

# Understanding Cloud Architecture

Session 6

# Learning Objectives

- Describe the Windows Azure platform architecture
- Describe the components of the Windows Azure architecture
- Explain the process to create, test, and deploy an Azure cloud service using Visual Studio
- Explain the process to create, test, and deploy a Web application on the cloud

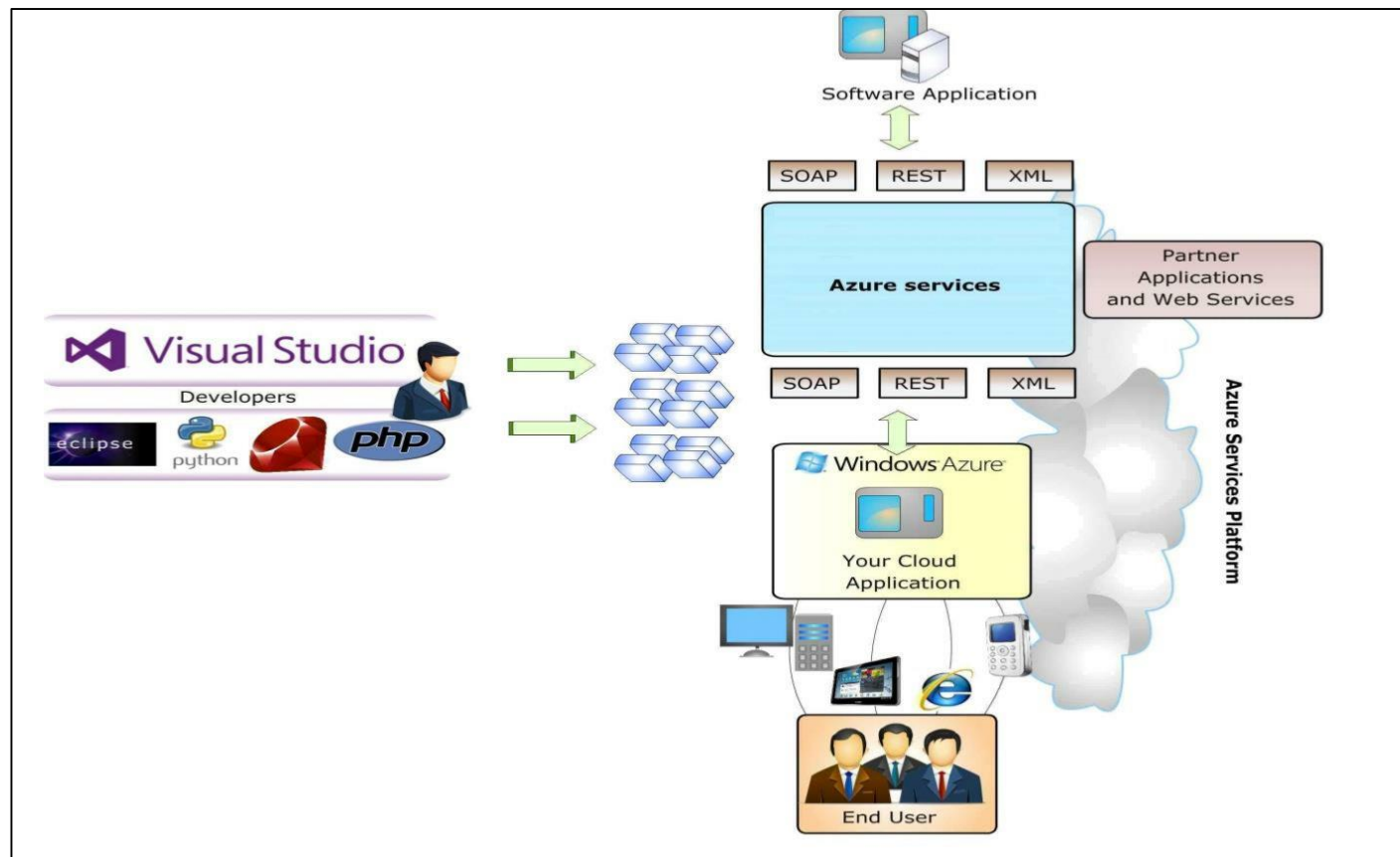
# Window Azure Architecture 1-5

- ❑ The Windows Azure Services Platform has Windows Azure as its cloud OS.
- ❑ The OS:
  - Facilitates business and consumer applications to be built, deployed, and hosted in the cloud.
  - Provides a platform to developers to address both business problems as well as customer needs.
  - Allows deployment of Web based applications both in-premise and/or off-premise applications.
  - Utilizes the organizations in-house services and serves as a runtime environment for applications.



# Window Azure Architecture 2-5

- ❏ The following figure shows the Windows Azure PaaS cloud architecture:



# Windows Azure Architecture 3-5

- ❑ The hosted services of Windows Azure OS can be categorized into one of the following roles:

## Web role

- This role instance accepts all incoming HTTP or HTTPS requests. It also supports web applications that are hosted in Internet Information Services(IIS).



## Worker role

- This role instance is similar to Web role, but it cannot support Web applications hosted in IIS.



## Virtual Machine (VM) role

- This role instance allows creation and running of code in the Windows Azure OS.



# Window Azure Architecture 4-5

## ☐ Applications hosted in the Windows Azure OS:

- Can be of one of the mentioned roles or a combination of these roles.
- Can be used to activate multiple virtual machines with multiple role instances.



## ☐ Windows Azure OS:

- ☐ Simplifies IT management and optimizes the up-front and ongoing expenses.
- Supports Microsoft based standards and protocols and also third-party standards, programming languages, and several other platforms.

For example, HTTP, HTTPS, XML, SOAP, Representational State Transfer (REST), Ruby, PHP, Python, and Eclipse.

# Window Azure Architecture 5-5



- ❑ Windows Azure Services Platform:
  - Has Windows Azure as its cloud OS.
  - Allows deployment of Web based applications both in-premise and/or off-premise applications.
  - Utilizes the organizations in-house services and serves as a runtime environment for applications.
  - Includes a mix of application, virtualization service layers, storage, security, and a desktop development environment.
  - Handles the load balancing, caching, redundancy, resource management, and life cycles of hosted services.

# Compute Service

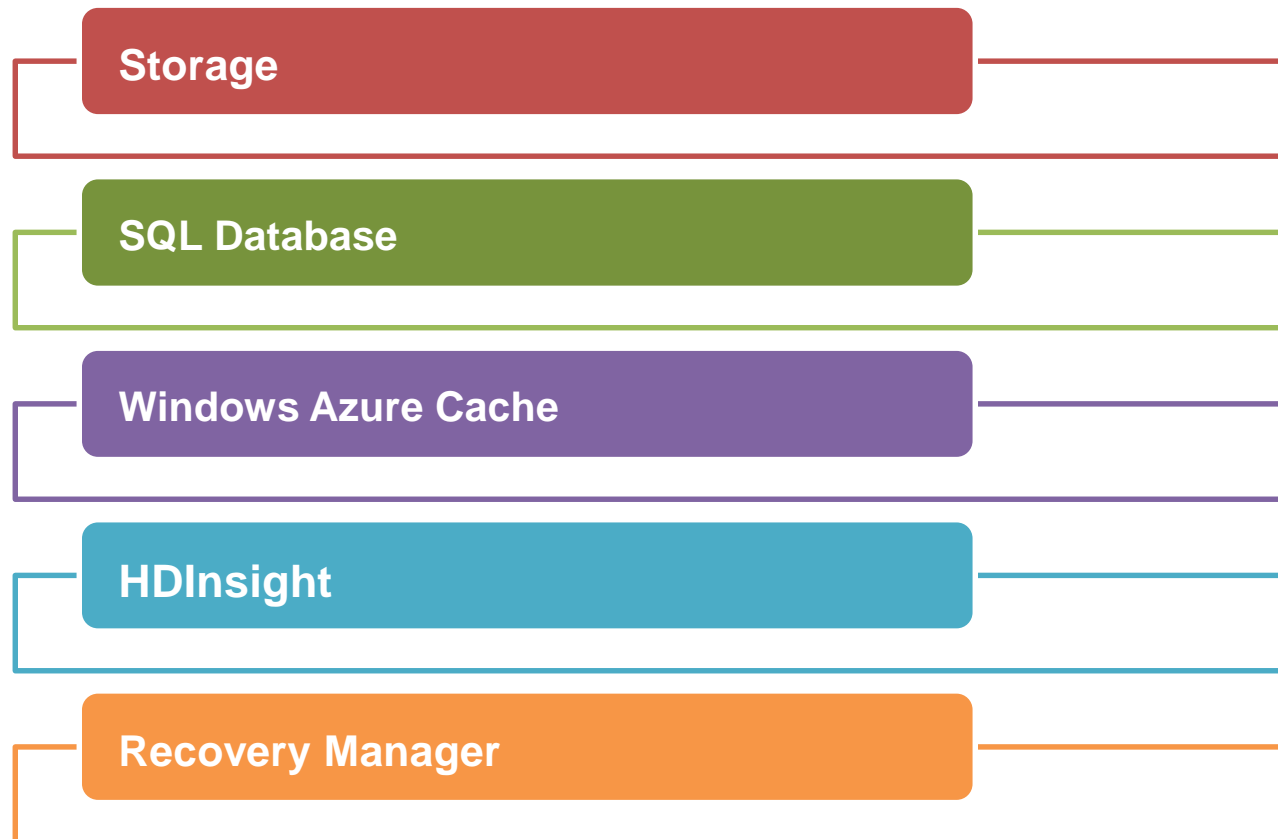
**Compute** is one of the core components of Windows Azure. Some of the commonly used compute services in Azure include:

1. **Virtual Machines (VMs)**: Azure Virtual Machines allow you to create and manage virtual machines in the cloud. You can choose from a wide range of pre-configured VM sizes and operating system images to run your applications.
2. **Azure App Service**: Azure App Service is a fully managed platform for hosting web applications, mobile app backends, and RESTful APIs. It supports various programming languages and frameworks like .NET, Java, Node.js, Python, and more.
3. **Azure Functions**: Azure Functions is a serverless compute service that allows you to run event-driven code snippets or functions in the cloud. You can write functions in different languages and trigger them based on various events like HTTP requests, timers, storage events, and more.
4. **Azure Container Instances (ACI)**: Azure Container Instances provide a way to run containers without managing the underlying infrastructure. It allows you to quickly deploy and run individual containers or groups of containers using Docker images.
5. **Azure Kubernetes Service (AKS)**: Azure Kubernetes Service is a managed Kubernetes offering that simplifies the deployment, management, and scaling of containerized applications. It provides a fully managed Kubernetes cluster where you can deploy and orchestrate your containers.
6. **Azure Batch**: Azure Batch is a cloud-based job scheduling service that enables you to run large-scale parallel and high-performance computing (HPC) workloads. It's designed for scenarios that require running tasks across multiple VMs and requires scale-out capabilities.



# Data Services 1-5

- Windows Azure provides five main data services:



# Data Services 2-5

## Storage

- ❑ Windows Azure Backup Storage services:
  - Allows the developers to store different kinds of data, such as unstructured, non-relational, and binary data.
  - Uses blobs for storing such data and queues for storing message information for the clients.



## SQL Database

- ❑ Windows Azure Backup Storage services:
  - Allows developers to either use the SQL Database on the cloud or share data between the SQL Database on the cloud and locally installed instances. These databases can then be used by the applications that are developed by the developers.



# Data Services 3-5

## Windows Azure Cache

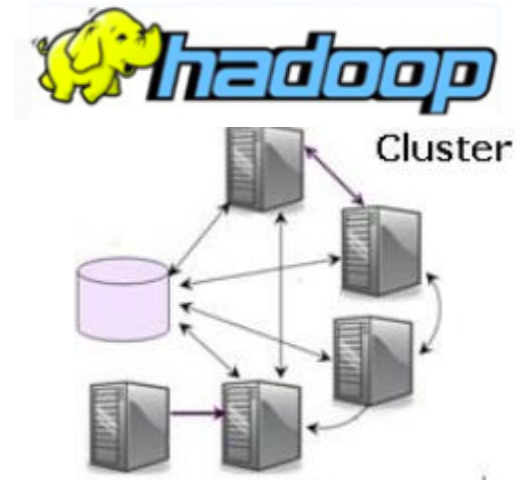


- Azure Cache for Redis is a fully managed, in-memory caching service that is compatible with Redis. It provides high-performance caching capabilities, data persistence, and support for various Redis features.
- Azure Cache for Redis offers features such as data partitioning, replication, automatic failover, and support for popular Redis data structures and commands.
- With Azure Cache for Redis, you can improve the performance and scalability of your applications by reducing data access latency and offloading the load on your backend systems. It is commonly used for scenarios like session caching, data caching, and real-time data processing.

