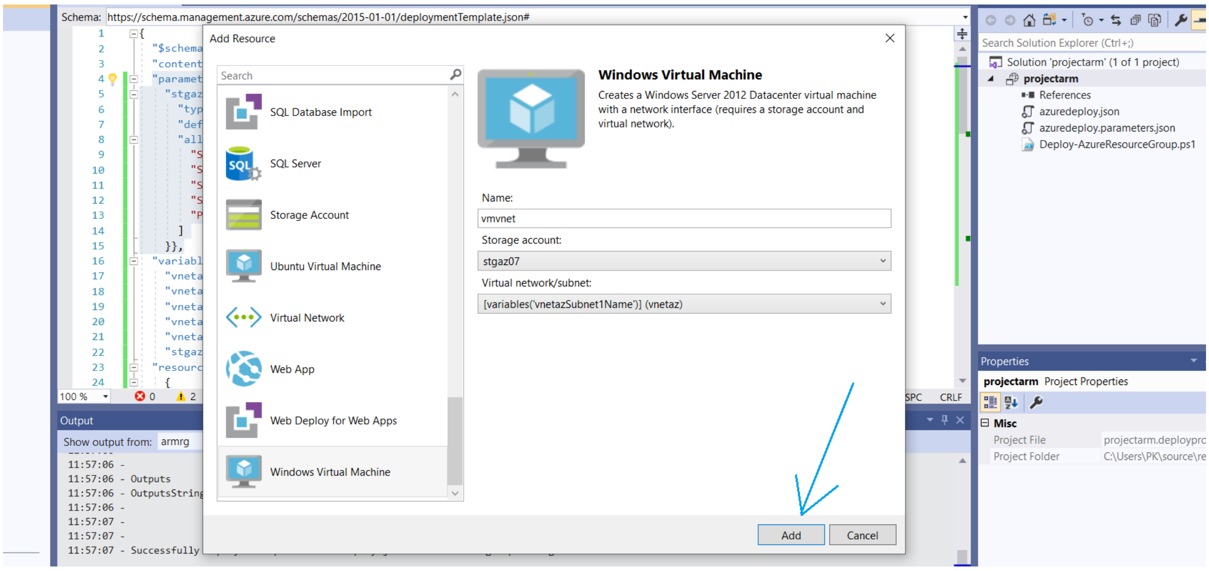
Deployments
2 Succeeded

Showing 1 to 2 of 2 records.

Name	Type	Location
stgaz07bf2yqzthncj56	Storage account	South India
vnetaz	Virtual network	South India

Step 2: Now add resource **“Windows virtual machine”** and give it a name **“vmvnet”** and select the following entries as shown in the window below and click on **“Add”**.

- Name : **vmvnet.**
- Storage account: **stgaz07.**
- Virtual Network/subnet:**[variables(‘vnetazSubnet1Name’)](vnetaz)**



Add Resource

Windows Virtual Machine

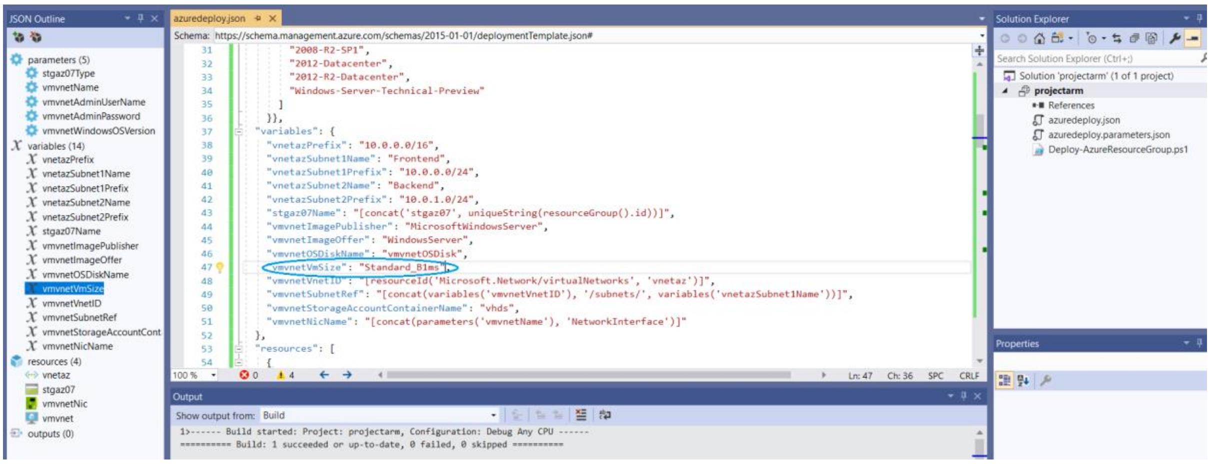
Creates a Windows Server 2012 Datacenter virtual machine with a network interface (requires a storage account and virtual network).

Name: vmvnet

Storage account: stgaz07

Virtual network/subnet: [variables('vnetazSubnet1Name')] (vnetaz)

Now change the **vmvnetVmsize** from "Standard_D2_v2" to "Standard_B1ms" as shown in the window below.



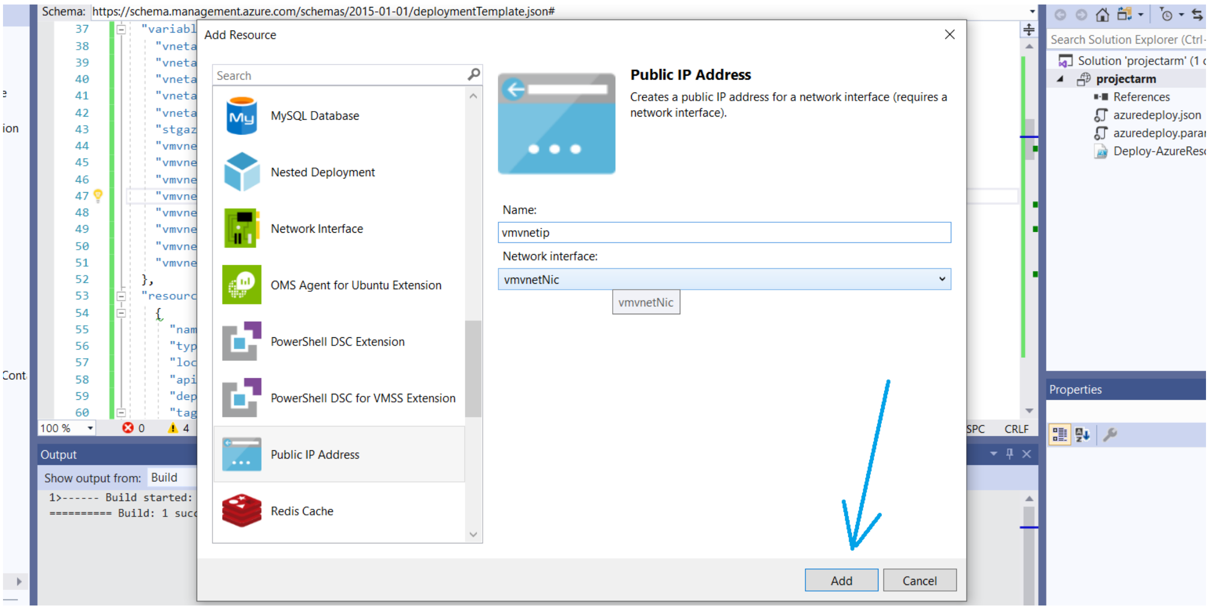
Schema: https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#

```
        "2008-R2-SP1",
        "2012-Datacenter",
        "2012-R2-Datacenter",
        "Windows-Server-Technical-Preview"
      ]
    }},
  "variables": {
    "vnetazPrefix": "10.0.0.0/16",
    "vnetazSubnet1Name": "Frontend",
    "vnetazSubnet1Prefix": "10.0.0.0/24",
    "vnetazSubnet2Name": "Backend",
    "vnetazSubnet2Prefix": "10.0.1.0/24",
    "stgaz07Name": "[concat('stgaz07', uniqueString(resourceGroup().id))]",
    "vmvnetImagePublisher": "MicrosoftWindowsServer",
    "vmvnetImageOffer": "WindowsServer",
    "vmvnetOSDiskName": "vmvnetOSDisk",
    "vmvnetVmSize": "Standard_B1ms",
    "vmvnetVnetID": "[resourceId('Microsoft.Network/virtualNetworks', 'vnetaz')]",
    "vmvnetSubnetRef": "[concat(variables('vmvnetVnetID'), '/subnets/', variables('vnetazSubnet1Name'))]",
    "vmvnetStorageAccountContainerName": "vhds",
    "vmvnetNicName": "[concat(parameters('vmvnetName'), 'NetworkInterface')]"
  },
  "resources": [
    {
```

Show output from: Build

```
1>------ Build started: Project: projectarm, Configuration: Debug Any CPU ------
========== Build: 1 succeeded or up-to-date, 0 failed, 0 skipped ==========
```

Step 3: Now add resource as **"Public IP Address"** and give it a name as **"vmvnetip"** as shown in the window below and click on **"Add"**.



Schema: https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#

Add Resource

Public IP Address

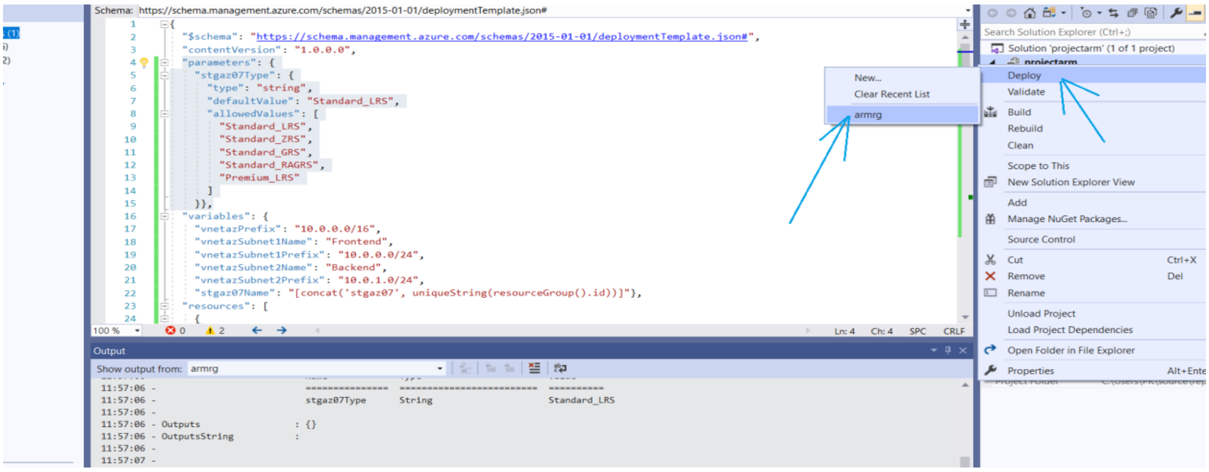
Creates a public IP address for a network interface (requires a network interface).

Name: vmvnetip

Network interface: vmvnetNic

```
1>------ Build started:
========== Build: 1 suc
```

Next, right click on project **"projectarm"** and select **"Deploy"** and select the resource group as **"armrg"** as shown in the window below.