# Data Services 4-5

## HDInsight

- ❑ The Windows Azure HDInsight Services:
  - Uses Apache Hadoop for drawing analytics from the unstructured data.
  - Allows the developers to build an appropriately sized Hadoop clusters as and when required for analysis of large amounts of unstructured data.
- ❑ The Windows Azure PowerShell Service:
  - Helps HDInsight Service to configure, run, and post-process Hadoop jobs. To enable these jobs, HDInsight Service uses .NET library classes.



# Data Services 5-5

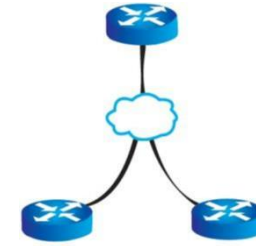
## Recovery Manager



- ❑ The Windows Azure Services:
  - Allows you to protect data.
  - Ensures that your clouds are protected.
  - Uses services such as Hyper-V Recover Manager to automate the recovery.
  - Provides backup services to automate the backups.

# Networking 1-4

- ❑ Datacenters are geographically spread out and they run Windows Azure cloud OS to manage and store applications and data.
  - You can use any datacenter to run your applications or to store your data. These datacenters can be connected in two ways:



## Windows Azure Virtual Private Network (VPN)

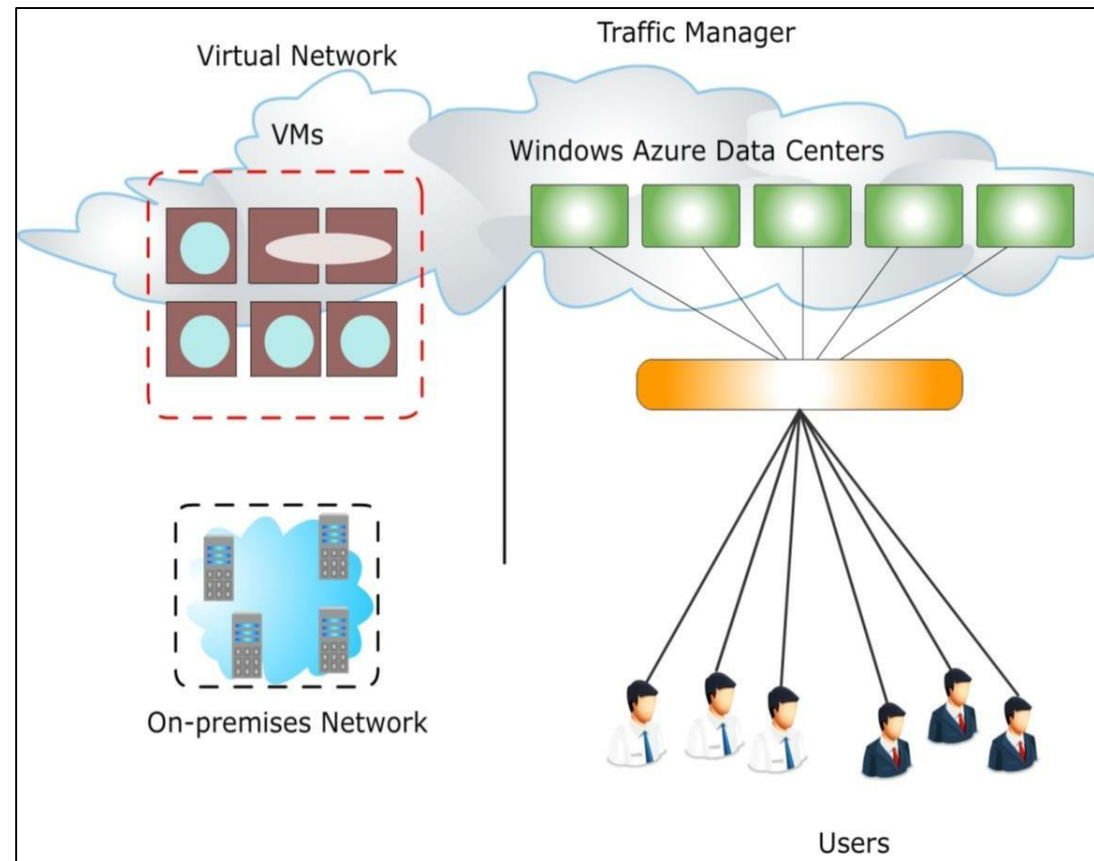
- You can use this network connection to connect an on-premises local network to a pre-defined set of Windows Azure VMs.

## Windows Azure Traffic Manager

- When applications are running at more than one datacenters in a Windows Azure environment. This will help you to route requests from one instance to another.

# Networking 2-4

- ❑ The following figure shows Cloud VPN and multiple datacenters for data routing:



# Networking 3-4

## Windows Azure Virtual Private Network (VPN)



Windows Azure Virtual Network helps in:



- Extending your network to the Windows Azure environment as part of your on-premises network.
- Interacting with the datacenter as though it is present in your own premises.
- Setting up the VPN between the local machine and the groups of VMs in Windows Azure environment.



# Networking 4-4

## Windows Azure Traffic Manager

### ❏ Windows Azure Traffic Manager:

- Directs the user request automatically to the nearest or another datacenter.
- Instructs the Traffic Manager to perform this task, an application owner needs to define rules of how the user request would be directed to the datacenters.



# App Services 1-7

## ❑ Service Bus Queues:

- Support messaging services.
- Use the queue to exchange the messages.

## ❑ Working of the queue:

- In an asynchronous manner, the message producer, which is a sender, sends the message to a queue and continues its processing.
- On the other side, the receiver or the message consumer pulls the message from the queue and gets back to its processing.
- Fundamentally, a queue has First In First Out (FIFO) mechanism of delivering the messages.
- To use the Service Bus Queue of Windows Azure, you need to create a service namespace.

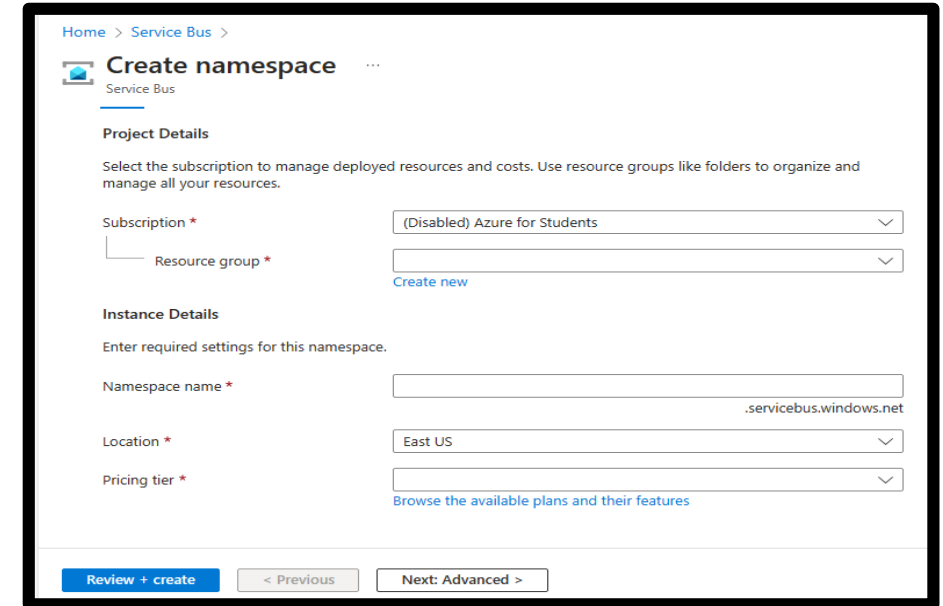
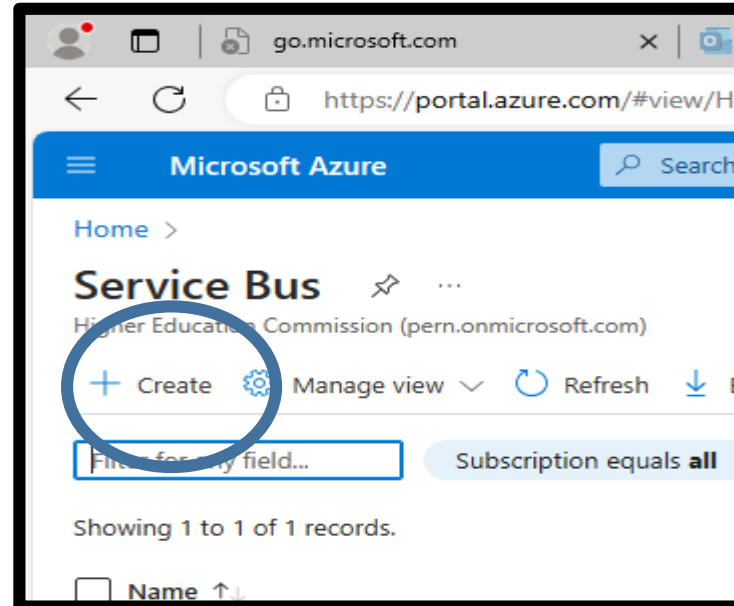
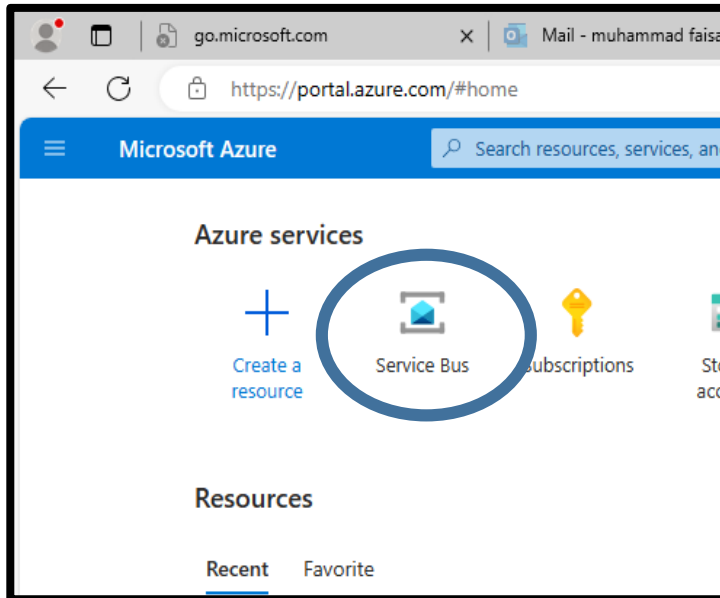


# App Services 2-7

❏ Steps to create a service namespace are:

- Step 1
  - Using your credentials, login to <https://azure.microsoft.com/>
- Step 2
  - Click **Service Bus**.
- Step 3
  - Click **Create**.
- Step 4
  - Enter the name of the namespace, in the **Add a new namespace** section. The system then checks if the name is available.
- Step 5
  - On confirmation, select the country where the namespace needs to be hosted (important to remember it should not be different from the country/region in which your computing resources resides).
- Step 6
  - Click the check mark. The system will create your service namespace and will enable it after you click the check mark.

# App Services 3-7



# App Services 4-7

- ❑ Steps to retrieve the credentials for the namespace in order to perform the management operations are:

Step 1

- Click the **Service Bus** node. This will display the list of available namespaces.

Step 2

- From the populated list, select the namespace that was just created.

Step 3

- Click **Connection Information**.

Step 4

- In the **Access connection information** dialog box, look for the **Default Issuer** and **Default Key** entries and store them for future use.

# App Services 5-7

## ❑ Creating a queue:

- You can use the `NamespaceManager` class to create and manage queues. The `NamespaceManager` is used for managing the namespace. You can use it to create a queue.
- For example, the following code can be used to create a queue:

• `uri` is the URI that represents the Azure Service Bus namespace you want to interact with. It specifies the address of the Service Bus namespace.

• `tokenProvider` is an instance of a class that implements the `ITokenProvider` interface. It provides the security token required for authentication and authorization to access the Service Bus namespace.

```
NamespaceManager namespaceManager = new
    NamespaceManager(uri, tokenProvider);

namespaceManager.CreateQueue("DataCollectionQueue");
```

Microsoft Azure

Search resources, services, and docs (G+/)

mfaisal.bukc@bahria.ed...  
HIGHER EDUCATION COMMISSL...

Home > Service Bus >

Service Bus

Higher Education Commission (pern.onmicrosoft.c...

+ Create

Manage view

Filter for any field...

Name

MyNameSpace75

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

Shared access policies

Geo-Recovery

Encryption

Configuration

Properties

Locks

Entities

Queues

Monitoring

Insights (Preview)

Alerts

Metrics

Diagnostic settings

MyNameSpace75

Service Bus Namespace

+ Queue

+ Topic

Refresh

Delete

Give feedback

Essentials

Resource group (move) : [FirstMLDemo](#)

Status : Not Active

Location : East US

Subscription (move) : [Azure for Students](#)

Subscription ID : e9ffe025-f63a-4456-8636-ba4e0bd4a436

Tags (edit) : [Add tags](#)

Created

Updated

Pricing tier

Local Authentication

Show data for the last: 1 hour 6 hours 12 hours 1 day 7 days 30 days

Requests

100

90

80

70

60

50

40

30

20

10

0

4 PM

4:15 PM

4:30 PM

UTC+05:00

Incoming Requests (Sum), mynamespace75 | 0

Successful Requests (Sum), mynamespace75 | 0

Server Errors. (Sum), mynamespace75 | 0

Messages

100

90

80

70

60

50

40

30

20

10

0

4 PM

Incoming Messages

Outgoing Messages

Create queue

Service Bus

Name \*

Max queue size

1 GB

Max delivery count \*

10

Message time to live

Days

Hours

Minutes

Seconds

14

0

0

0

Lock duration

Days

Hours

Minutes

Seconds

0

0

1

0

☐ Enable dead lettering on message expiration

☐ Enable partitioning

Create

Give feedback

# App Services 6-7

## ❑ Sending messages to a queue:

```
MessagingFactory factory = MessagingFactory.Create(uri, tokenProvider);
QueueClient queueClient =
    messagingFactory.CreateQueueClient("DataCollectionQueue");

try
{
    BrokeredMessage message = new BrokeredMessage("Hello, World!");
    queueClient.Send(message);
    Console.WriteLine("Message sent successfully.");
}
finally
{
    queueClient.Close();
    factory.Close();
}
```



# App Services 7-7

- ❑ Receiving messages from a queue:
  - To be able to receive message, an application must use the `MessageReceiver` object, which is created from `MessagingFactory` using `CreateMessageReceiver`.
  - For example, the following code can be used to create the `MessageReceiver` object:

```
MessageReceiver receiver = factory.CreateMessageReceiver  
                                ("DataCollectionQueue");  
BrokeredMessage receivedMessage = receiver.Receive();
```

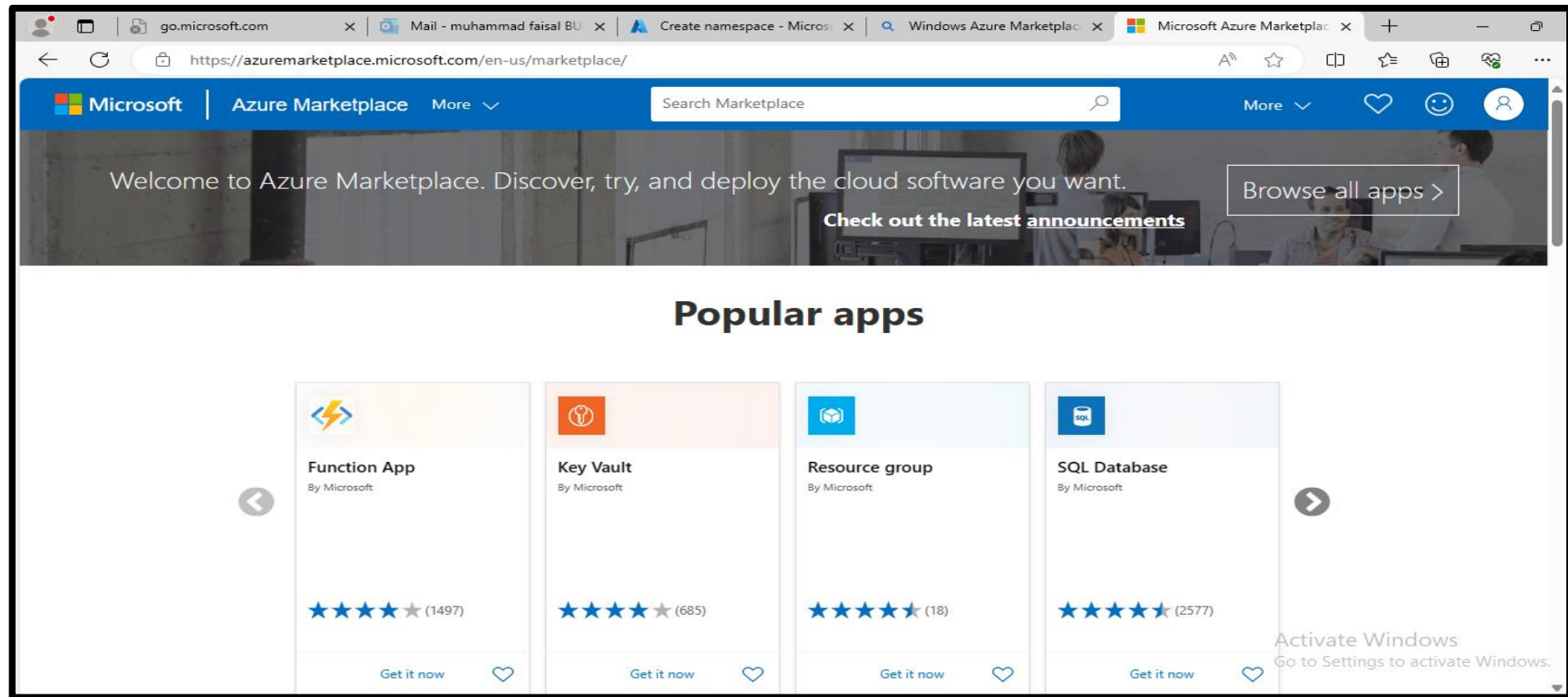
# Commerce 1-3

- ❑ With the onset of Software as a Service (SaaS):
  - Creation and usage of cloud applications have become commercial.
  - Usage of cloud applications and the associated payments are made online.
- ❑ Windows Azure:
  - Allows potential customers to search and buy Windows Azure applications and commercial datasets deployed in the cloud.
  - The service providers for such commercial exchange include **Windows Azure Marketplace** and **Windows Azure Store**.



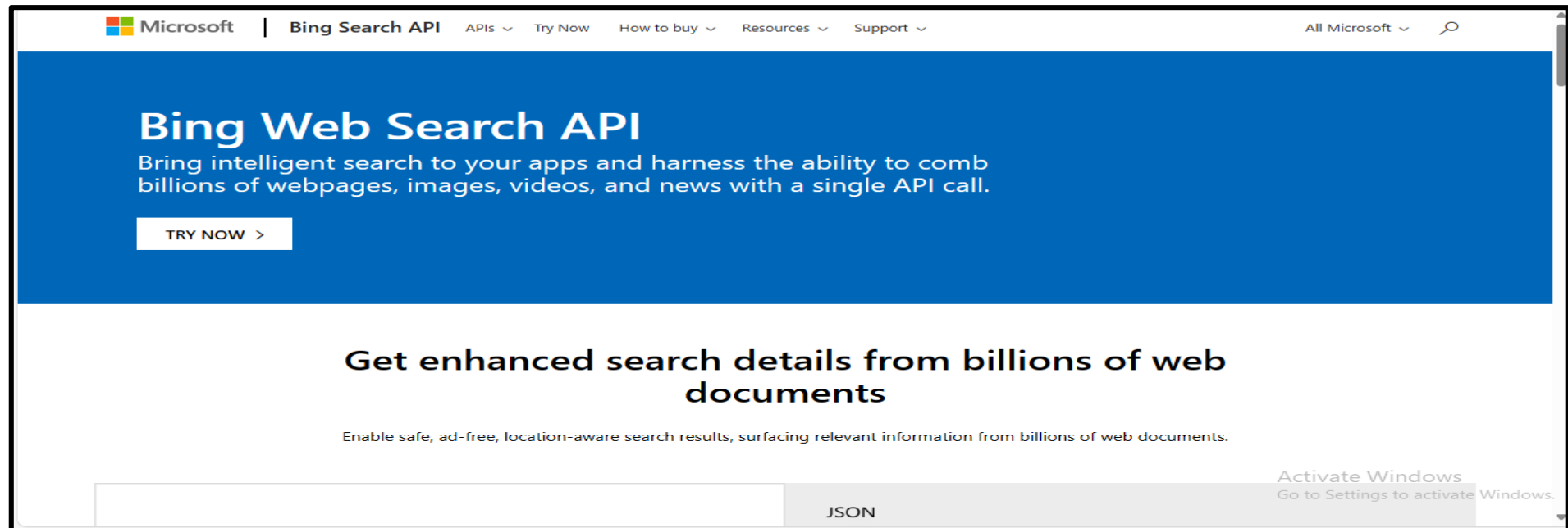
# Commerce 2-3

□ The following figure shows the Windows Azure Marketplace:



# Commerce 3-3

- Customers can search for their required applications or datasets and then sign up to use them through the application's creator or directly either through the Marketplace or Store.
- The Bing Search Application Programming Interface (API) can also be used to search for applications through the Marketplace.



# Creating a Azure Cloud Service

- ❑ A cloud service must be created before it can be published in Windows Azure.
- ❑ There are two methods to create a service:

**Using the Windows  
Azure Management  
Portal**



OR

**Using Visual Studio**



# Creating a Azure Cloud Service Using VS

- ❏ Steps to create an Azure cloud service using Visual Studio are:

**1. Create an Account**

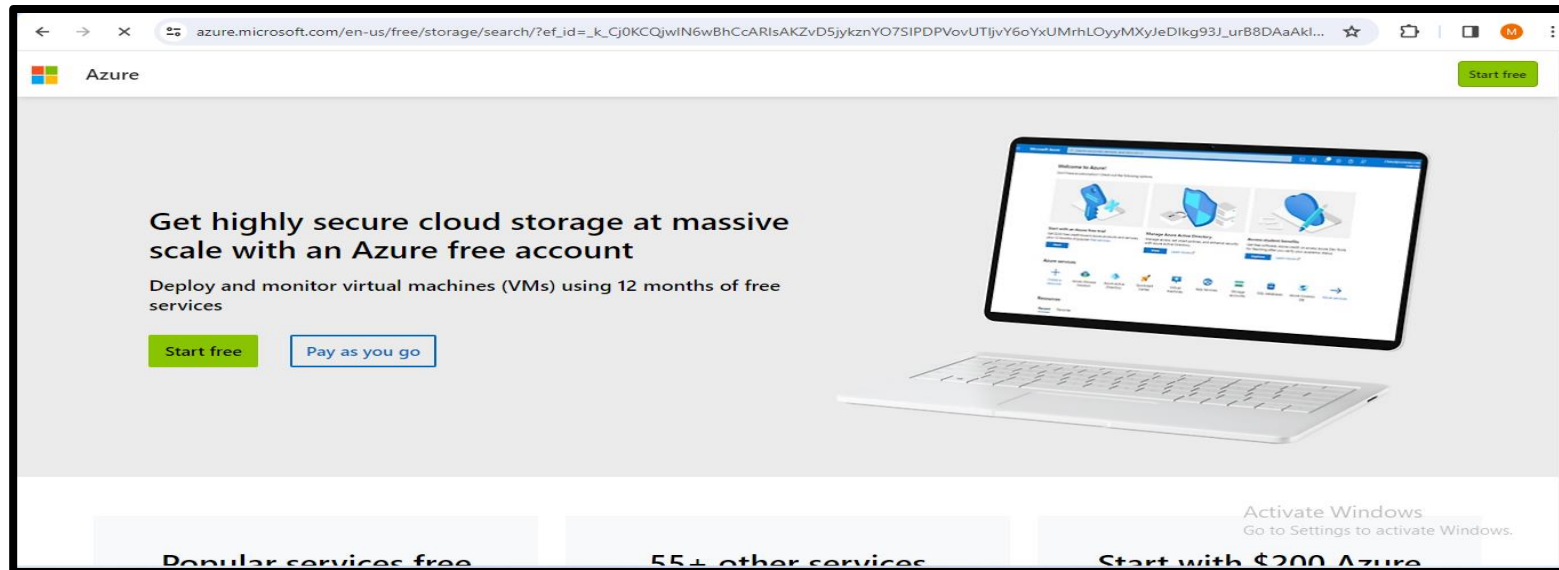
**2. Create a Cloud Service**



# Creating a Azure Cloud Service Using VS

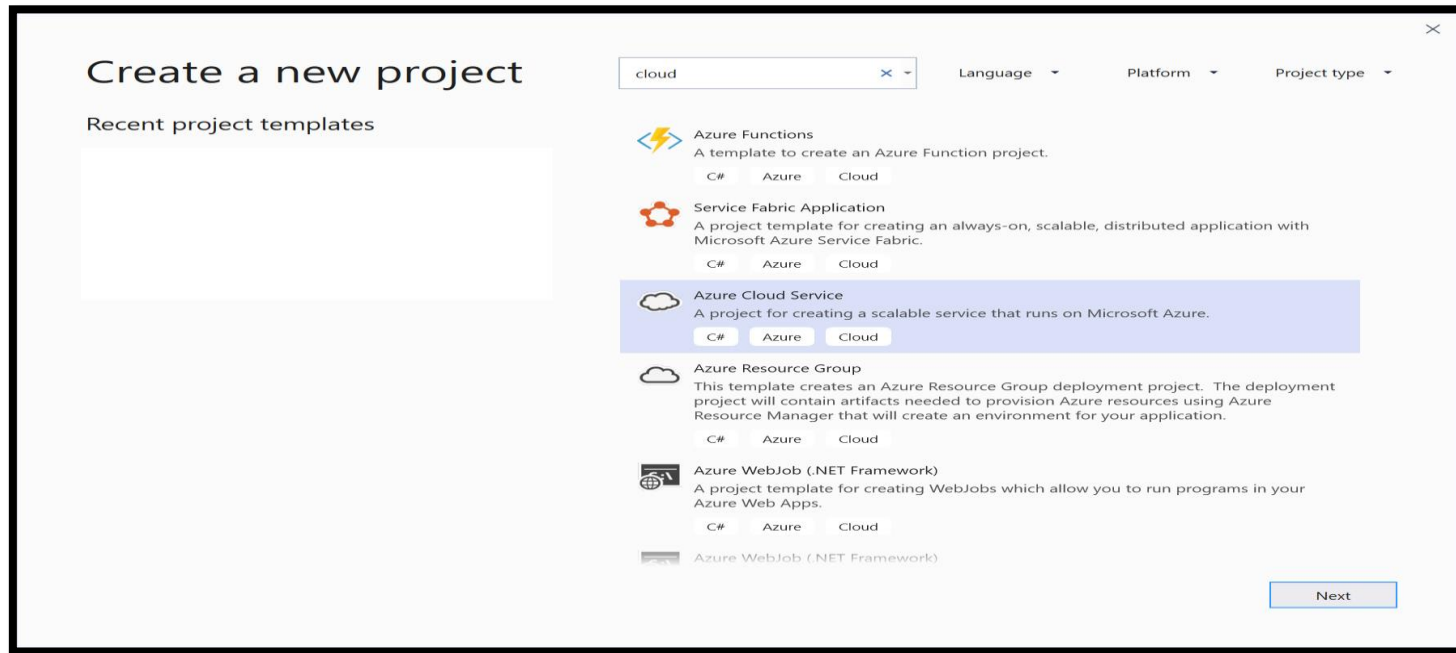
## 1. Create an Account

- ❑ Connect to <https://azure.microsoft.com> and create an account, if you do not have one.
    - If you already have an account, access it by clicking the **Portal** button.
- Windows Azure also provides a free trial as shown in the following figure:



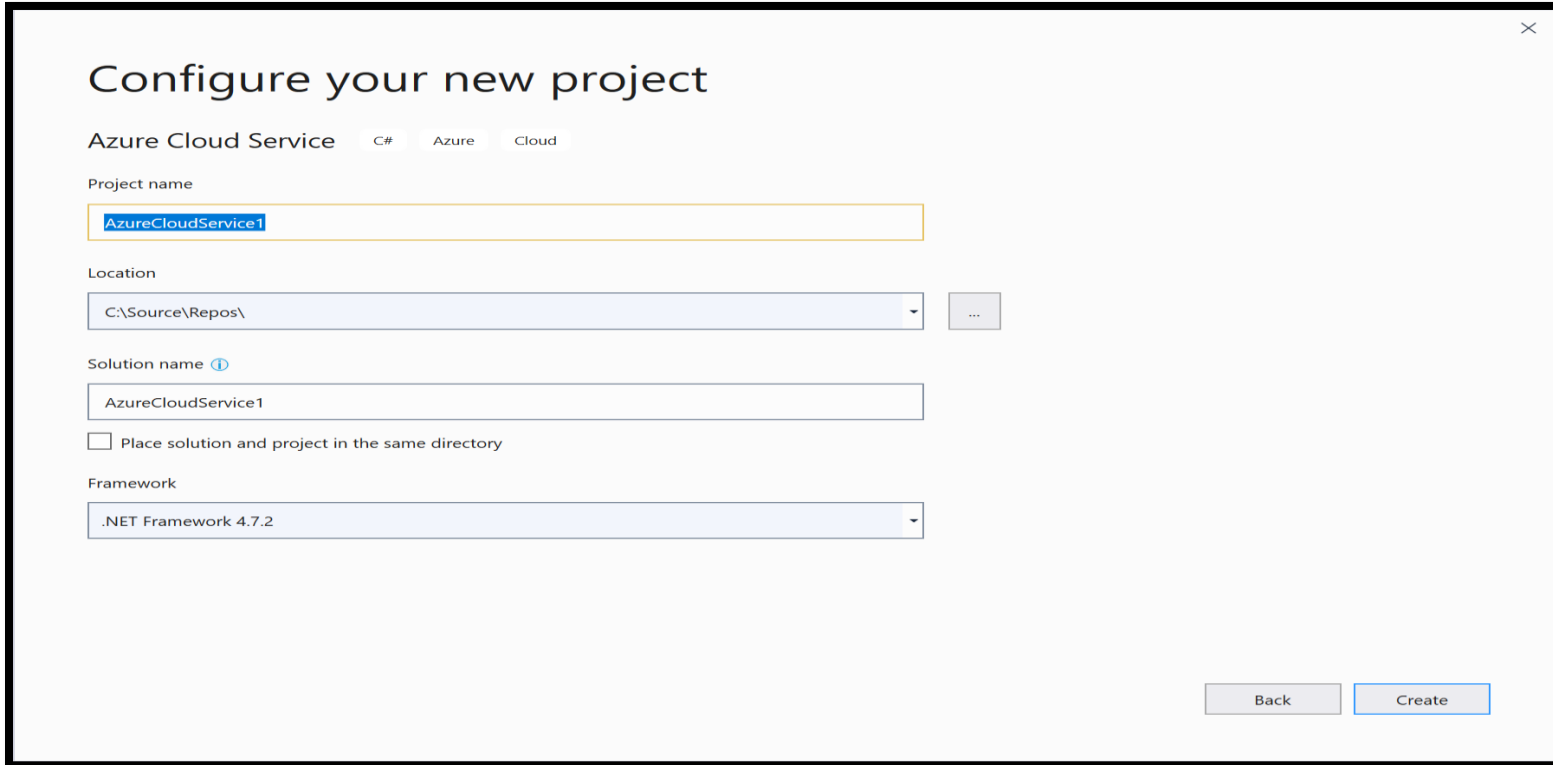
## 2. Create a Cloud Service

1. From the start window, choose **Create a new project**.
2. In the search box, type in *Cloud*, and then choose **Azure Cloud Service**.





### 3. Give the project a name and choose **Create**.



Configure your new project

Azure Cloud Service C# Azure Cloud

Project name

AzureCloudService1

Location

C:\Source\Repos\

Solution name ⓘ

AzureCloudService1

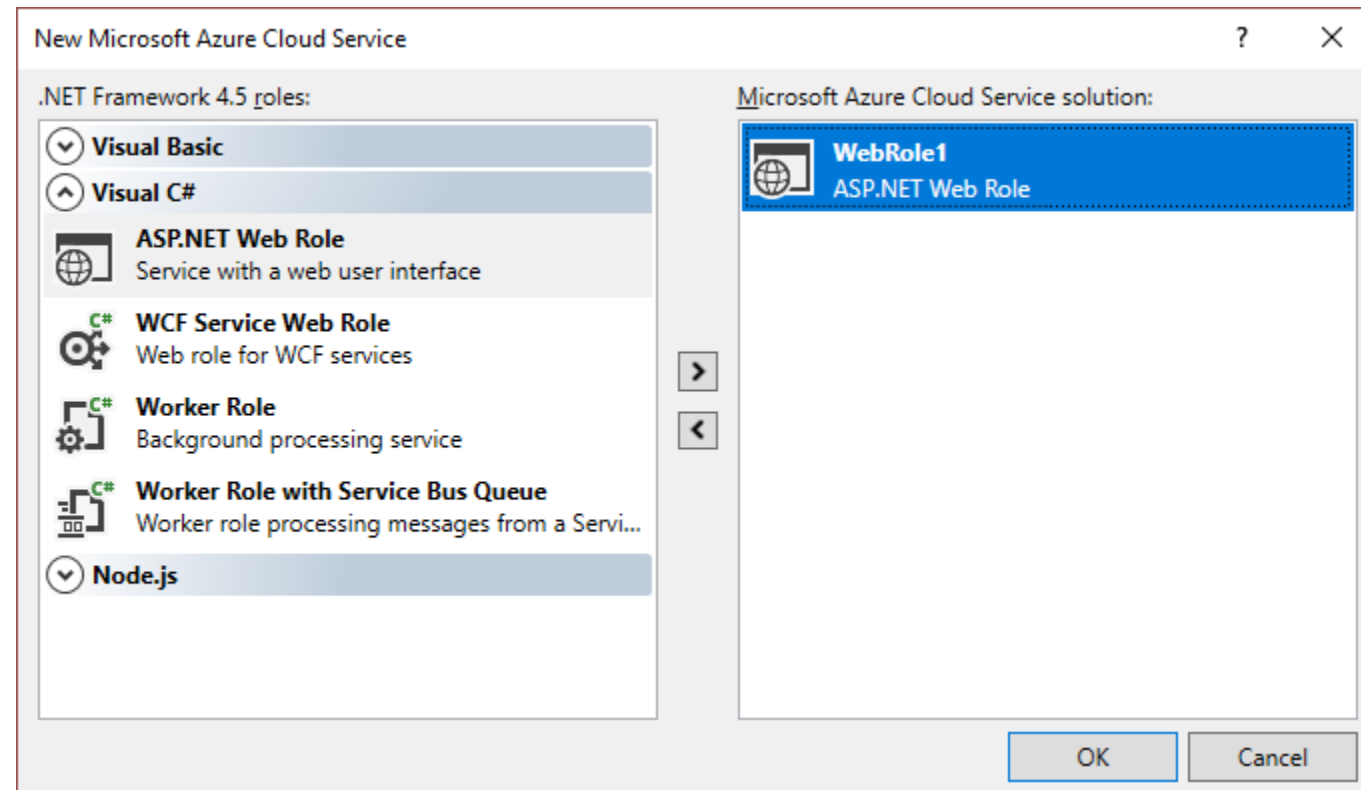
☐ Place solution and project in the same directory