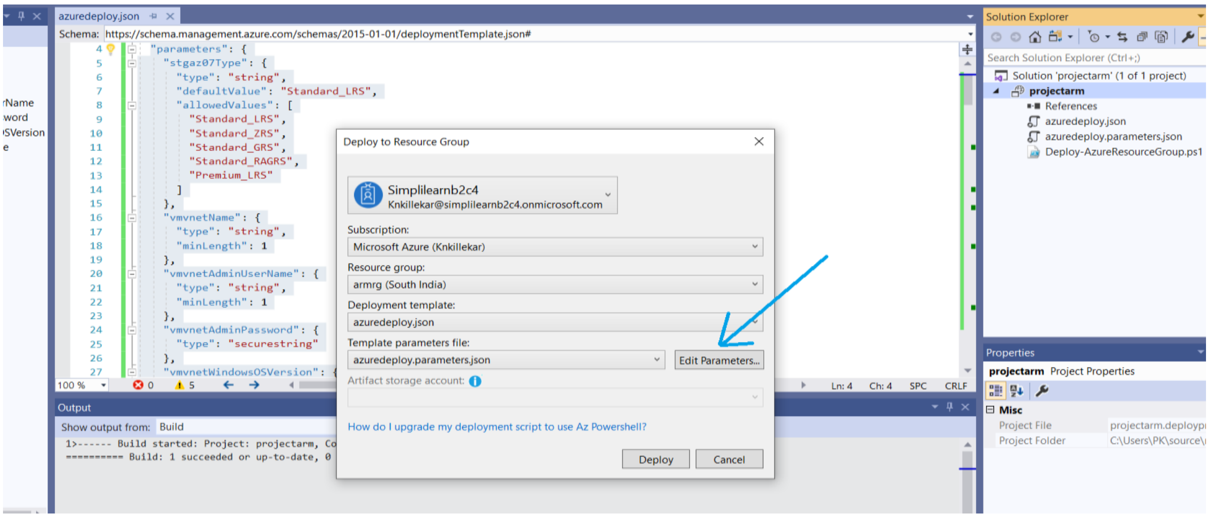
Schema: https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#

```
{
  "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {
    "stgaz07Type": {
      "type": "string",
      "defaultValue": "Standard_LRS",
      "allowedValues": [
        "Standard_LRS",
        "Standard_ZRS",
        "Standard_GRS",
        "Standard_RAGRS",
        "Premium_LRS"
      ]
    }},
  "variables": {
    "vnetazPrefix": "10.0.0.0/16",
    "vnetazSubnet1Name": "Frontend",
    "vnetazSubnet1Prefix": "10.0.0.0/24",
    "vnetazSubnet2Name": "Backend",
    "vnetazSubnet2Prefix": "10.0.1.0/24",
    "stgaz07Name": "[concat('stgaz07', uniqueString(resourceGroup().id))]"},
  "resources": [
    {
```

Output — Show output from: armrg

```
11:57:06 -                              ===============    ========================    ==========
11:57:06 -                              stgaz07Type        String                      Standard_LRS
11:57:06 -
11:57:06 - Outputs                    : {}
11:57:06 - OutputsString              :
11:57:06 -
11:57:07 -
```

Now click on **"Edit Parameters"** option as show in the window below and enter the following details and click on **"Save"**.



azuredeploy.json

Schema: https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#

```
  "parameters": {
    "stgaz07Type": {
      "type": "string",
      "defaultValue": "Standard_LRS",
      "allowedValues": [
        "Standard_LRS",
        "Standard_ZRS",
        "Standard_GRS",
        "Standard_RAGRS",
        "Premium_LRS"
      ]
    },
    "vmvnetName": {
      "type": "string",
      "minLength": 1
    },
    "vmvnetAdminUserName": {
      "type": "string",
      "minLength": 1
    },
    "vmvnetAdminPassword": {
      "type": "securestring"
    },
    "vmvnetWindowsOSVersion": {
```

Output — Show output from: Build

```
1>------ Build started: Project: projectarm, Co
========== Build: 1 succeeded or up-to-date, 0
```

Deploy to Resource Group

Simplilearnb2c4
Knkillekar@simplilearnb2c4.onmicrosoft.com

Subscription: Microsoft Azure (Knkillekar)

Resource group: armrg (South India)

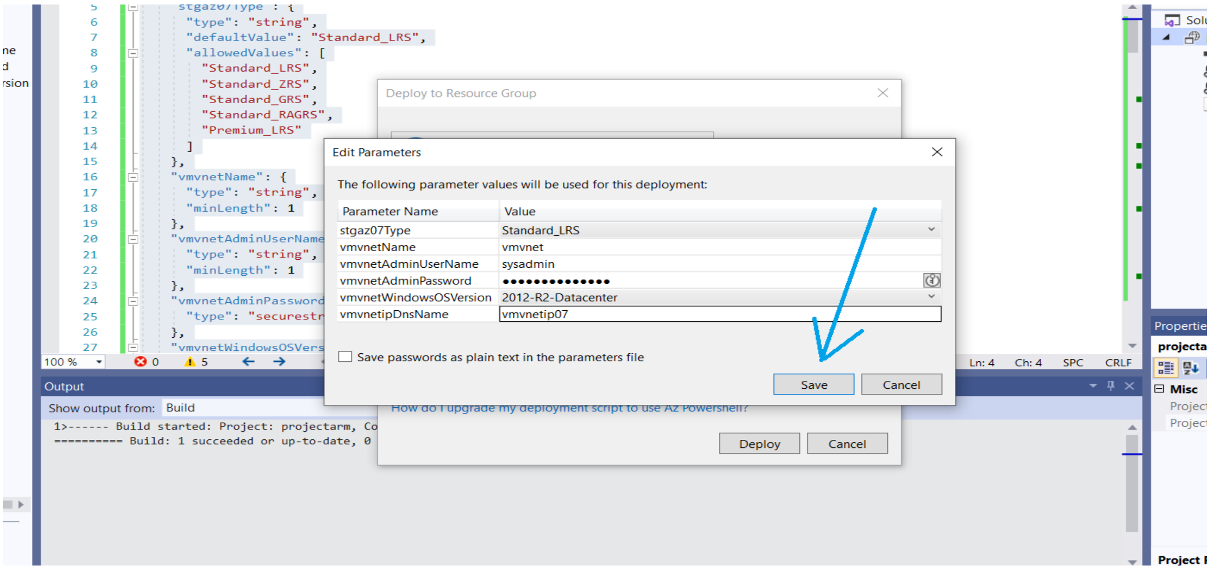
Deployment template: azuredeploy.json

Template parameters file: azuredeploy.parameters.json

Artifact storage account:

How do I upgrade my deployment script to use Az Powershell?

Edit Parameters...
Deploy
Cancel
Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'projectarm' (1 of 1 project)
projectarm
References
azuredeploy.json
azuredeploy.parameters.json
Deploy-AzureResourceGroup.ps1
Properties
projectarm Project Properties
Misc
Project File projectarm.deployp
Project Folder C:\Users\PK\source\



```
      "type": "string",
      "defaultValue": "Standard_LRS",
      "allowedValues": [
        "Standard_LRS",
        "Standard_ZRS",
        "Standard_GRS",
        "Standard_RAGRS",
        "Premium_LRS"
      ]
    },
    "vmvnetName": {
      "type": "string",
      "minLength": 1
    },
    "vmvnetAdminUserName
      "type": "string",
      "minLength": 1
    },
    "vmvnetAdminPassword
      "type": "securestr
    },
    "vmvnetWindowsOSVers
```

Deploy to Resource Group

Edit Parameters

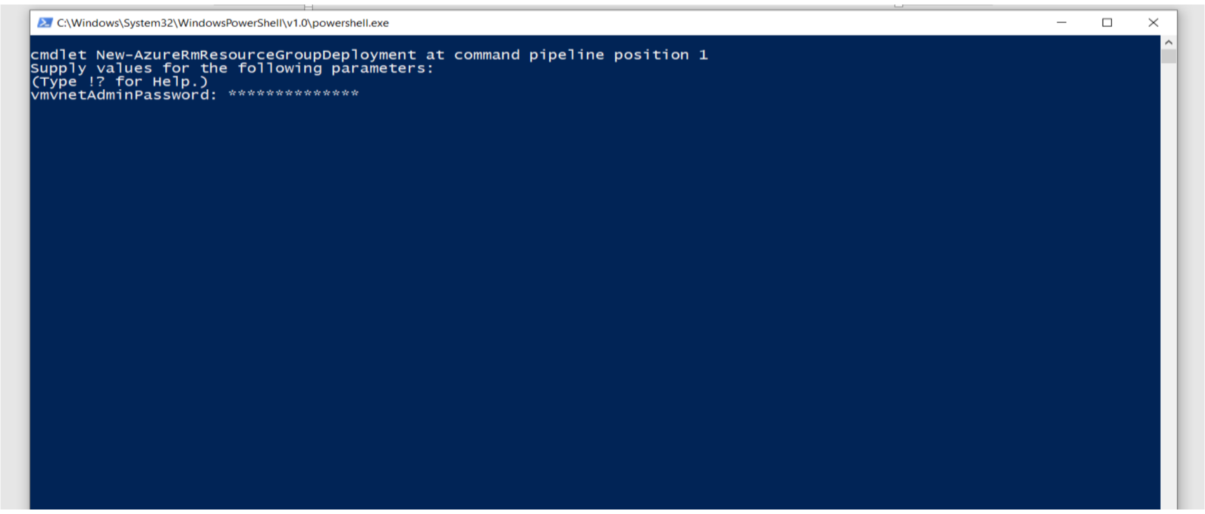
The following parameter values will be used for this deployment:

Parameter Name	Value
stgaz07Type	Standard_LRS
vmvnetName	vmvnet
vmvnetAdminUserName	sysadmin
vmvnetAdminPassword	••••••••••••••
vmvnetWindowsOSVersion	2012-R2-Datacenter
vmvnetipDnsName	vmvnetip07

□ Save passwords as plain text in the parameters file

```
1>------ Build started: Project: projectarm, Co
========== Build: 1 succeeded or up-to-date, 0
```

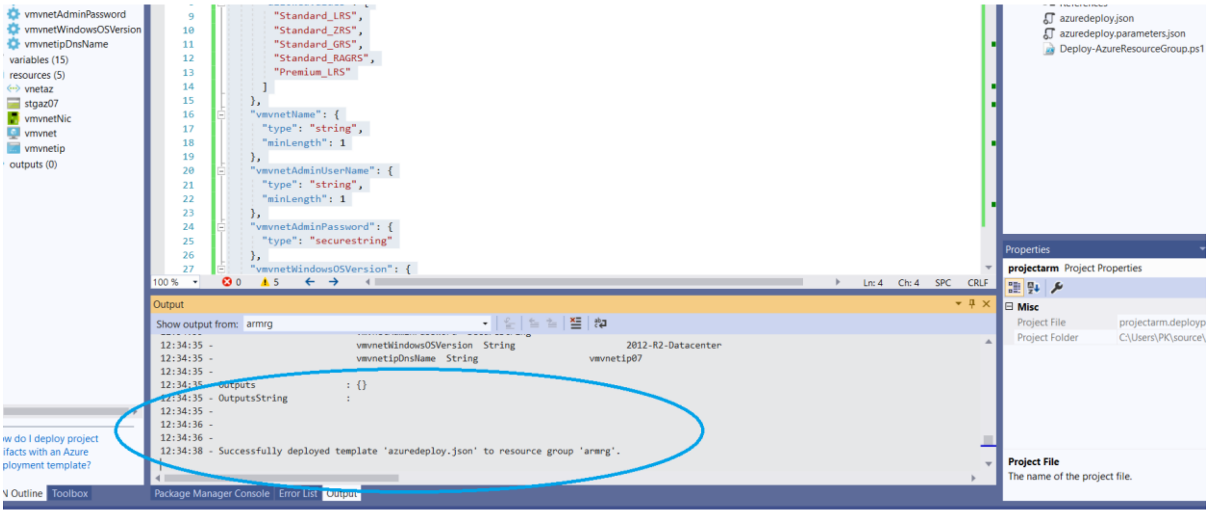
Finally, click on **"Deploy"** and a powershell window will appear asking to enter the password as shown in the window below.