Framework

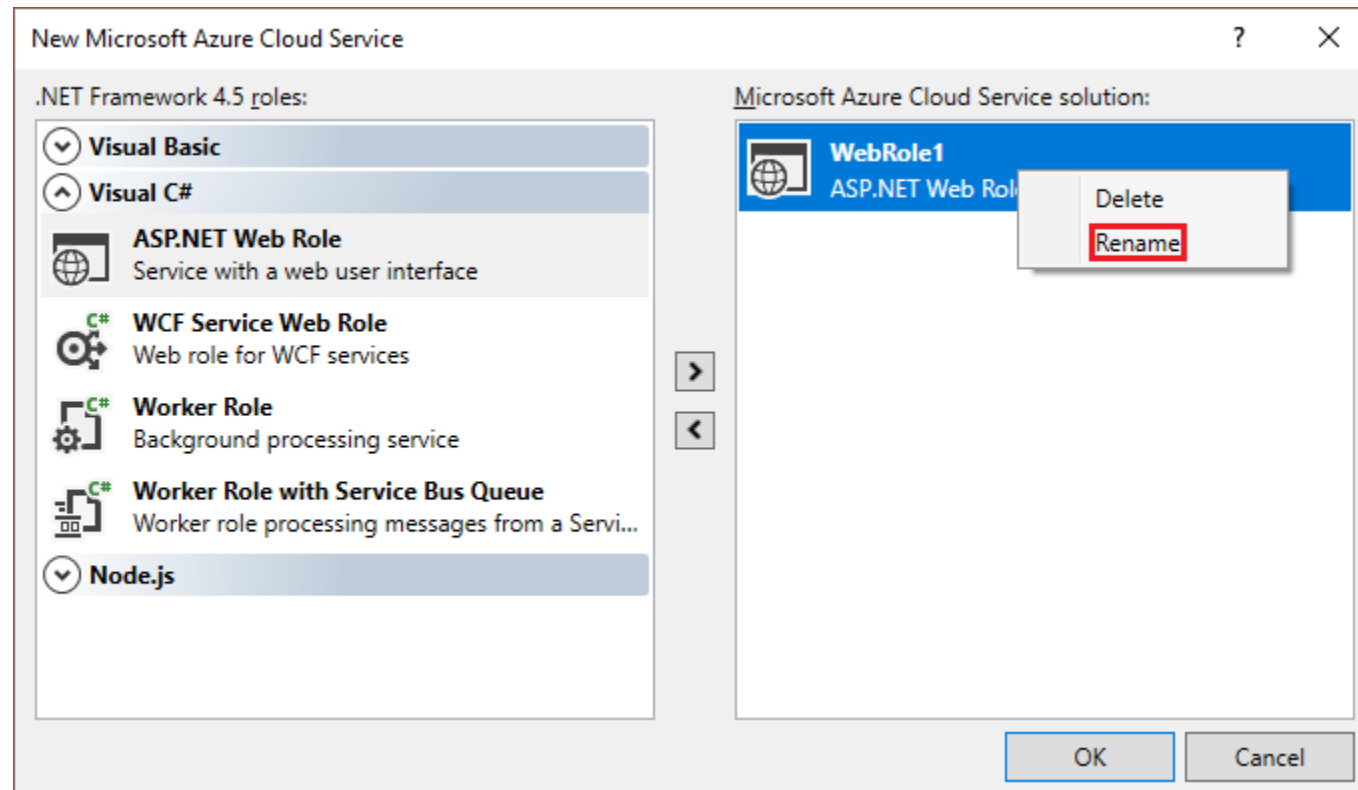
.NET Framework 4.7.2

Back Create

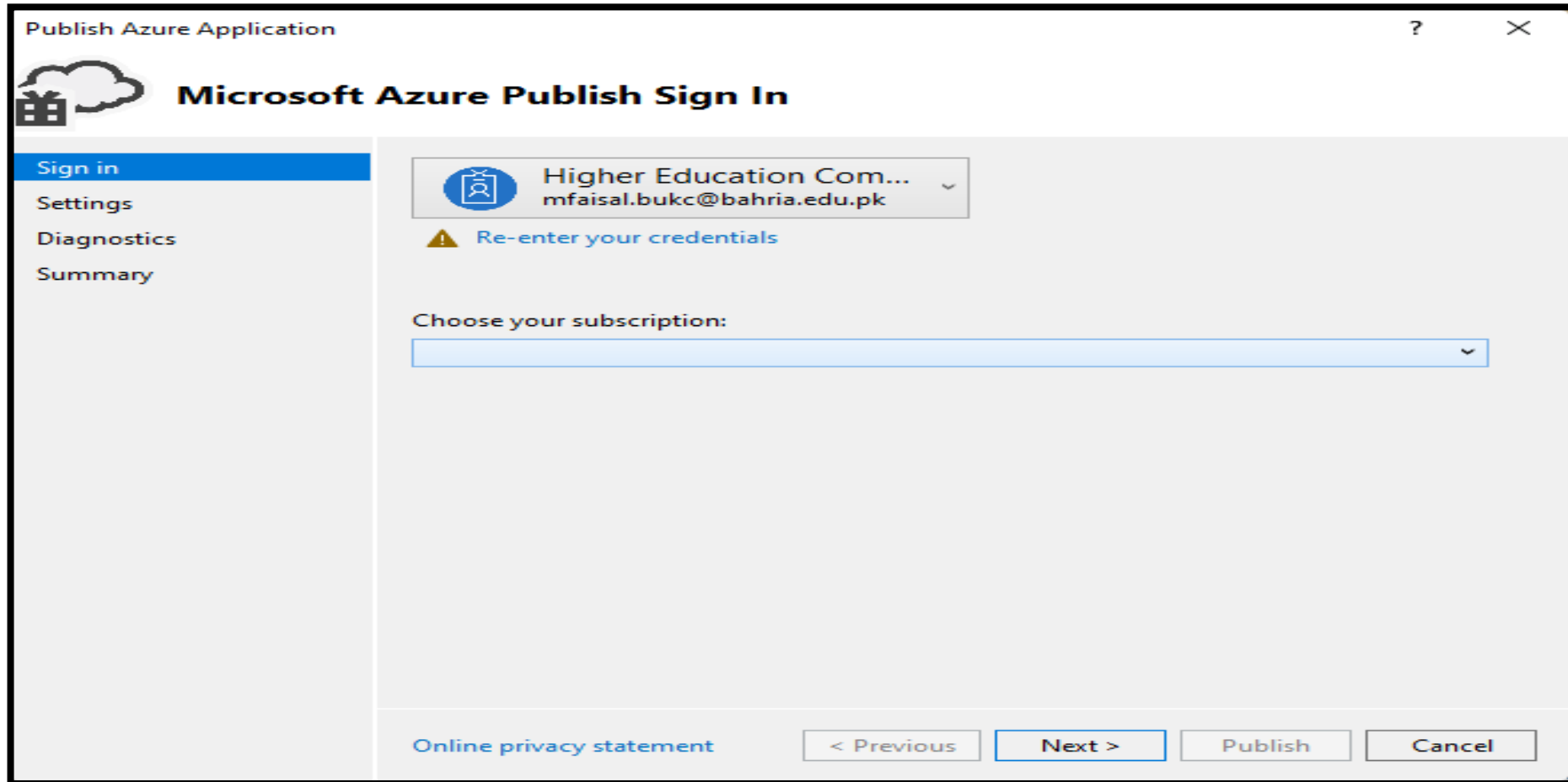
4. In the **New Microsoft Azure Cloud Service** dialog, select the roles that you want to add, and choose the right arrow button to add them to your solution.



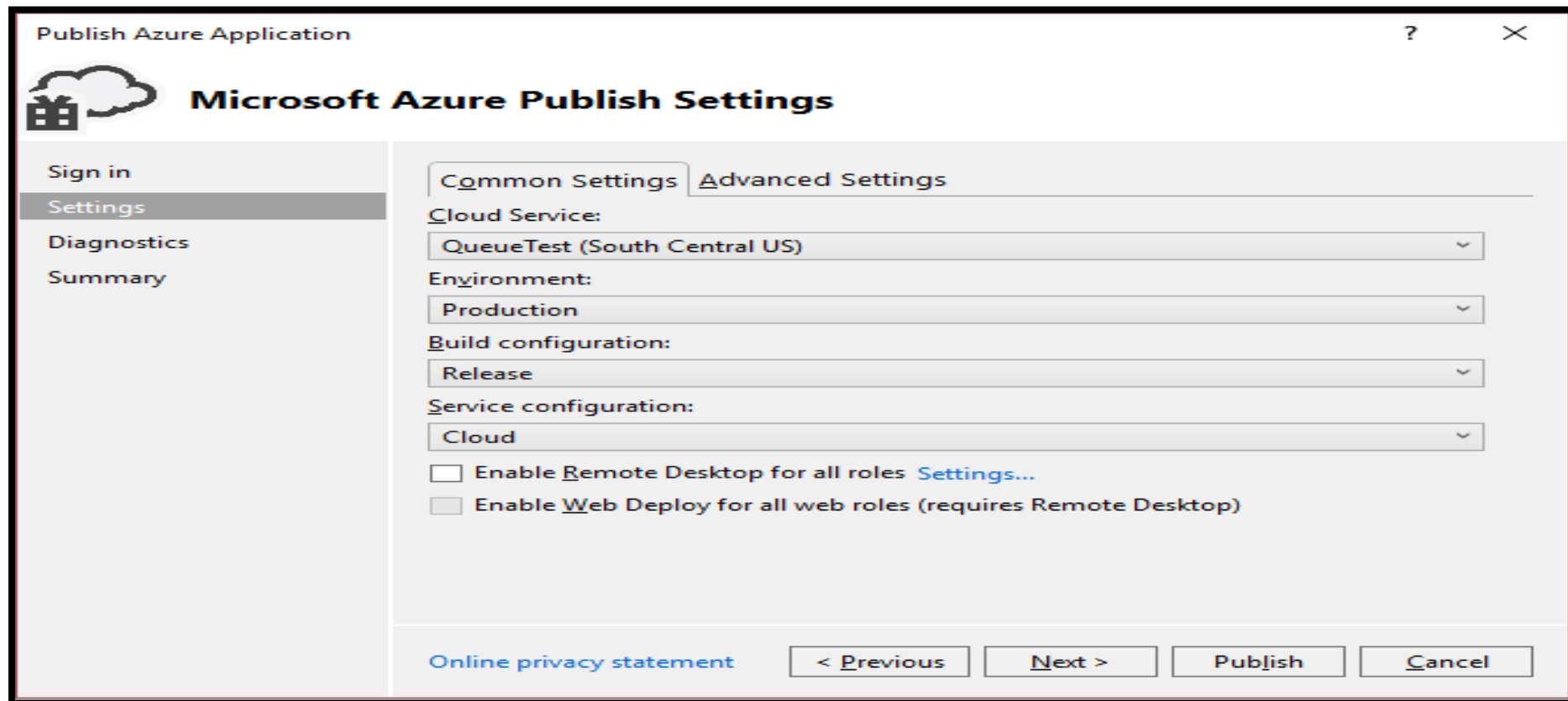
5. To rename a role that you've added, hover on the role in the **New Microsoft Azure Cloud Service** dialog, and, from the context menu, select **Rename**.



After writing your code, in **Solution Explorer**, right-click the created Azure project, and, from the context menu, select **Publish**.