C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe

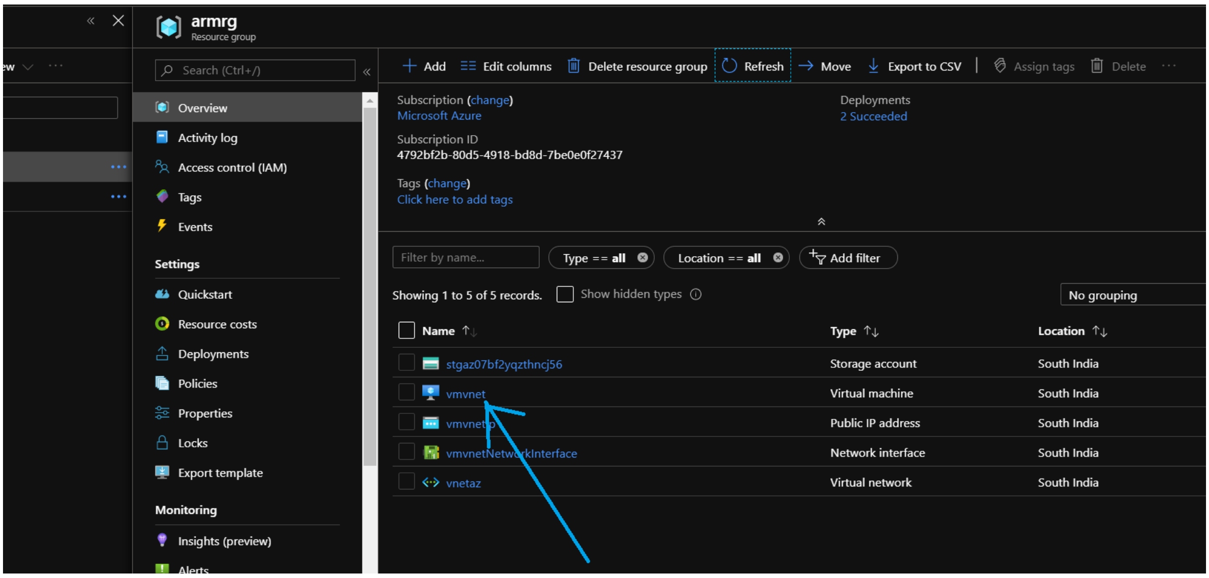
```
cmdlet New-AzureRmResourceGroupDeployment at command pipeline position 1
Supply values for the following parameters:
(Type !? for Help.)
vmvnetAdminPassword: **************
```

The template is deployed successfully as shown in the window below.



```
12:34:35 -                vmvnetWindowsOSVersion  String                 2012-R2-Datacenter
12:34:35 -                vmvnetipDnsName  String                 vmvnetip07
12:34:35 -
12:34:35 - Outputs                  : {}
12:34:35 - OutputsString            :
12:34:35 -
12:34:36 -
12:34:36 -
12:34:38 - Successfully deployed template 'azuredeploy.json' to resource group 'armrg'.
```

The same can be verified in the **“Azure Portal”** as shown in the window below,the **“vmvnet”** is deployed successfully.



Subscription (change)
Microsoft Azure

Subscription ID
4792bf2b-80d5-4918-bd8d-7be0e0f27437

Tags (change)
Click here to add tags

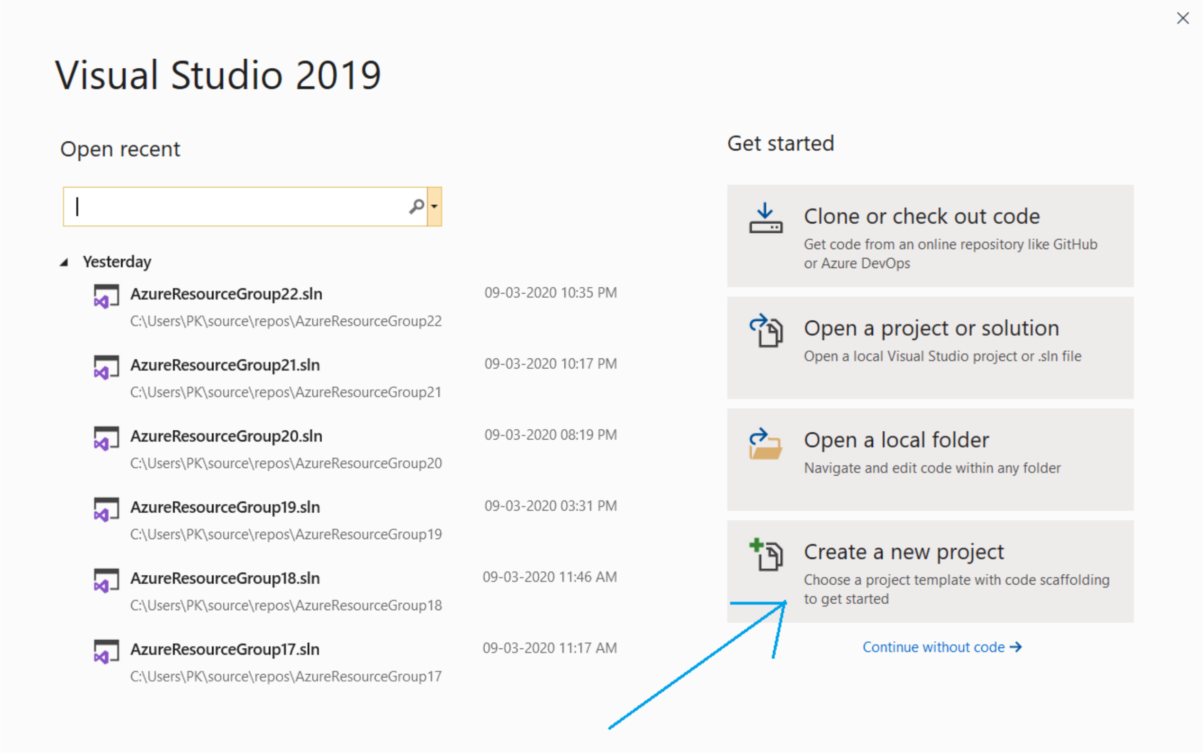
Deployments
2 Succeeded

Showing 1 to 5 of 5 records.

Name	Type	Location
stgaz07bf2yqzthncj56	Storage account	South India
vmvnet	Virtual machine	South India
vmvnetip	Public IP address	South India
vmvnetNetworkInterface	Network interface	South India
vnetaz	Virtual network	South India

Task 3: Create the Storage account for Images & implement CDN.

Step 1: Open visual studio 2019 and click on **“Create a new project”** as shown in the window below.



Visual Studio 2019

Open recent

Yesterday

AzureResourceGroup22.sln C:\Users\PK\source\repos\AzureResourceGroup22	09-03-2020 10:35 PM
AzureResourceGroup21.sln C:\Users\PK\source\repos\AzureResourceGroup21	09-03-2020 10:17 PM
AzureResourceGroup20.sln C:\Users\PK\source\repos\AzureResourceGroup20	09-03-2020 08:19 PM
AzureResourceGroup19.sln C:\Users\PK\source\repos\AzureResourceGroup19	09-03-2020 03:31 PM
AzureResourceGroup18.sln C:\Users\PK\source\repos\AzureResourceGroup18	09-03-2020 11:46 AM
AzureResourceGroup17.sln C:\Users\PK\source\repos\AzureResourceGroup17	09-03-2020 11:17 AM

Get started

Clone or check out code
Get code from an online repository like GitHub or Azure DevOps

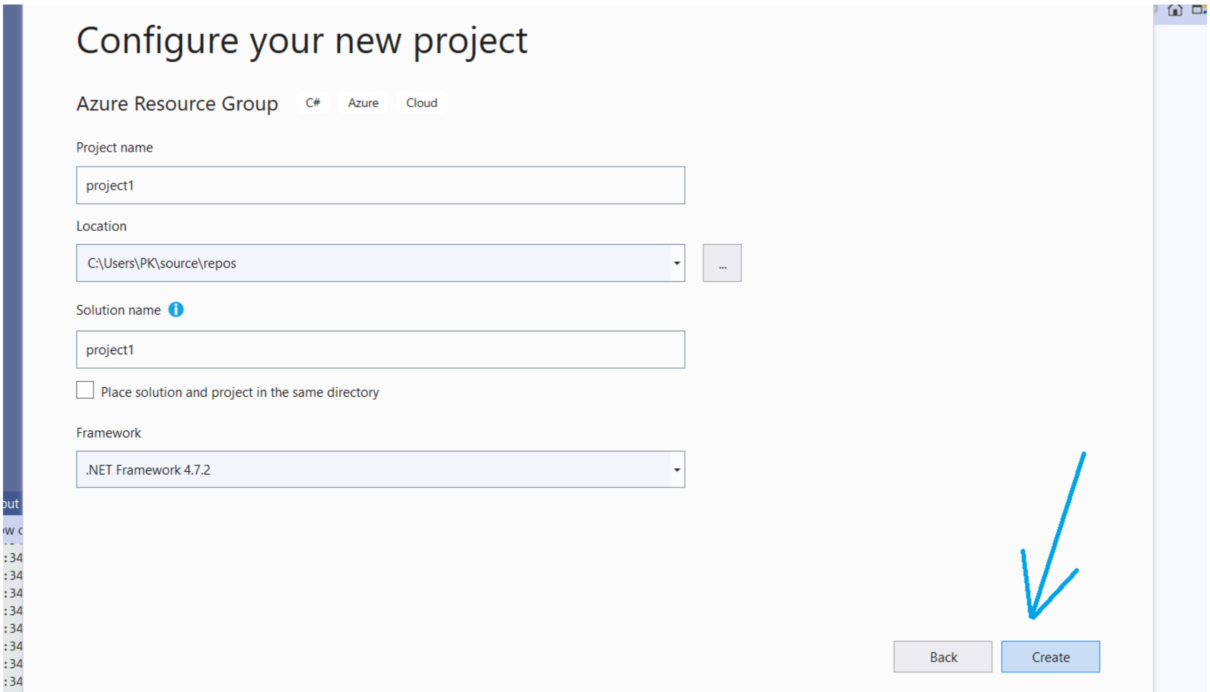
Open a project or solution
Open a local Visual Studio project or .sln file

Open a local folder
Navigate and edit code within any folder

Create a new project
Choose a project template with code scaffolding to get started

Continue without code →

Now, enter the project name as **“project1”** as shown in the window below and click on **“Create”.**



Configure your new project

Azure Resource Group C# Azure Cloud

Project name: project1

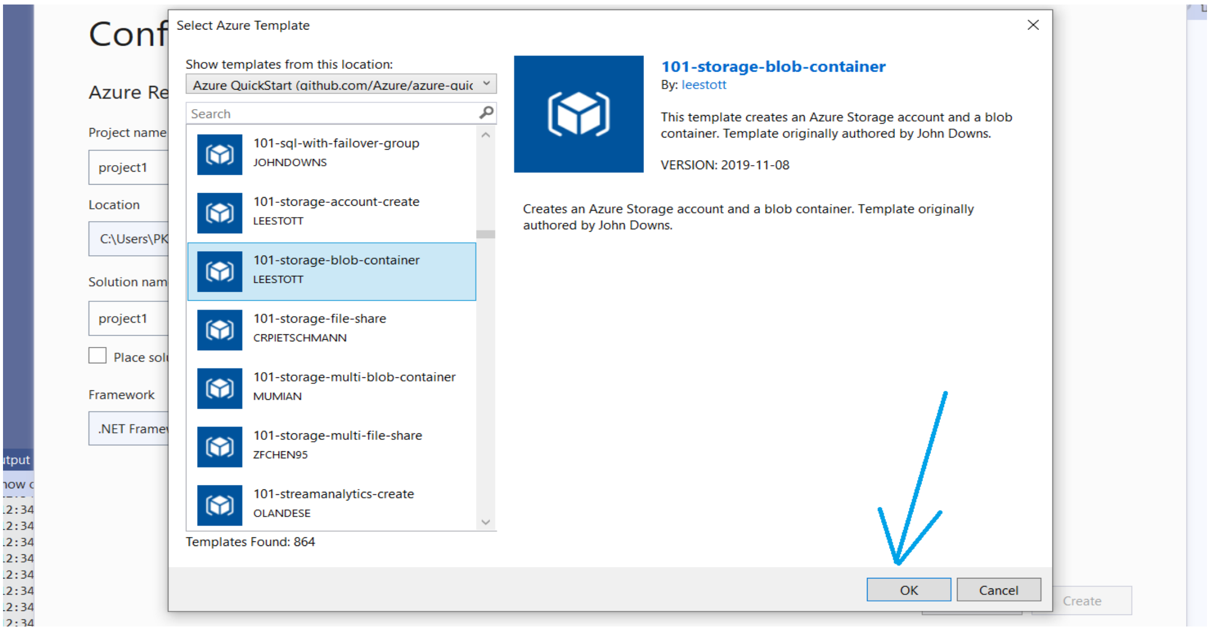
Location: C:\Users\PK\source\repos

Solution name: project1

□ Place solution and project in the same directory

Framework: .NET Framework 4.7.2

Step 2: Now select **“Azure Quickstart”** from drop down list and select **“storage-blob-container”** as shown in the window below and click **“ok”.**



Select Azure Template

Show templates from this location: Azure QuickStart (github.com/Azure/azure-quic

Search

- 101-sql-with-failover-group JOHNDOWNS
- 101-storage-account-create LEESTOTT
- 101-storage-blob-container LEESTOTT
- 101-storage-file-share CRPIETSCHMANN
- 101-storage-multi-blob-container MUMIAN
- 101-storage-multi-file-share ZFCHEN95
- 101-streamanalytics-create OLANDESE

Templates Found: 864

101-storage-blob-container

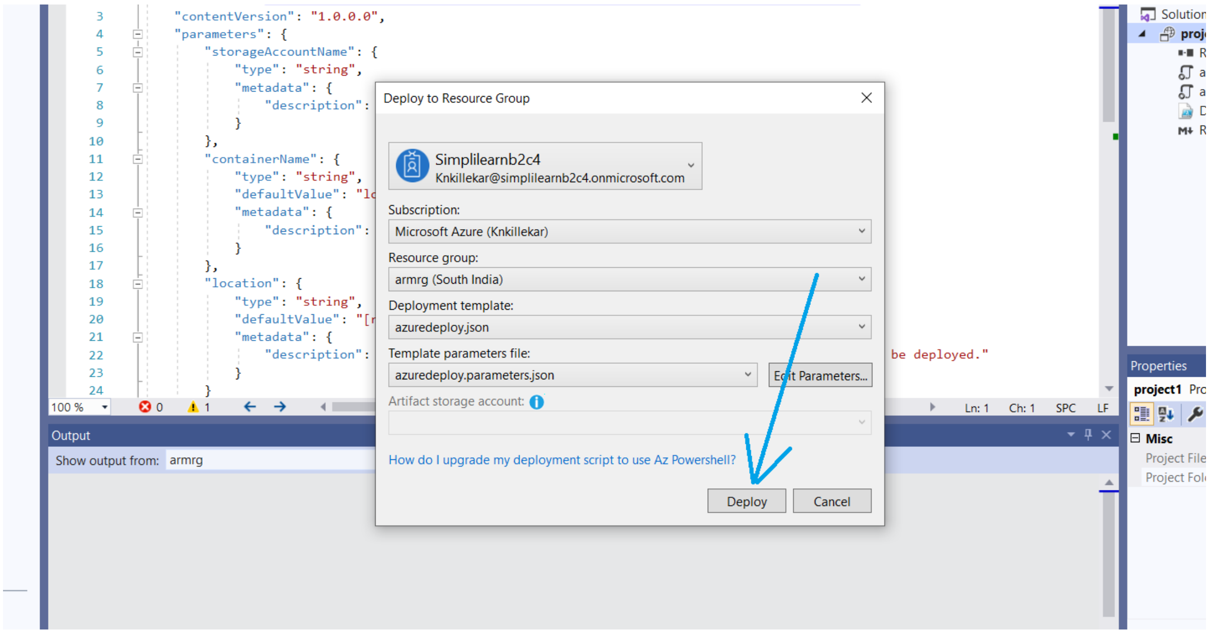
By: leestott

This template creates an Azure Storage account and a blob container. Template originally authored by John Downs.

VERSION: 2019-11-08

Creates an Azure Storage account and a blob container. Template originally authored by John Downs.

Now, right click on project **“project1”** and select **“Deploy”** and **“New”** as shown in the window below. Select the details as shown in the window below. Finally, click on **“Deploy”.**



```
"contentVersion": "1.0.0.0",
"parameters": {
    "storageAccountName": {
        "type": "string",
        "metadata": {
            "description":
        }
    },
    "containerName": {
        "type": "string",
        "defaultValue": "lc
        "metadata": {
            "description":
        }
    },
    "location": {
        "type": "string",
        "defaultValue": "[r
        "metadata": {
            "description":
        }
    }
```

Deploy to Resource Group

Simplilearnb2c4
Knkillekar@simplilearnb2c4.onmicrosoft.com

Subscription: Microsoft Azure (Knkillekar)

Resource group: armrg (South India)

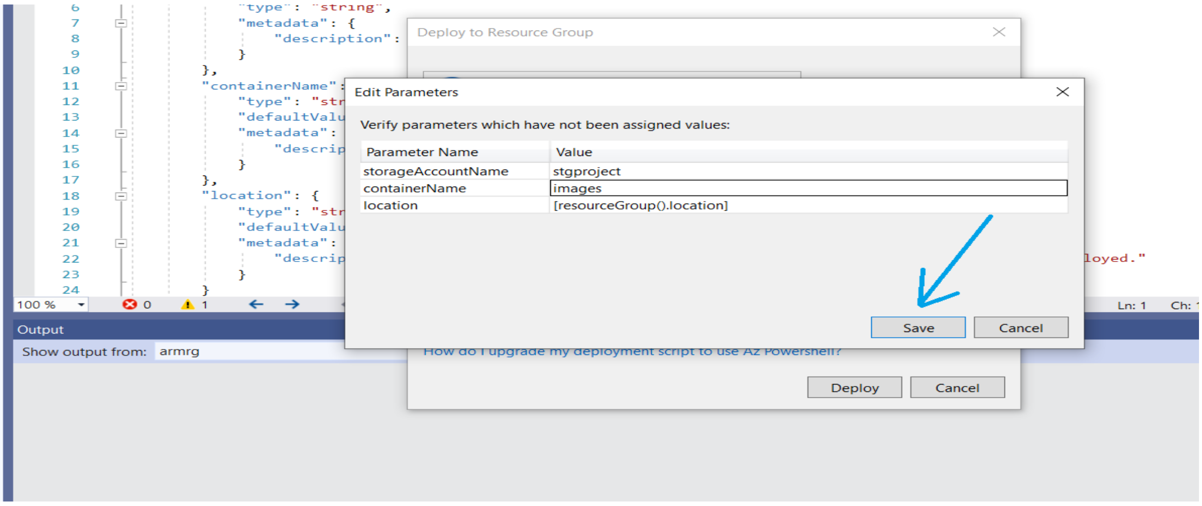
Deployment template: azuredeploy.json

Template parameters file: azuredeploy.parameters.json

Artifact storage account:

How do I upgrade my deployment script to use Az Powershell?

Give the storage account name as **“stgproject”** and container as **“images”** as shown in the window below and click on **“Save”.**

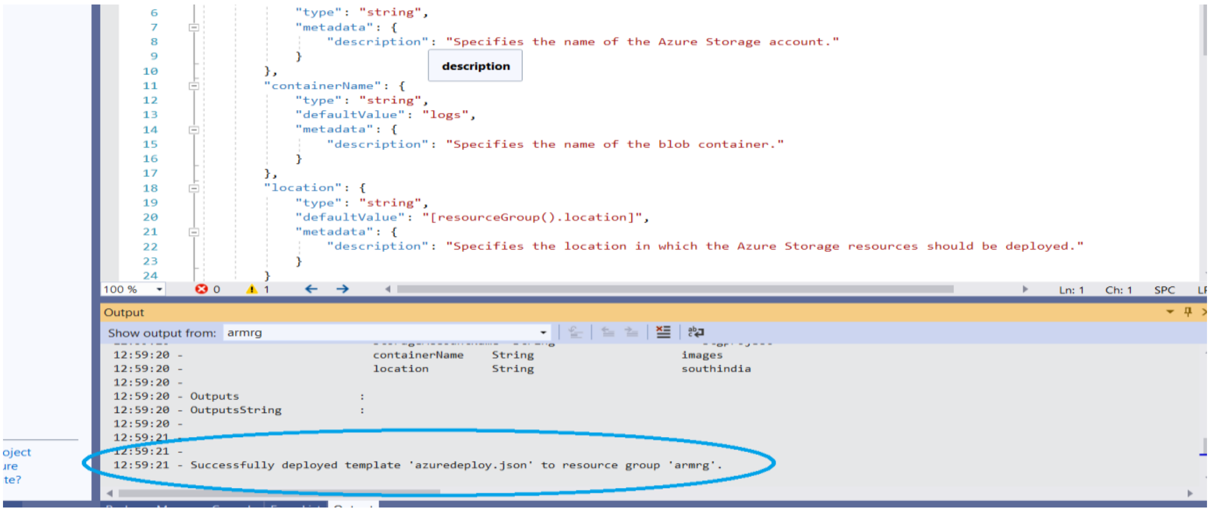


Edit Parameters

Verify parameters which have not been assigned values:

Parameter Name	Value
storageAccountName	stgproject
containerName	images
location	[resourceGroup().location]

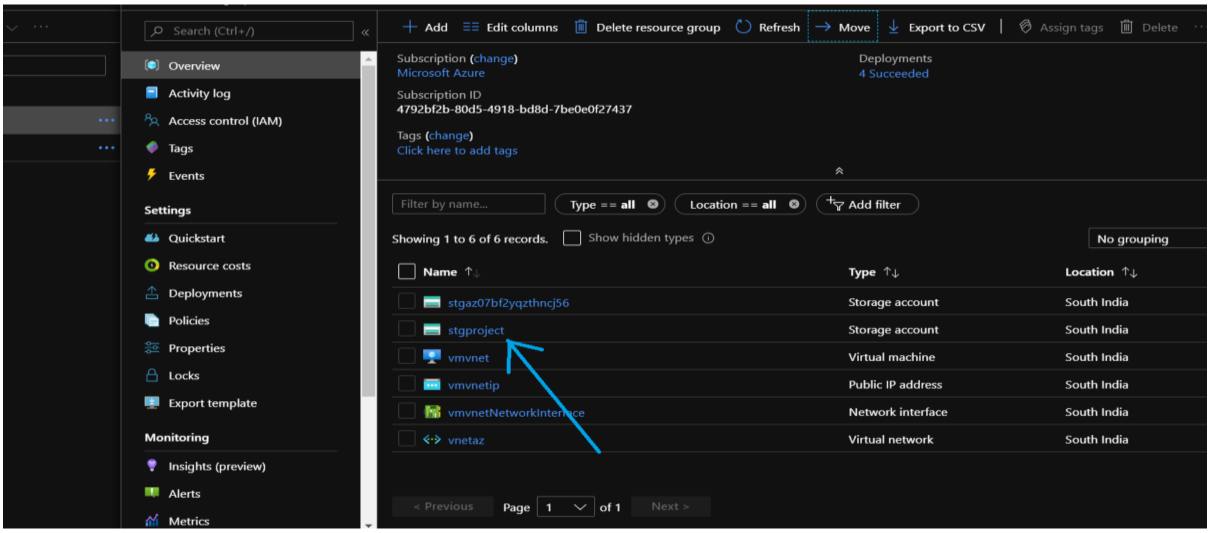
The template is deployed successfully as shown in the window below.



```
        "type": "string",
        "metadata": {
            "description": "Specifies the name of the Azure Storage account."
        }
    },
    "containerName": {
        "type": "string",
        "defaultValue": "logs",
        "metadata": {
            "description": "Specifies the name of the blob container."
        }
    },
    "location": {
        "type": "string",
        "defaultValue": "[resourceGroup().location]",
        "metadata": {
            "description": "Specifies the location in which the Azure Storage resources should be deployed."
        }
    }
```

```
12:59:20 -                    containerName     String              images
12:59:20 -                    location          String              southindia
12:59:20 -
12:59:20 - Outputs                    :
12:59:20 - OutputsString              :
12:59:20 -
12:59:21 -
12:59:21 -
12:59:21 - Successfully deployed template 'azuredeploy.json' to resource group 'armrg'.
```

The same can be verified in the **"Azure Portal"** as shown in the window below, the **"stgproject"** is deployed successfully.