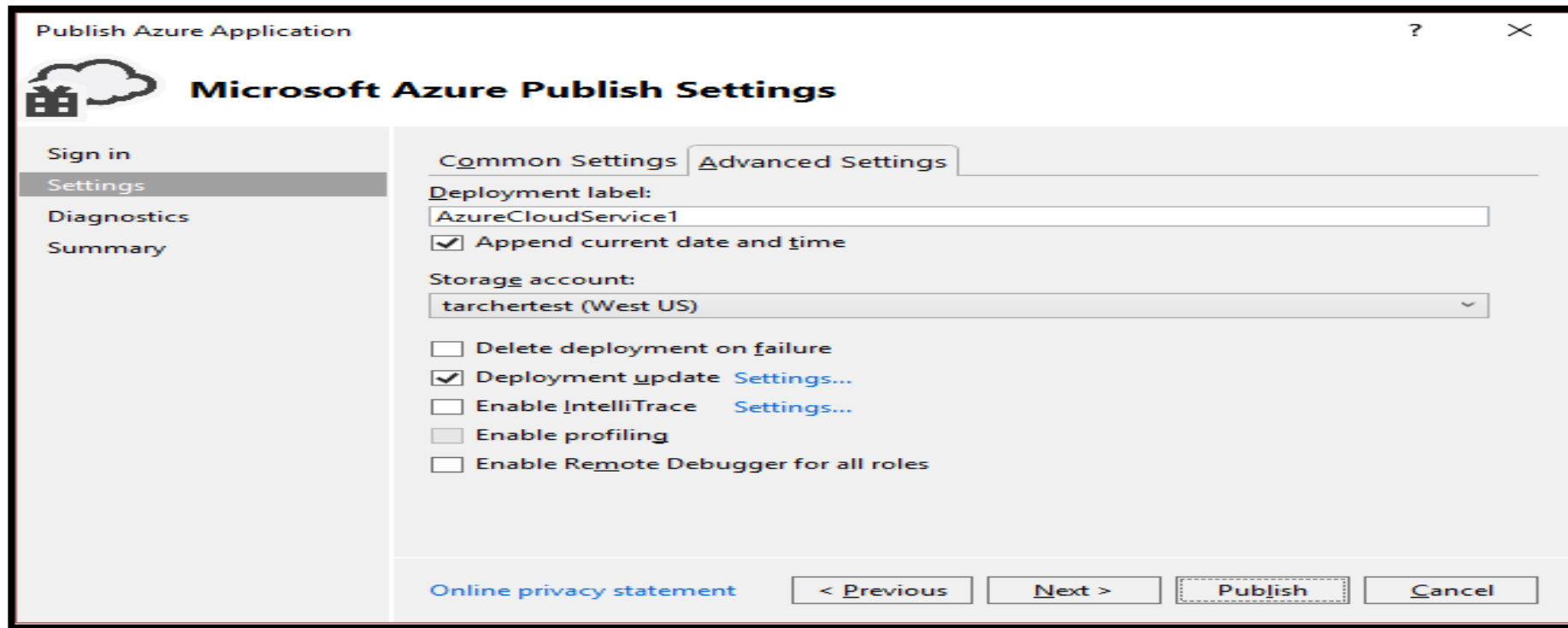


- **Environment** - Select either **Production** or **Staging**. Choose the staging environment if you want to deploy your application in a test environment.
- **Build configuration** - Select either **Debug** or **Release**.
- **Service configuration** - Select either **Cloud** or **Local**.



The screenshot shows the 'Publish Azure Application' dialog box, titled 'Microsoft Azure Publish Settings'. The dialog has a sidebar on the left with links: 'Sign in', 'Settings' (selected), 'Diagnostics', and 'Summary'. The main area has two tabs: 'Common Settings' (active) and 'Advanced Settings'. Under 'Common Settings', there are four dropdown menus: 'Cloud Service:' set to 'QueueTest (South Central US)', 'Environment:' set to 'Production', 'Build configuration:' set to 'Release', and 'Service configuration:' set to 'Cloud'. Below these are two checkboxes: 'Enable Remote Desktop for all roles' (unchecked) and 'Enable Web Deploy for all web roles (requires Remote Desktop)' (unchecked). At the bottom, there is a link for 'Online privacy statement' and four buttons: '< Previous', 'Next >', 'Publish', and 'Cancel'.

- **Deployment label** - Either accept the default name, or enter a name of your choosing. To append the date to the deployment label, leave the checkbox selected.
- **Storage account** - Select the storage account to use for this deployment, \*\*<Create New> to create a storage account. The datacenter displays in parentheses for each storage account. It is recommended that the datacenter location for the storage account is the same as the datacenter location for the cloud service (Common Settings).

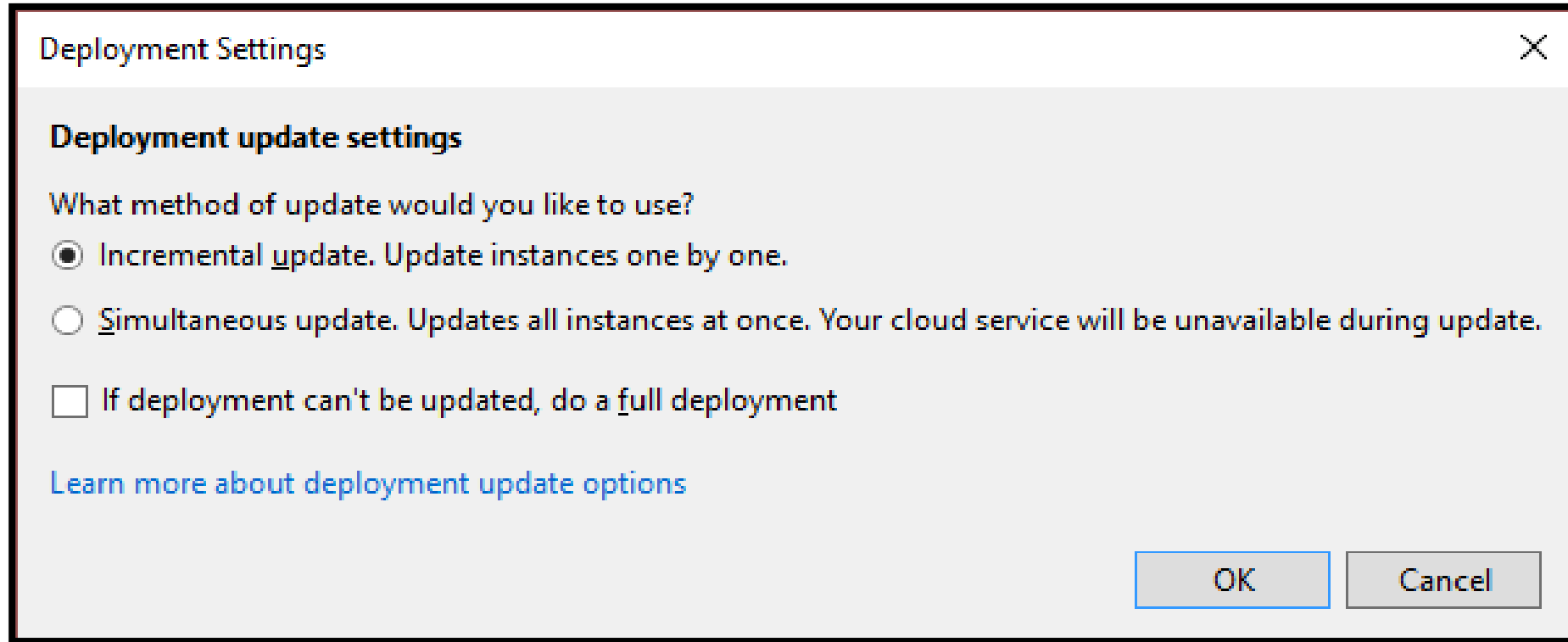


The screenshot shows the 'Publish Azure Application' dialog box, titled 'Microsoft Azure Publish Settings'. The left sidebar contains a navigation menu with 'Sign in', 'Settings' (selected), 'Diagnostics', and 'Summary'. The main area has two tabs: 'Common Settings' (active) and 'Advanced Settings'. Under 'Common Settings', the 'Deployment label' is 'AzureCloudService1'. The 'Append current date and time' checkbox is checked. The 'Storage account' dropdown is set to 'tarchertest (West US)'. Below these are five unchecked checkboxes: 'Delete deployment on failure', 'Deployment update Settings...', 'Enable IntelliTrace Settings...', 'Enable profiling', and 'Enable Remote Debugger for all roles'. At the bottom, there is a link for 'Online privacy statement' and four buttons: '< Previous', 'Next >', 'Publish' (highlighted with a dashed border), and 'Cancel'.

## Cont....

- **Delete deployment on failure** - Select this option to have the deployment deleted if any errors are encountered during publishing. *This should be unchecked if you want to **maintain a constant virtual IP address** for your cloud service.*
- **Deployment update** - Select this option if you want to **deploy only updated components**. This type of deployment can be faster than a **full deployment**. This should be checked if you want to maintain a constant virtual IP address for your cloud service.
- **Deployment update - settings** - This dialog is used to further specify how you want the roles to be updated. If you choose **Incremental update**, each instance of your application is **updated one after another**, so that the **application is always available**. If you choose **Simultaneous update**, all instances of your application are updated at the same time. Simultaneous updating is faster, but your service might not be available during the update process.

# Deployment update settings

A screenshot of a 'Deployment Settings' dialog box. The title bar says 'Deployment Settings' with a close button (X) on the right. The main content area has a section header 'Deployment update settings'. Below it is the question 'What method of update would you like to use?'. There are three options: 1. 'Incremental update. Update instances one by one.' with a selected radio button. 2. 'Simultaneous update. Updates all instances at once. Your cloud service will be unavailable during update.' with an unselected radio button. 3. 'If deployment can't be updated, do a full deployment' with an unselected checkbox. At the bottom left is a blue link 'Learn more about deployment update options'. At the bottom right are 'OK' and 'Cancel' buttons.

Deployment Settings

**Deployment update settings**

What method of update would you like to use?

☒ Incremental update. Update instances one by one.

☐ Simultaneous update. Updates all instances at once. Your cloud service will be unavailable during update.

☐ If deployment can't be updated, do a full deployment


[Learn more about deployment update options](#)

OK Cancel



# Click Publish

Publish Azure Application

 **Microsoft Azure Publish Summary**

Sign in  
Settings  
Diagnostics  
**Summary**

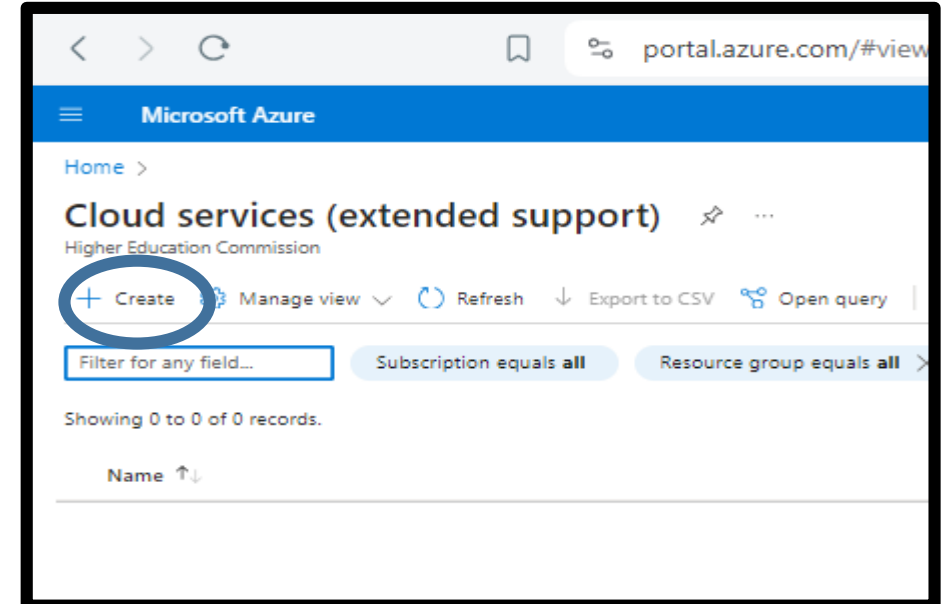
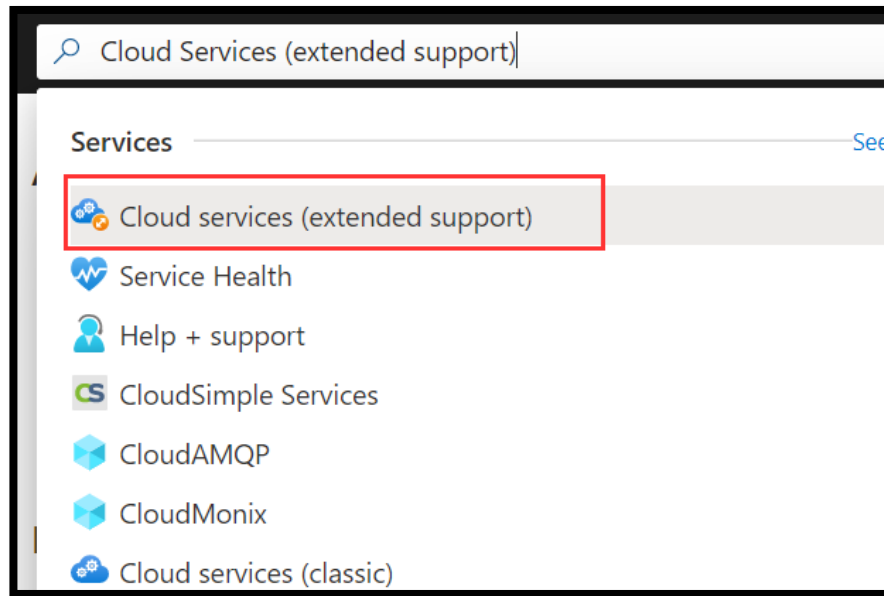
Target profile: New profile \*