Subscription (change)
Microsoft Azure

Subscription ID
4792bf2b-80d5-4918-bd8d-7be0e0f27437

Tags (change)
Click here to add tags

Deployments
4 Succeeded

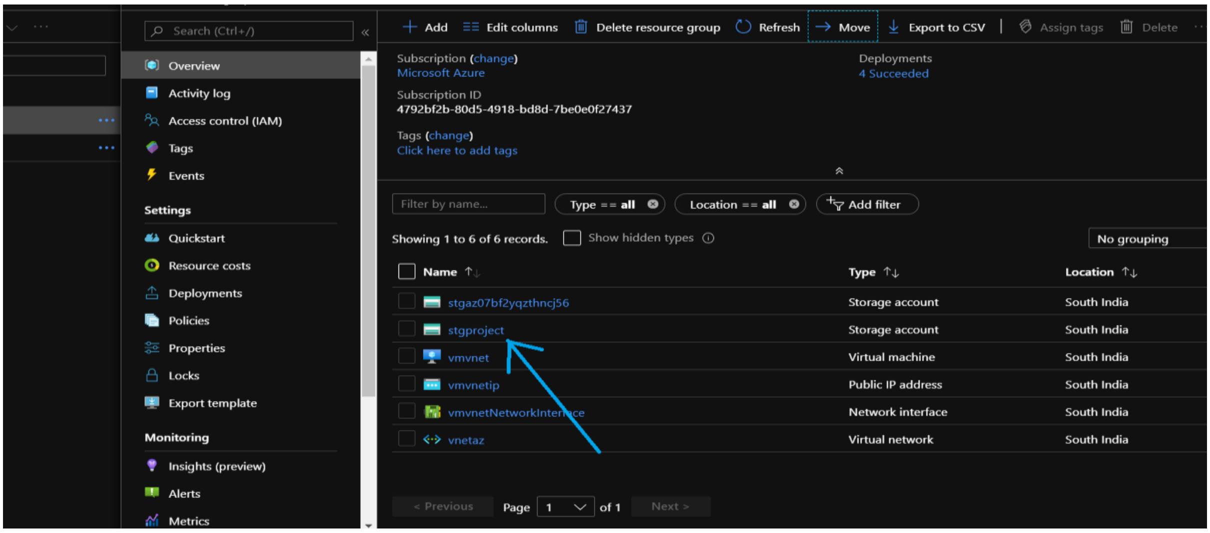
Type == all   Location == all

Showing 1 to 6 of 6 records. □ Show hidden types

□ Name	Type	Location
□ stgaz07bf2yqzthncj56	Storage account	South India
□ stgproject	Storage account	South India
□ vmvnet	Virtual machine	South India
□ vmvnetip	Public IP address	South India
□ vmvnetNetworkInterface	Network interface	South India
□ vnetaz	Virtual network	South India

Page 1 of 1

Step 3: Now to upload file to the container **"images"**, click on **"stgproject"** and then on **"Container"** as shown in the below.



Subscription (change)
Microsoft Azure

Subscription ID
4792bf2b-80d5-4918-bd8d-7be0e0f27437

Tags (change)
Click here to add tags

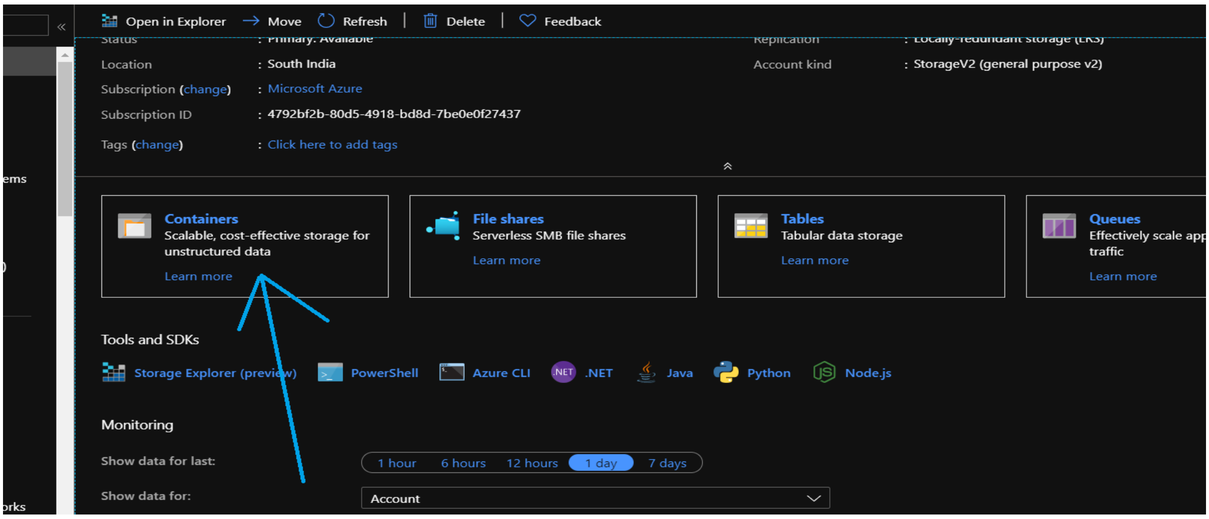
Deployments
4 Succeeded

Type == all   Location == all

Showing 1 to 6 of 6 records. □ Show hidden types

□ Name	Type	Location
□ stgaz07bf2yqzthncj56	Storage account	South India
□ stgproject	Storage account	South India
□ vmvnet	Virtual machine	South India
□ vmvnetip	Public IP address	South India
□ vmvnetNetworkInterface	Network interface	South India
□ vnetaz	Virtual network	South India

Page 1 of 1



Status : Primary: Available

Location : South India

Subscription (change) : Microsoft Azure

Subscription ID : 4792bf2b-80d5-4918-bd8d-7be0e0f27437

Tags (change) : Click here to add tags

Replication : Locally-redundant storage (LRS)

Account kind : StorageV2 (general purpose v2)

Containers
Scalable, cost-effective storage for unstructured data
Learn more

File shares
Serverless SMB file shares
Learn more

Tables
Tabular data storage
Learn more

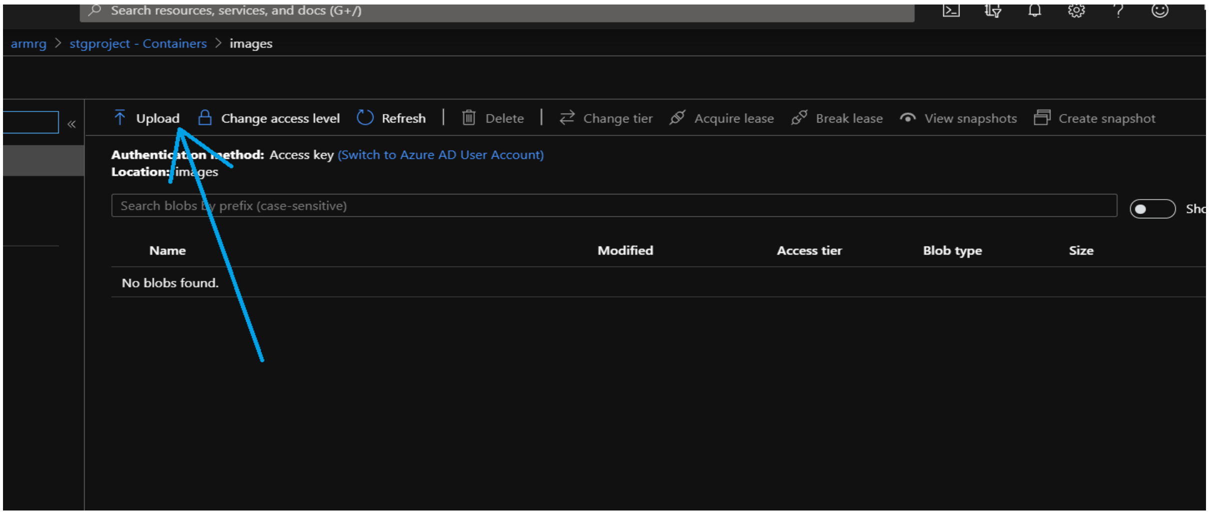
Queues
Effectively scale app traffic
Learn more

Monitoring

Show data for last: 1 hour 6 hours 12 hours 1 day 7 days

Show data for: Account

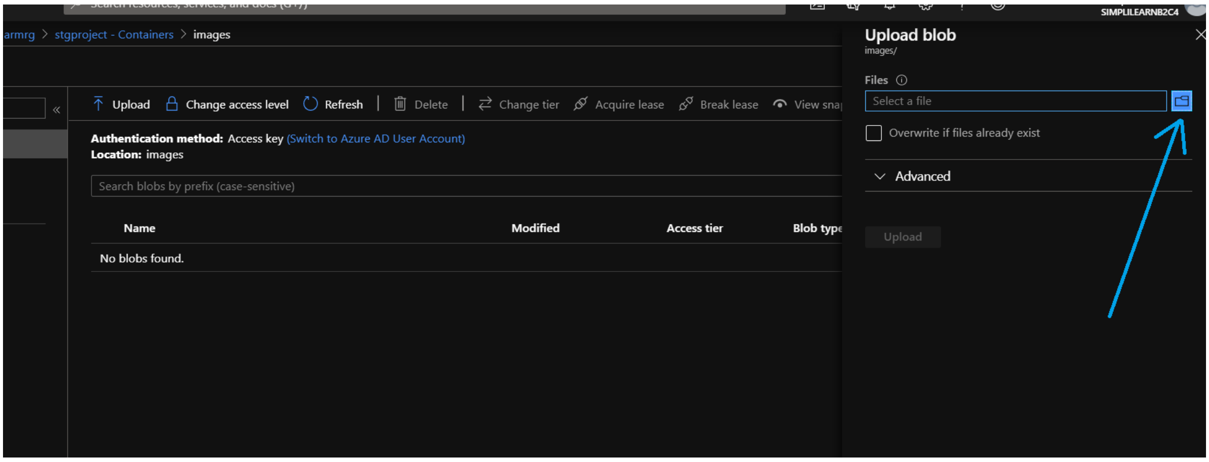
Now click on the container **“images”** and click on **“Upload”** as shown in the window below.



Authentication method: Access key (Switch to Azure AD User Account)
Location: images

No blobs found.

Now choose file to upload as shown in the window below.