<b>Deployment update:</b>	Enabled
<b>Subscription:</b>	Tom Archer Azure Subscription
<b>Cloud Service:</b>	QueueTest (South Central US)
<b>Environment:</b>	Production
<b>Build configuration:</b>	Release
<b>Service configuration:</b>	Cloud
<b>Remote Desktop:</b>	Disabled
<b>Web Deploy:</b>	Disabled
<b>Deployment label:</b>	AzureCloudService1 - 3/6/2017 6:13:18 PM
<b>Storage account:</b>	tarchertest (West US)
<b>Delete deployment on failure:</b>	Disabled
<b>IntelliTrace:</b>	Disabled

[Online privacy statement](#) < Previous Next > **Publish** Cancel

# Deploying the Cloud Service on Windows Azure

- Sign in to the [Azure portal](#).
- Using the search bar located at the top of the Azure portal, search for and select **Cloud Services**



# In the Cloud Services (extended support) pane select **Create**.

- The Cloud Services creation window will open to the **Basics** tab.
  - Select a Subscription.
  - Choose a resource group or create a new one.
- Enter the desired name for your Cloud Service deployment.
- Select the region to deploy to.

Create a cloud service (extended support) (preview)

Basics Configuration Tags Review + create

**i** Cloud Services (extended support) is in public preview. To create a cloud service (extended support), you must register your subscription. [Learn more](#)

Cloud Services (extended support) is a new Azure Resource Manager (ARM) based deployment model for Azure Cloud Services. Azure Cloud Services is a platform as a service (PaaS) offering that allows you to quickly deploy and manage powerful applications and services. It manages provisioning, load balancing, and health monitoring. [Learn more about Azure Cloud Services \(extended support\)](#)

**Project details**

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* ⓘ ASR PM team subscription 3

Resource group \* ⓘ myResourceGroup  
[Create new](#)

**Instance details**

Cloud service name \* ⓘ myNewCloudService ✓

**i** The DNS name of the cloud service is specified by the DNS name label of the public IP address and can be modified in the public IP section on the configuration tab.

Region \* ⓘ (US) West US 2

# Click Review + Create

## •Service Definition

The cloud service definition file (.csdef) defines the service model, including the number of roles.

## •Service Configuration

The cloud service configuration file (.cscfg) provides configuration settings for the cloud service and individual roles, including the number of role instances.

## •Service Package


The service package (.cspkg) contains the application code and configurations and the service definition file.


### Cloud service configuration, package, and service definition


Include your cloud service configuration (.cscfg), application package (.cspkg, .zip), and service definition (.csdef) files. We will store them in your selected storage account and use them to configure your cloud service. If you are coming from Cloud Services (classic), you will need to make some changes to your service configuration (.cscfg) and service definition (.csdef) files as a pre-requisite to deploy a cloud service (extended support). [Learn more about these pre-requisites](#)

Package/configuration/service definition location ☒ From local ☐ From blob

Storage account \* ⓘ

Upload a package (.cspkg, .zip) ⓘ  

Upload a configuration (.cscfg) ⓘ  

Upload a service definition (.csdef) ⓘ  

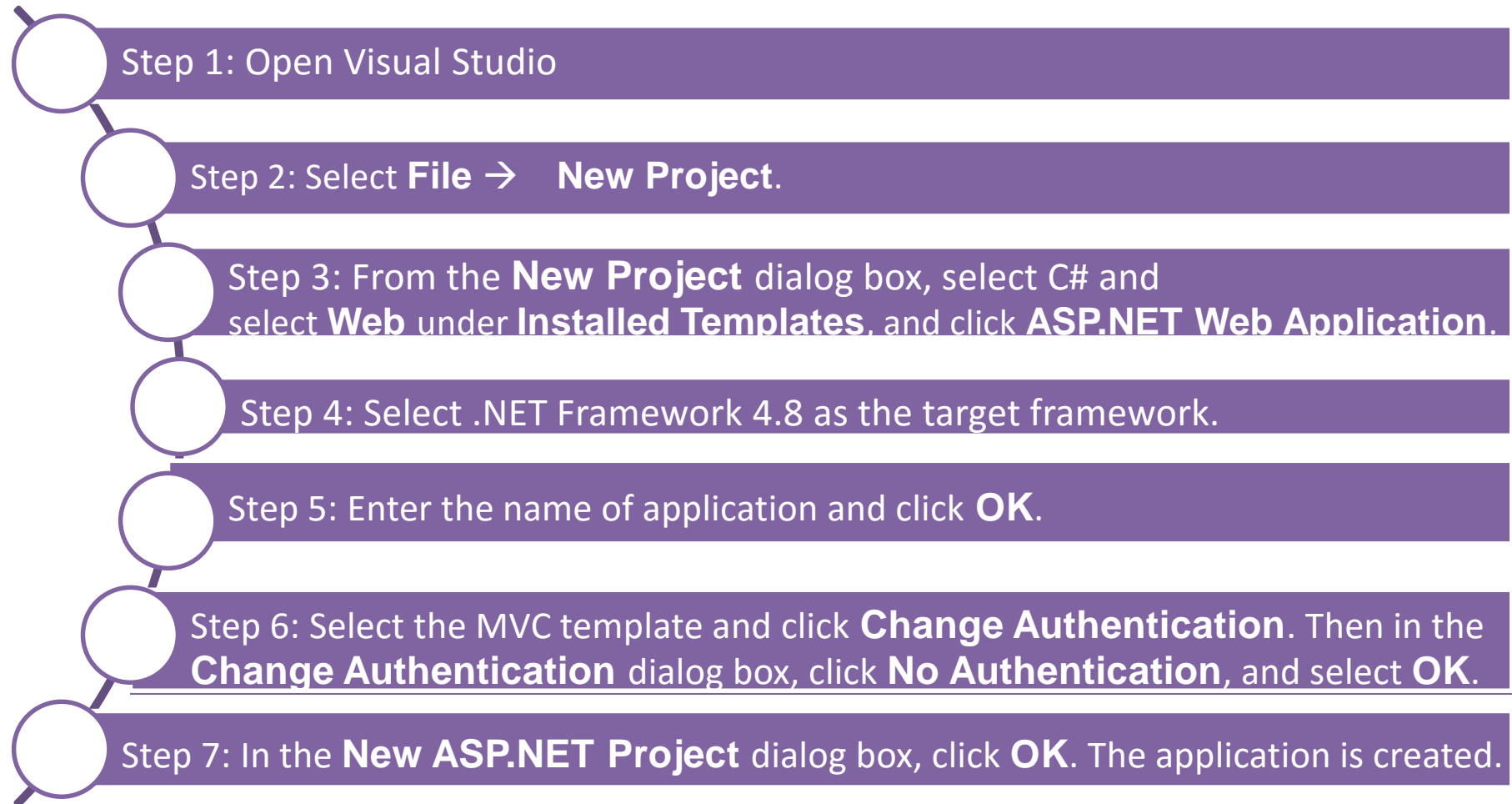
Review + create< PreviousNext : Configuration >

# Creating and Deploying a Web Application to Windows Azure

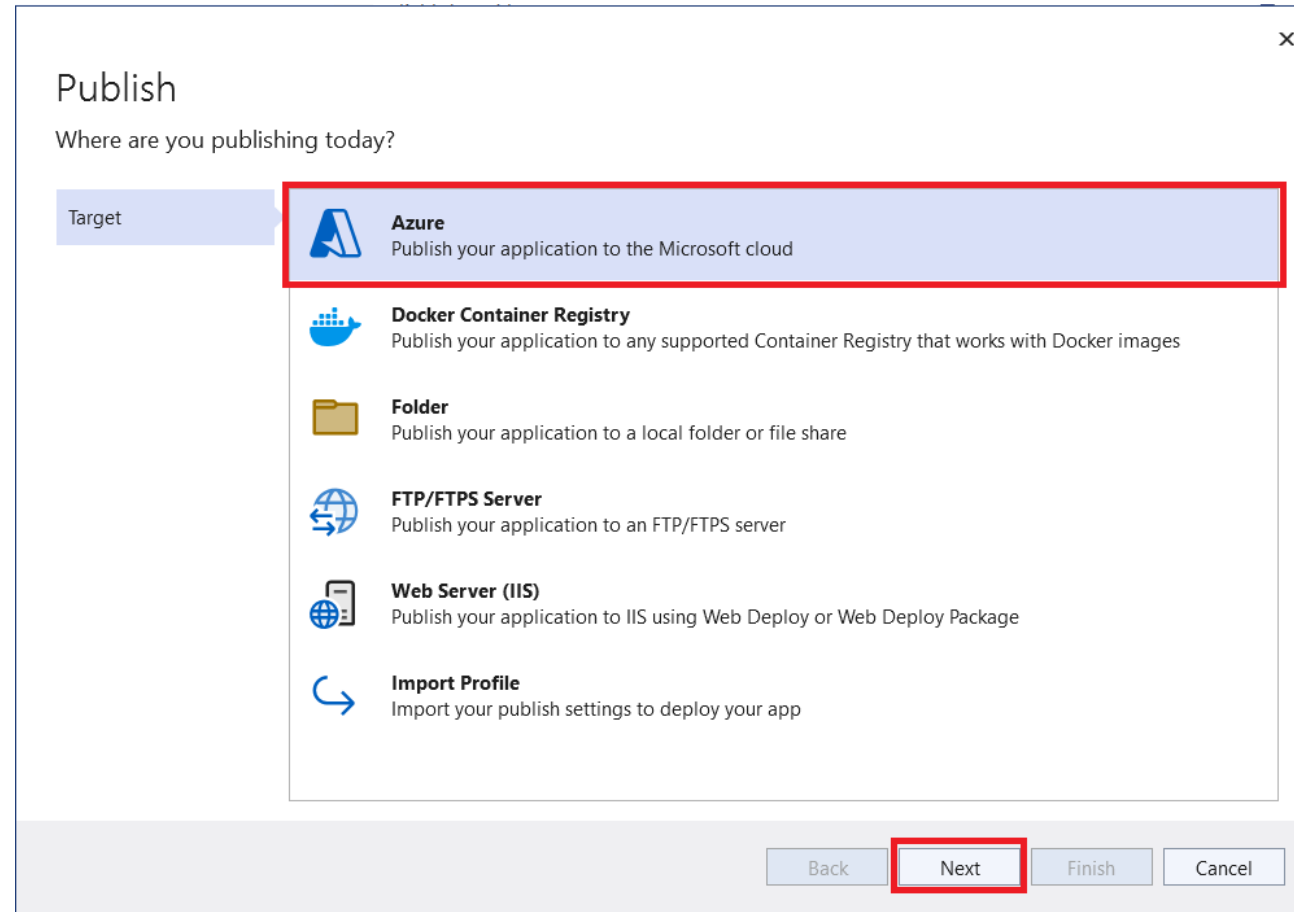
- ❑ With the Windows Azure SDK, it is easy for the Web developers to:
  - Create and deploy Web applications and deploy them to Windows Azure.
- ❑ The developer must download and install Windows Azure SDK for .NET.