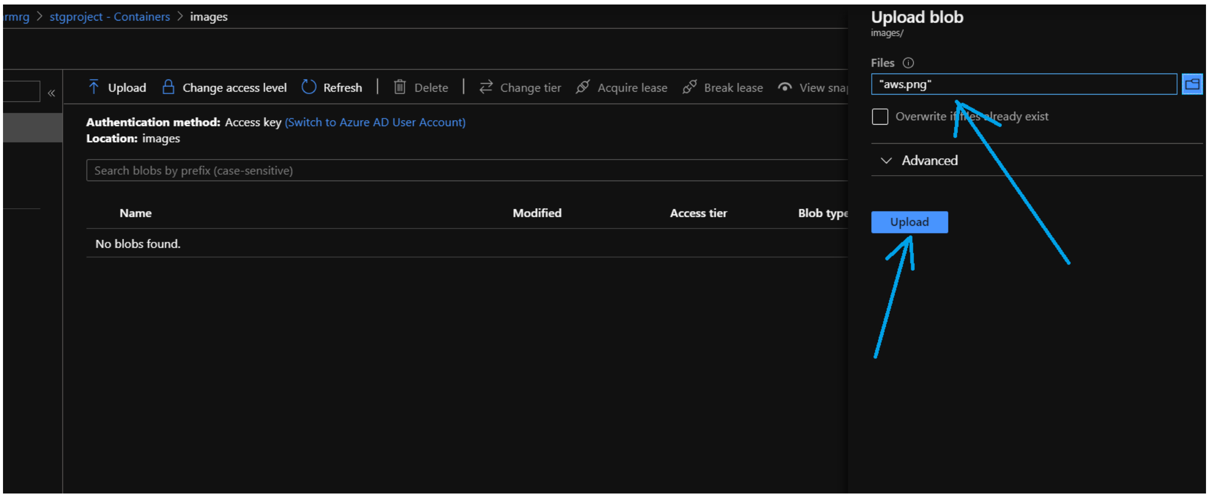


Authentication method: Access key (Switch to Azure AD User Account)
Location: images

No blobs found.

□ Overwrite if files already exist

Next, click on **“Upload”** as shown in the window below.



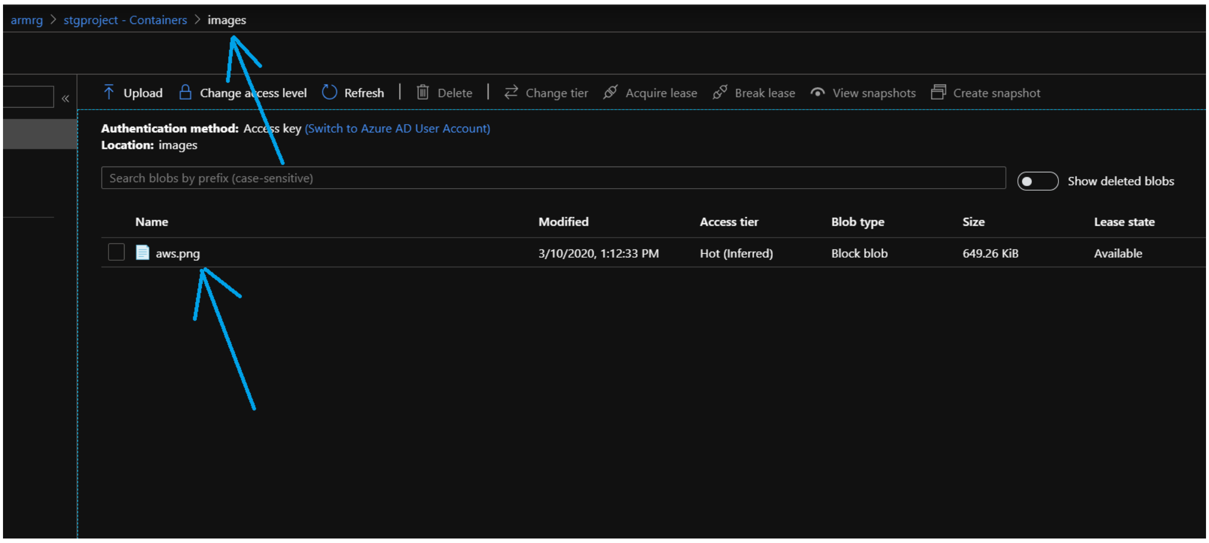
Authentication method: Access key (Switch to Azure AD User Account)
Location: images

No blobs found.

Files "aws.png"

□ Overwrite if files already exist

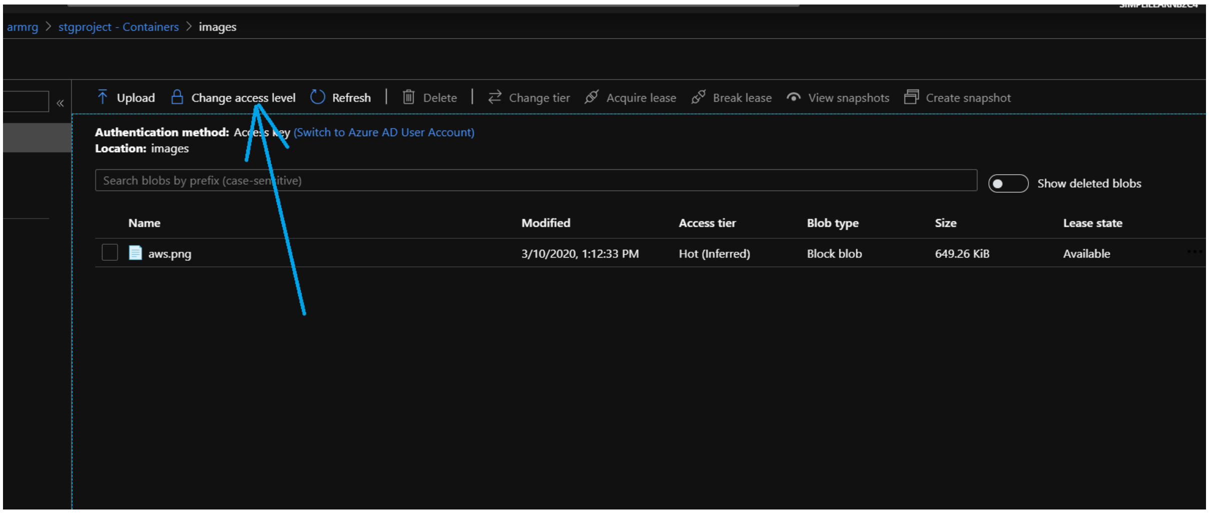
The file **“aws.png”** is successfully uploaded to the container **“images”** as shown in the window below.



Authentication method: Access key (Switch to Azure AD User Account)
Location: images

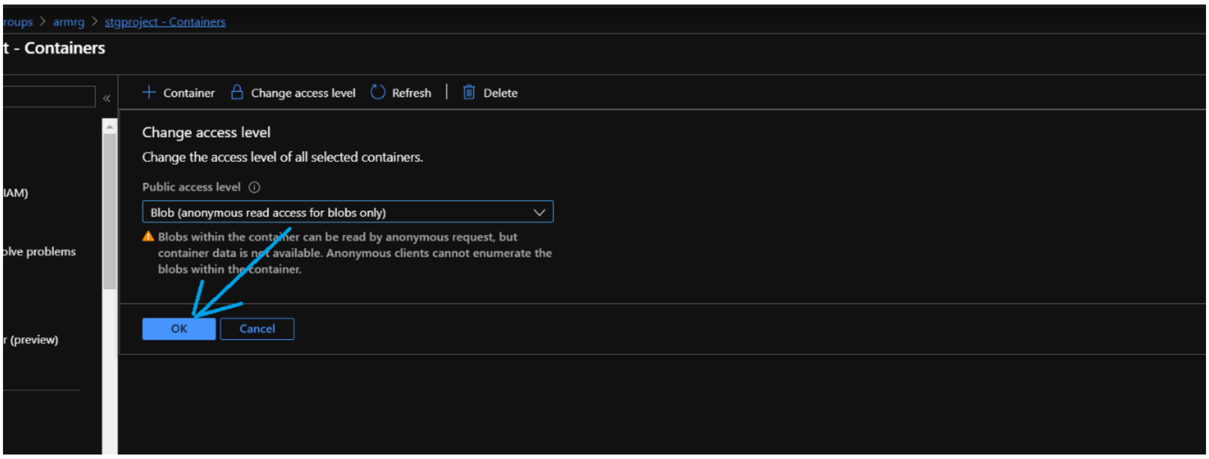
Name	Modified	Access tier	Blob type	Size	Lease state
aws.png	3/10/2020, 1:12:33 PM	Hot (Inferred)	Block blob	649.26 KiB	Available

Now click on **“Change access level”** , select public access level to **“Blob (anonymous read access for blob only)** and click on **“ok”** as shown in the window below.



Authentication method: Access key (Switch to Azure AD User Account)
Location: images

Name	Modified	Access tier	Blob type	Size	Lease state
aws.png	3/10/2020, 1:12:33 PM	Hot (Inferred)	Block blob	649.26 KiB	Available



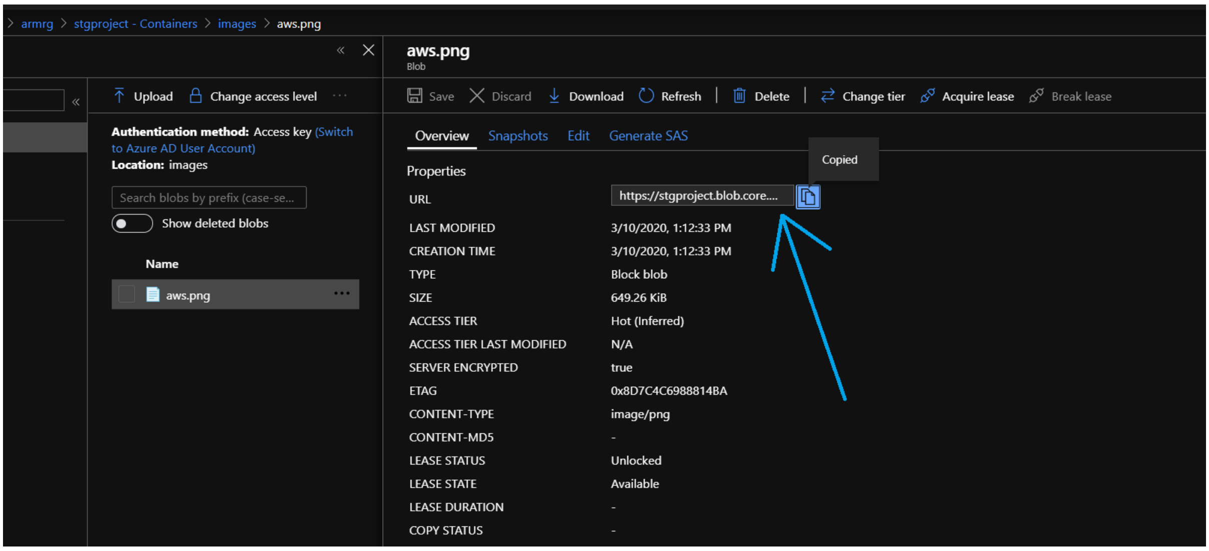
Change access level

Change the access level of all selected containers.

Public access level: Blob (anonymous read access for blobs only)

Blobs within the container can be read by anonymous request, but container data is not available. Anonymous clients cannot enumerate the blobs within the container.

Now click on container **“images”** and click on the file **“aws.png”** .Copy the url as shown in the window below.



armrg > stgproject - Containers > images > aws.png

aws.png
Blob

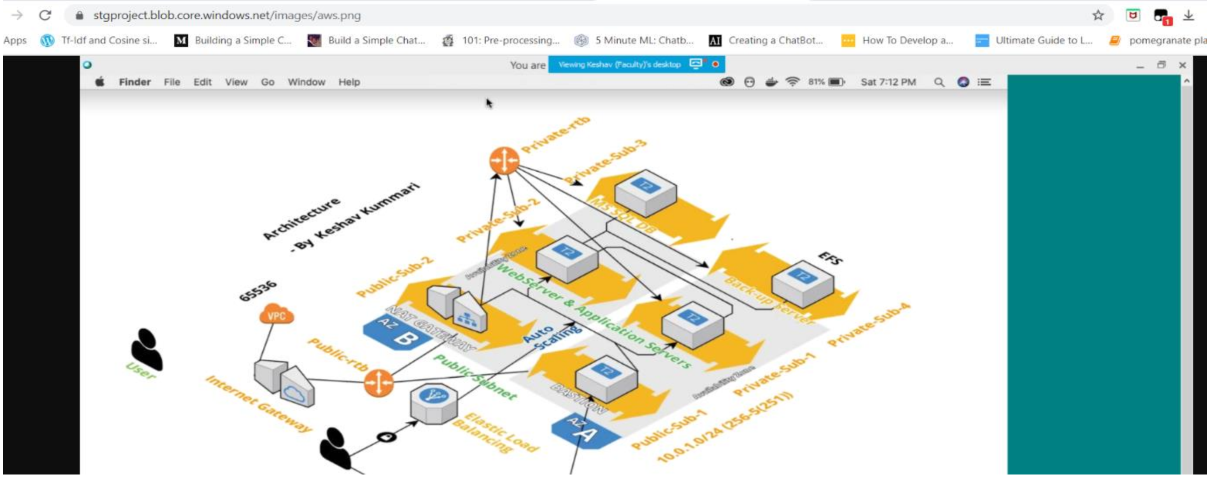
Authentication method: Access key (Switch to Azure AD User Account)
Location: images

aws.png

Properties

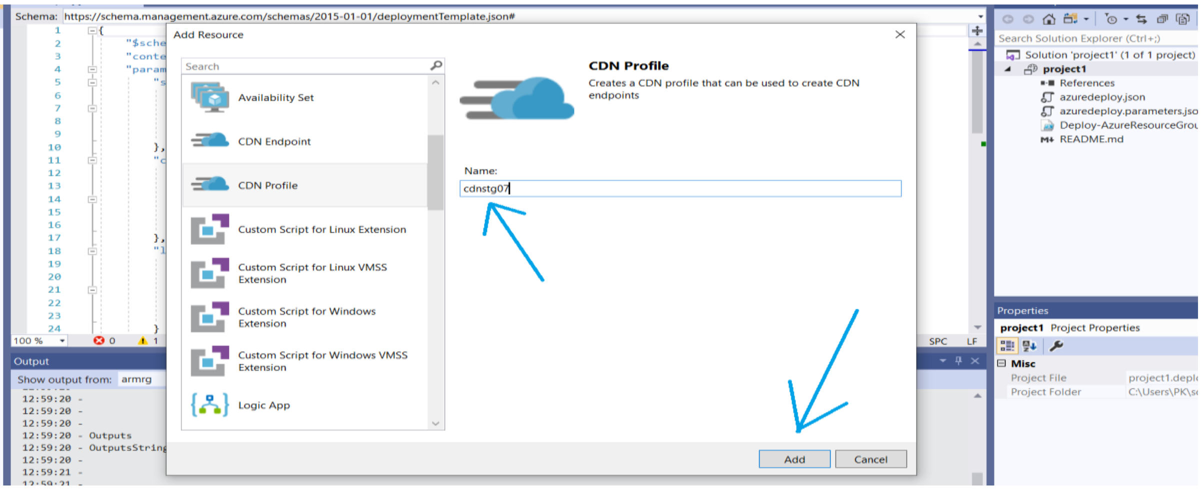
URL	https://stgproject.blob.core....
LAST MODIFIED	3/10/2020, 1:12:33 PM
CREATION TIME	3/10/2020, 1:12:33 PM
TYPE	Block blob
SIZE	649.26 KiB
ACCESS TIER	Hot (Inferred)
ACCESS TIER LAST MODIFIED	N/A
SERVER ENCRYPTED	true
ETAG	0x8D7C4C6988814BA
CONTENT-TYPE	image/png
CONTENT-MD5	-
LEASE STATUS	Unlocked
LEASE STATE	Available
LEASE DURATION	-
COPY STATUS	-

Now, paste the url in the browser and see the output image as shown in the window below.



stgproject.blob.core.windows.net/images/aws.png

Step 4: Now to implement **cdn**, add resource as **“CDN Profile”** and give it a name as **“cdnstg07”** as shown in the window below and click on **“Add”**.



Schema: https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#

Add Resource

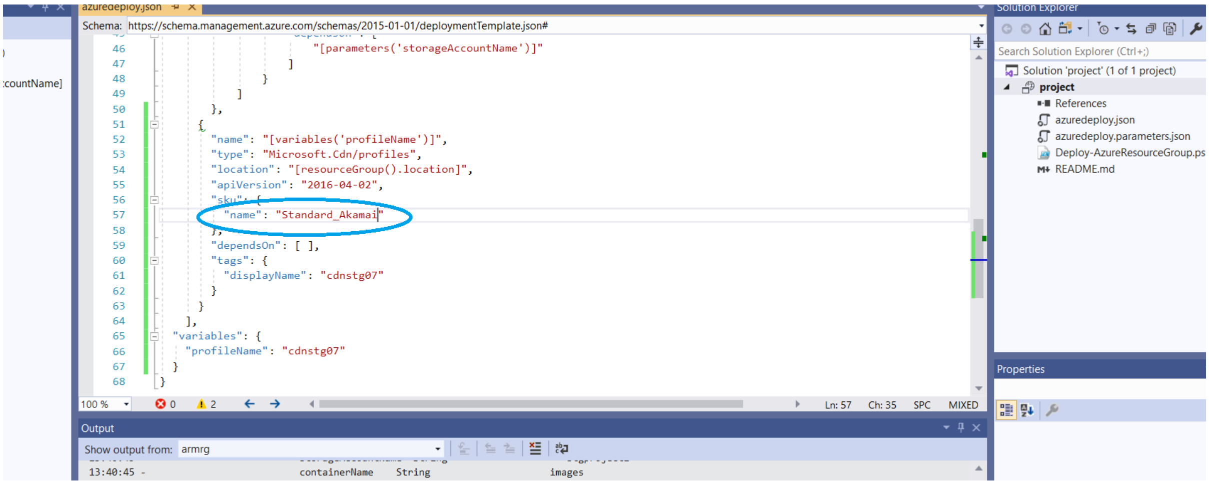
Search
Availability Set
CDN Endpoint
CDN Profile
Custom Script for Linux Extension
Custom Script for Linux VMSS Extension
Custom Script for Windows Extension
Custom Script for Windows VMSS Extension
Logic App
Add
Cancel
100 %
Output
Show output from: armrg
Search Solution Explorer (Ctrl+;)
Solution 'project1' (1 of 1 project)
project1
References
azuredeploy.json
azuredeploy.parameters.jso
Deploy-AzureResourceGrou
README.md
Properties
project1 Project Properties
Misc
Project File project1.depl
Project Folder C:\Users\PK\s
SPC
LF

CDN Profile
Creates a CDN profile that can be used to create CDN endpoints

Name: cdnstg07

```
12:59:20 -
12:59:20 -
12:59:20 -
12:59:20 - Outputs
12:59:20 - OutputsString
12:59:20 -
12:59:21 -
12:59:21 -
```

Now change the sku to **“Standard_Akamai”** as shown in the window.


azuredeploy.json
Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'project' (1 of 1 project)
project
References
azuredeploy.json
azuredeploy.parameters.json
Deploy-AzureResourceGroup.ps
README.md
Properties
100 %
0
2
Ln: 57
Ch: 35
SPC
MIXED
Output

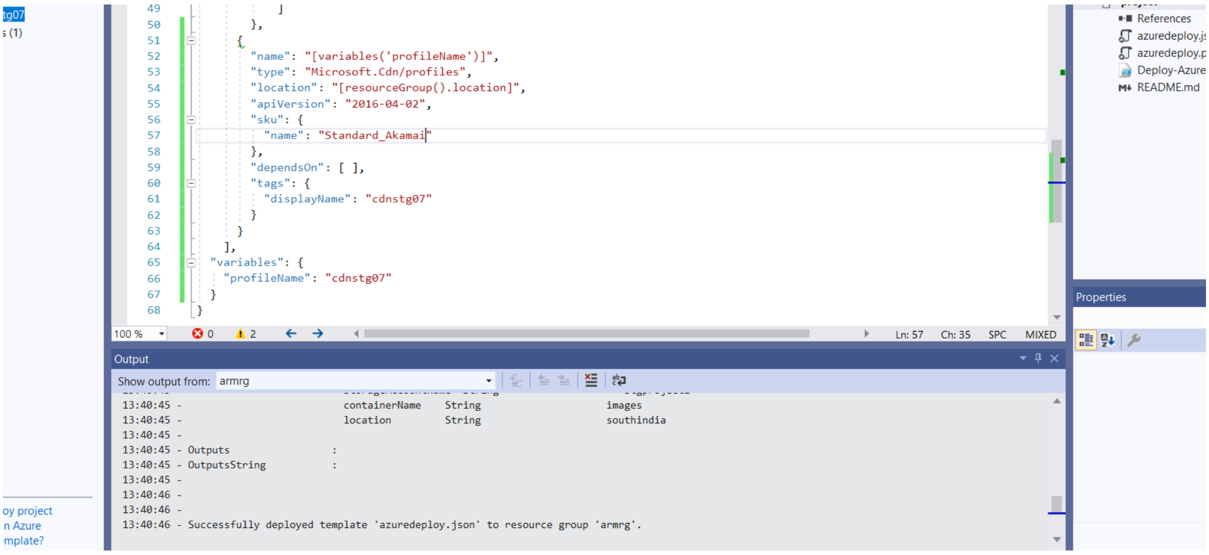
Schema: https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#

```
              "[parameters('storageAccountName')]"
            ]
          }
        ]
      },
      {
        "name": "[variables('profileName')]",
        "type": "Microsoft.Cdn/profiles",
        "location": "[resourceGroup().location]",
        "apiVersion": "2016-04-02",
        "sku": {
          "name": "Standard_Akamai"
        },
        "dependsOn": [ ],
        "tags": {
          "displayName": "cdnstg07"
        }
      }
    ],
    "variables": {
      "profileName": "cdnstg07"
    }
}
```

Show output from: armrg

```
13:40:45 -                    containerName    String                 images
```

Now click on **“Deploy”** and the template is deployed successfully as shown in the window.