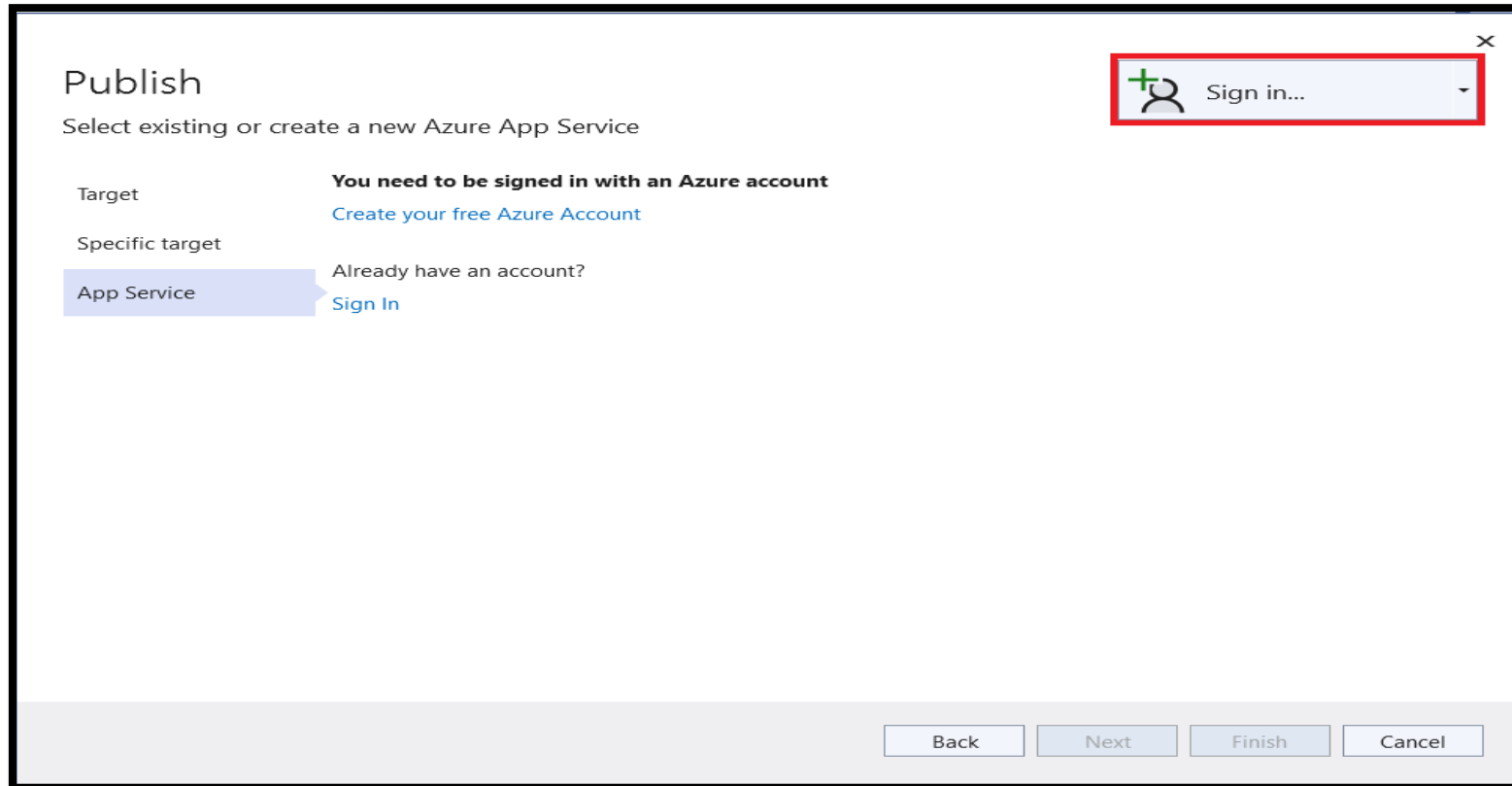
# Creating an ASP.NET MVC Application



1. In **Solution Explorer**, right-click the **MyFirstAzureWebApp** project and select **Publish**.
2. In **Publish**, select **Azure** and then **Next**.

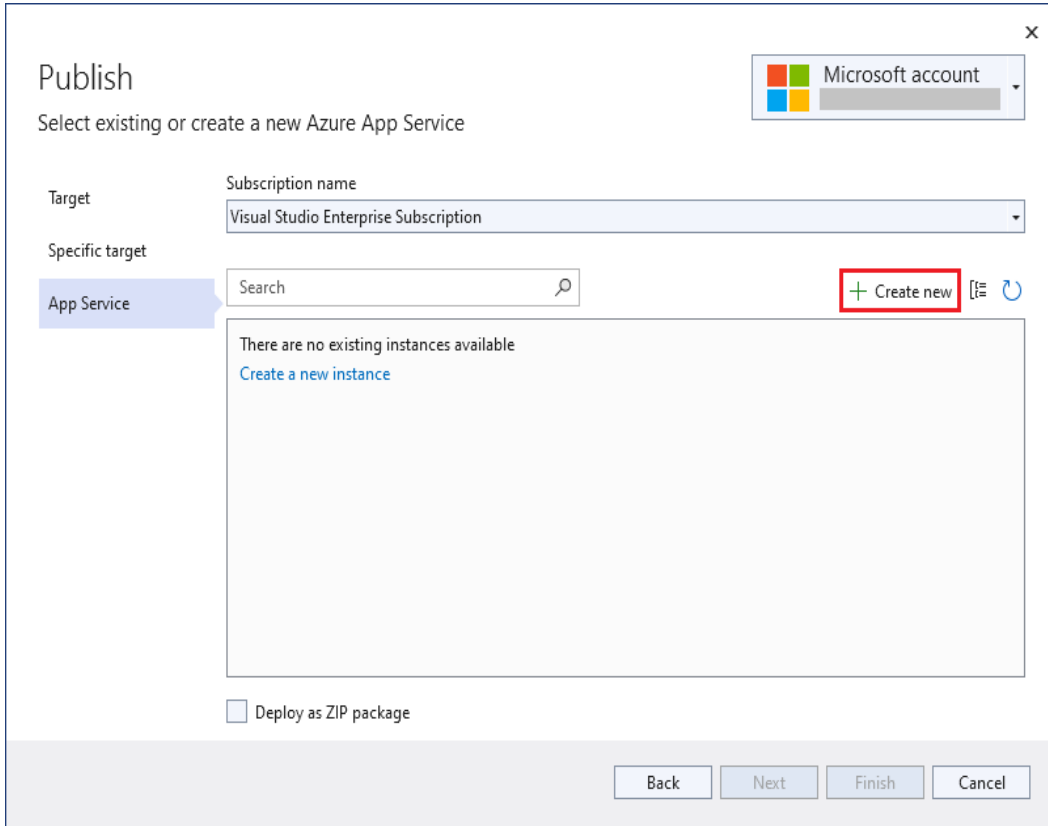


3. Choose the **Specific target**, either **Azure App Service (Linux)** or **Azure App Service (Windows)**. Then, select **Next**.
4. Your options depend on whether you're signed in to Azure already and whether you have a Visual Studio account linked to an Azure account. Select either **Add an account** or **Sign in** to sign in to your Azure subscription. If you're already signed in, select the account you want.





# Cont....



**Publish**  
Select existing or create a new Azure App Service

Microsoft account

Target: Subscription name  
Visual Studio Enterprise Subscription

Specific target: App Service

Search

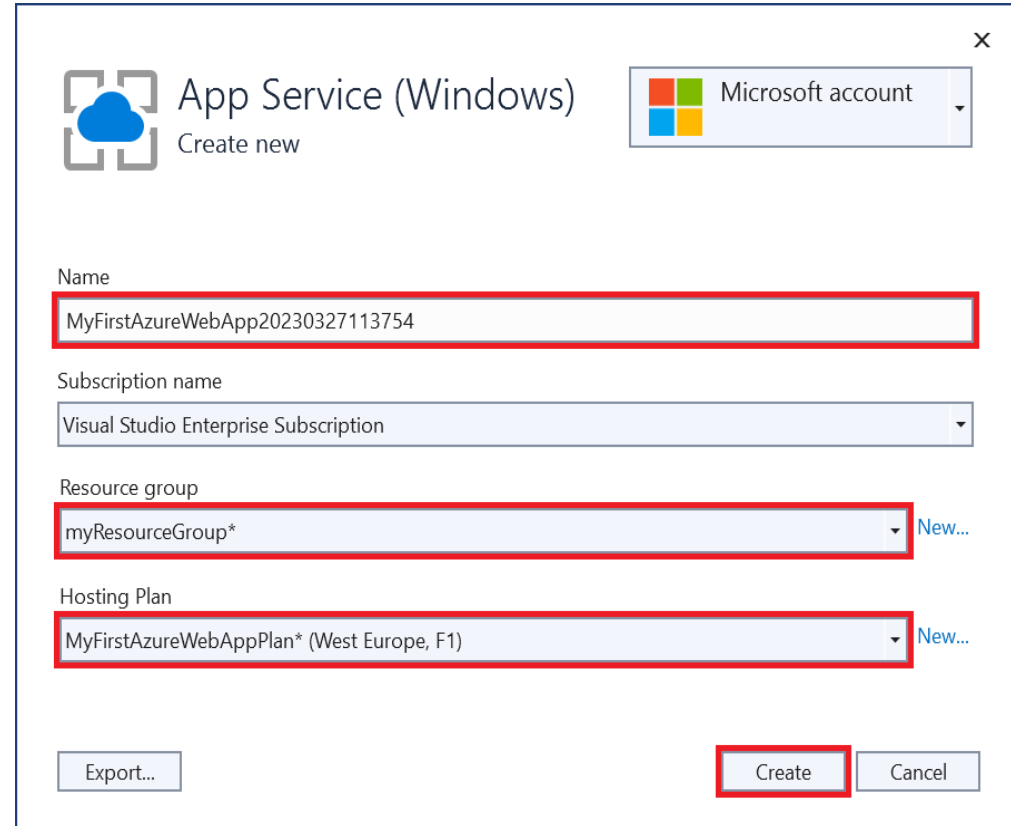
**+ Create new**

There are no existing instances available  
[Create a new instance](#)

☐ Deploy as ZIP package

Back Next Finish Cancel

**Click Create new**



**App Service (Windows)**  
Create new

Microsoft account

Name: MyFirstAzureWebApp20230327113754

Subscription name: Visual Studio Enterprise Subscription

Resource group: myResourceGroup\* [New...](#)

Hosting Plan: MyFirstAzureWebAppPlan\* (West Europe, F1) [New...](#)

Export... **Create** Cancel

**Click Create**

## Cont....

- In the **Publish** dialog, ensure your new App Service app is selected, then select **Finish**, then select **Close**. Visual Studio creates a publish profile for you for the selected App Service app.
- In the **Publish** page, select **Publish**. If you see a warning message, select **Continue**.
- Visual Studio builds, packages, and publishes the app to Azure, and then launches the app in the default browser.

# Summary 1-2

- ❑ The Windows Azure OS is the central component of the Windows Azure Services Platform.
- ❑ Windows Azure OS functions handle load balancing, caching, redundancy, resource management, and life cycles of hosted services.
- ❑ Windows Azure provides four core services, namely Virtual Machines, Cloud Services, Websites, and Mobile Services.
- ❑ Windows Azure provides three main data services, namely, Cache, HDInsight, and Recovery Services.
- ❑ Datacenters are geographically spread out and they run Windows Azure cloud OS to manage and store applications and data.

# Summary 2-2

- ❑ Windows Azure Marketplace allows potential customers to search and buy the Windows Azure applications and commercial data sets deployed in the cloud.
- ❑ A cloud service must be created before it can be published in Windows Azure.
- ❑ Windows Azure Compute Emulator and Windows Storage Emulator, together help you to test basic functionalities of your services, such as configuration, roles, its instances, and its user interface status.
- ❑ Using Visual Studio, you can perform a number of tests on a cloud service.