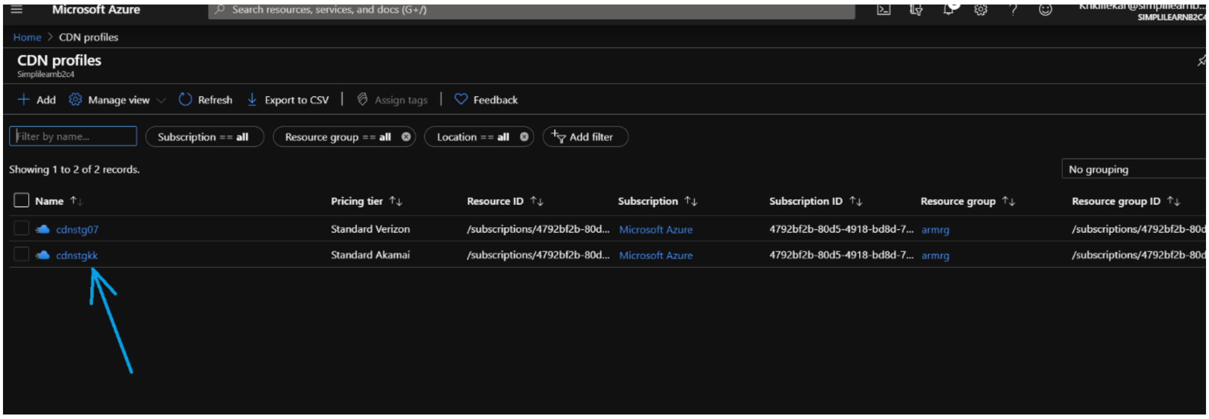


```
        ]
      },
      {
        "name": "[variables('profileName')]",
        "type": "Microsoft.Cdn/profiles",
        "location": "[resourceGroup().location]",
        "apiVersion": "2016-04-02",
        "sku": {
          "name": "Standard_Akamai"
        },
        "dependsOn": [ ],
        "tags": {
          "displayName": "cdnstg07"
        }
      }
    ],
    "variables": {
      "profileName": "cdnstg07"
    }
}
```

Show output from: armrg

```
13:40:45 -                    containerName    String                 images
13:40:45 -                    location         String                 southindia
13:40:45 -
13:40:45 - Outputs                 :
13:40:45 - OutputsString           :
13:40:45 -
13:40:46 -
13:40:46 -
13:40:46 - Successfully deployed template 'azuredeploy.json' to resource group 'armrg'.
```

Step 5: Now go to **“Azure Portal”** and click on **“CDN profiles”.** As we can see **“cdnstgkk”** is created successfully as shown in the window below.

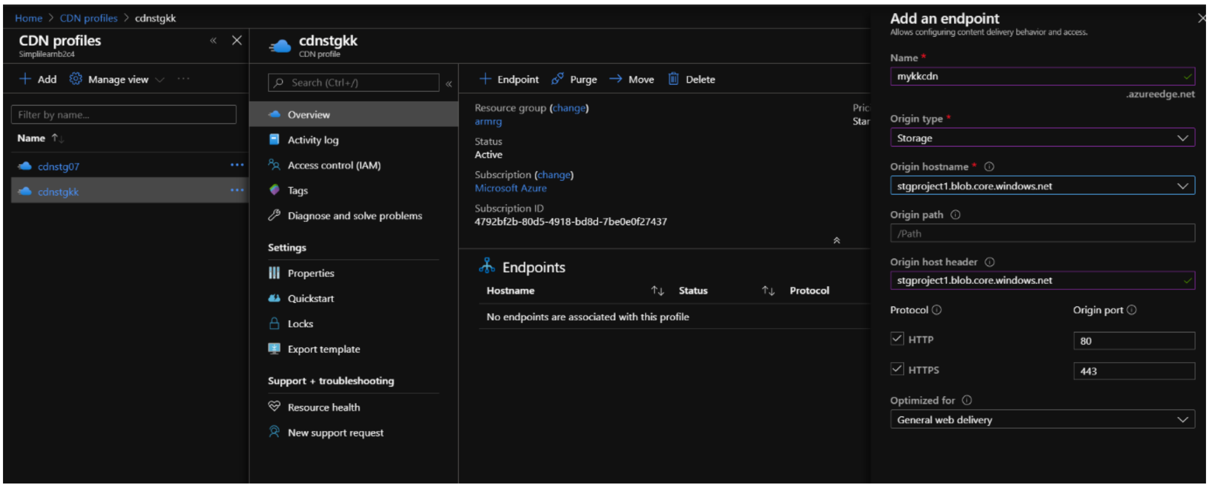


CDN profiles
Simplilearnb2c4

Showing 1 to 2 of 2 records.

Name	Pricing tier	Resource ID	Subscription	Subscription ID	Resource group	Resource group ID
cdnstg07	Standard Verizon	/subscriptions/4792bf2b-80d...	Microsoft Azure	4792bf2b-80d5-4918-bd8d-7...	armrg	/subscriptions/4792bf2b-80d
cdnstgkk	Standard Akamai	/subscriptions/4792bf2b-80d...	Microsoft Azure	4792bf2b-80d5-4918-bd8d-7...	armrg	/subscriptions/4792bf2b-80d

Now click on **"cdnstgkk"** and then click on **"Endpoint"**, enter the following details as shown in the window below and click on **"Add".**



Resource group (change)
armrg

Status
Active

Subscription (change)
Microsoft Azure

Subscription ID
4792bf2b-80d5-4918-bd8d-7be0e0f27437

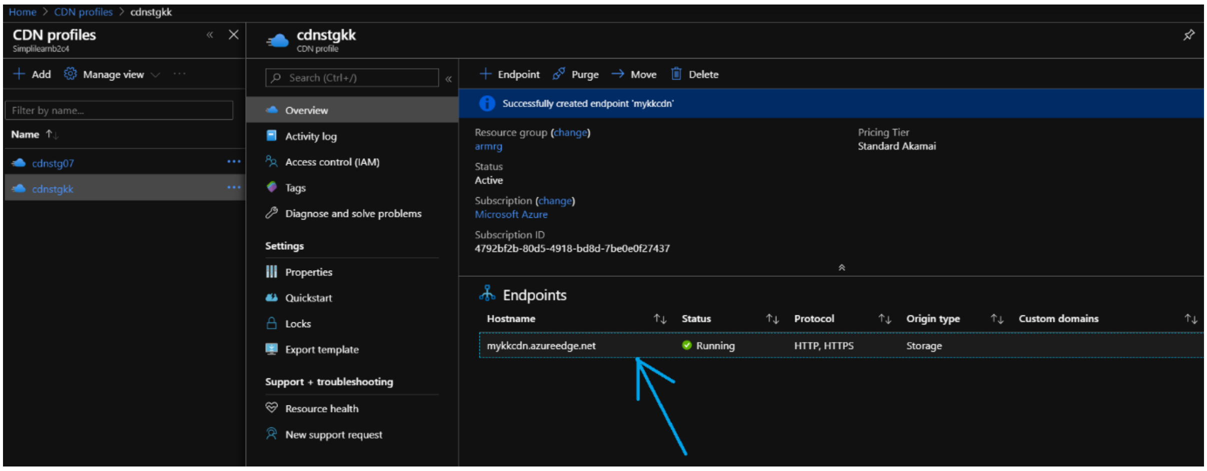
Endpoints

Hostname	Status	Protocol
No endpoints are associated with this profile		

Add an endpoint
Allows configuring content delivery behavior and access.

Name *: mykkcdn .azureedge.net
Origin type *: Storage
Origin hostname *: stgproject1.blob.core.windows.net
Origin path: /Path
Origin host header: stgproject1.blob.core.windows.net
Protocol: ☑ HTTP ☑ HTTPS
Origin port: 80, 443
Optimized for: General web delivery

The endpoint https://mykkcdn.azureedge.net is created successfully as shown in the window below.



Successfully created endpoint 'mykkcdn'

Resource group (change)
armrg

Pricing Tier
Standard Akamai

Status
Active

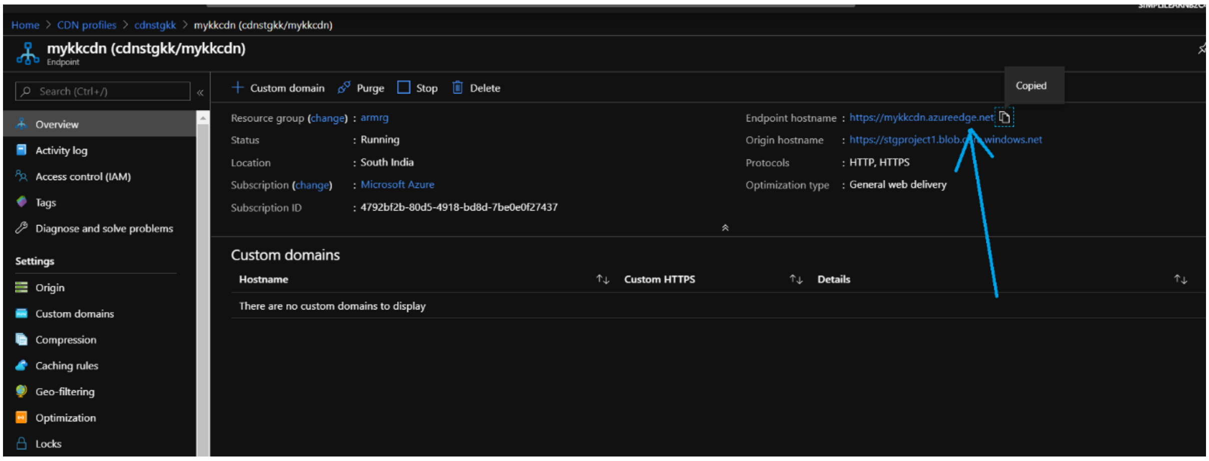
Subscription (change)
Microsoft Azure

Subscription ID
4792bf2b-80d5-4918-bd8d-7be0e0f27437

Endpoints

Hostname	Status	Protocol	Origin type	Custom domains
mykkcdn.azureedge.net	Running	HTTP, HTTPS	Storage	

Next, click on the endpoint "https://mykkcdn.azureedge.net" and copy the endpoint hostname as shown in the window below.



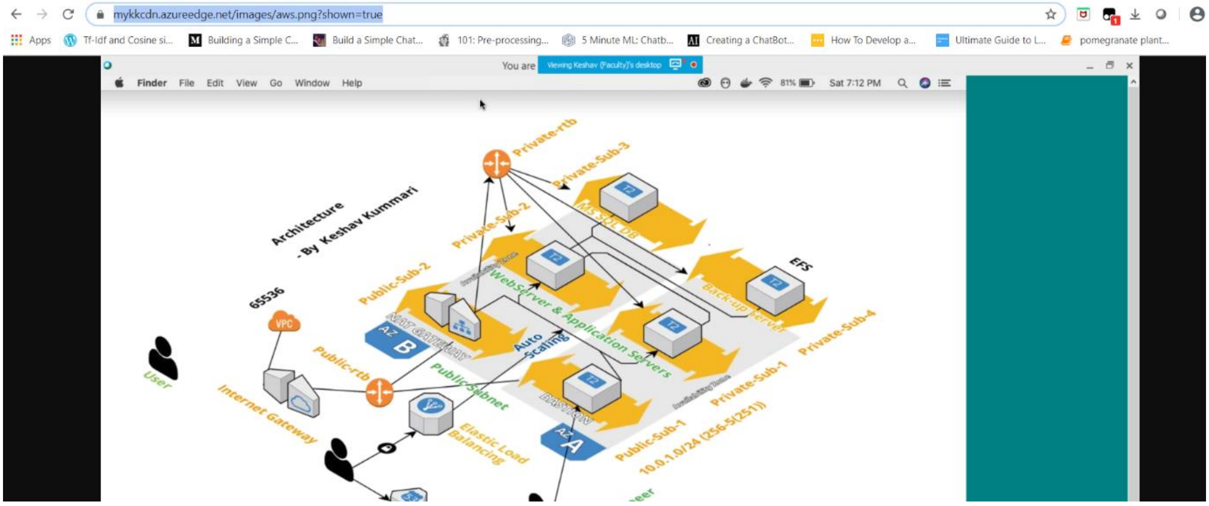
Resource group (change) : armrg
Status : Running
Location : South India
Subscription (change) : Microsoft Azure
Subscription ID : 4792bf2b-80d5-4918-bd8d-7be0e0f27437

Endpoint hostname : https://mykkcdn.azureedge.net
Origin hostname : https://stgproject1.blob.core.windows.net
Protocols : HTTP, HTTPS
Optimization type : General web delivery

Custom domains

Hostname	Custom HTTPS	Details
There are no custom domains to display		

Now enter the url "https://mykkcdn.azureedge.net/images/aws.png" in the browser and check the output as shown in the window below.



****************************END***