

Session 14: Deploying Web Applications and Services

Session Objectives

- Explain Azure App Services
- Describe how to work with Azure App Services
- Explain deployment of Azure Web Applications
- Outline the process of deploying WCF data services

For Aptech Centre Use Only

Introduction to Azure App Services and Pricing Models

Azure App Service Web apps (or Web apps) is useful to host Web applications, REST APIs, and mobile backends.



Web Apps

Key features of Web apps:

Supports .NET, Java, PHP, Node.js, and Python

Supports load balancing and has built-in autoscale

Promises high availability with auto-patching and offer automated management

Offers benefits of DevOps optimization capabilities

Supports continuous integration, has deployment slots, provides testing in production, and Web jobs

Includes an extensive list of templates such as WordPress, Umbraco, Joomla, and Drupal

Mobile Apps

Key features of Mobile Apps in Azure:

- Built-in autoscale support for Mobile Apps
- Send push notifications with customer segmentation
- Provides native sync experience which enables app to work offline
- Supports social integration with Facebook, Twitter, and Google
- Supports traffic management
- Supports continuous integration/deployment with Visual Studio Online, GitHub, and Bitbucket
- Offers hybrid connections and virtual networking support to on-premises datacenters to access data securely
- Supports staged deployment and production testing
- For long-running background tasks, apps offer WebJobs support



Logic Apps

Key features of Logic apps:

Creating business processes and workflows visually

Helps in offering integration capabilities in Web, Mobile, and API Apps

Supports integrating with SaaS and enterprise applications

Supports automating EAI/B2B and business process

Helps connecting to on-premises data

API Apps

Key features of API apps:

Supports integrating with SaaS and enterprise applications

Helps automating versions and deploying of API Apps

Secure APIs with Single Sign-On, Active Directory, and OAuth

Supports internal sharing of APIs with organizational gallery

Web Apps

- Web Apps that scale with business

Mobile Apps

- Build mobile apps for any device

Azure App Services

Logic Apps

- Automate business process across SaaS and on-premises

API Apps

- Easily build and consume APIs in the cloud

App Service Plans

- An App Service plan provides resources for a Web app to run in Azure App Service.
- This tier controls what App Service features a subscriber receives and what is the payment for the plan.
- Pricing tiers are classified into following categories:

Shared compute

Dedicated compute

Isolated

Consumption

App Services Pricing Model

Different options on Free and Shared (preview) plans allow the developers to test apps according to the budget.

Basic, Standard, and Premium plans

- Production workloads and apps that execute on dedicated Virtual Machine instances that support various application and domains

• Isolated plan

- Apps that run on a private dedicated Azure environment
- Apps that needs to be securely connected with on-premises network or requires additional performance and scaling

Working with Azure App Services

Azure App Service capabilities:

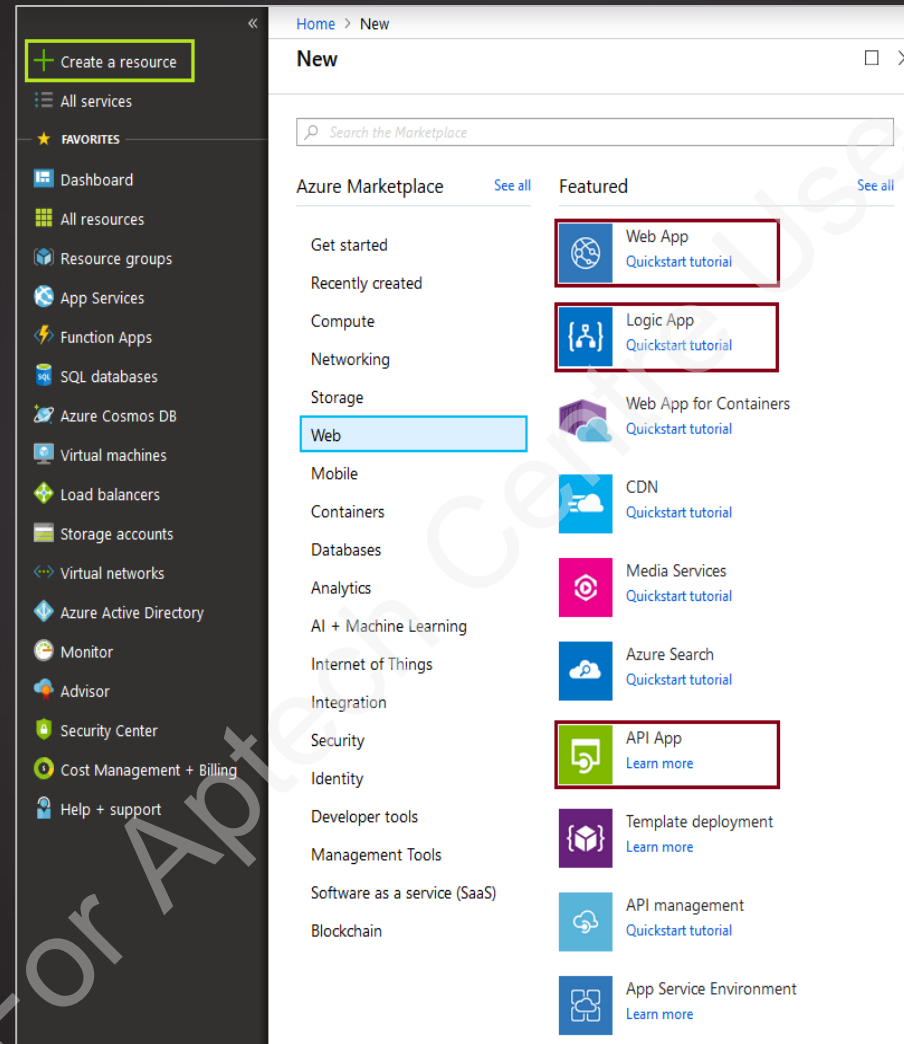
Deploying Web Apps and Mobile Apps happens in seconds

Building appealing and agreeable apps on Windows, iOS, and Android platforms

Using visual design experience, automating business processes gets easier

Integrating with 'Software as a Service' (SaaS) applications and on-premises applications is simple

Creating Web App, API App, and Logic App [1-2]



Creating Web App, API App, and Logic App

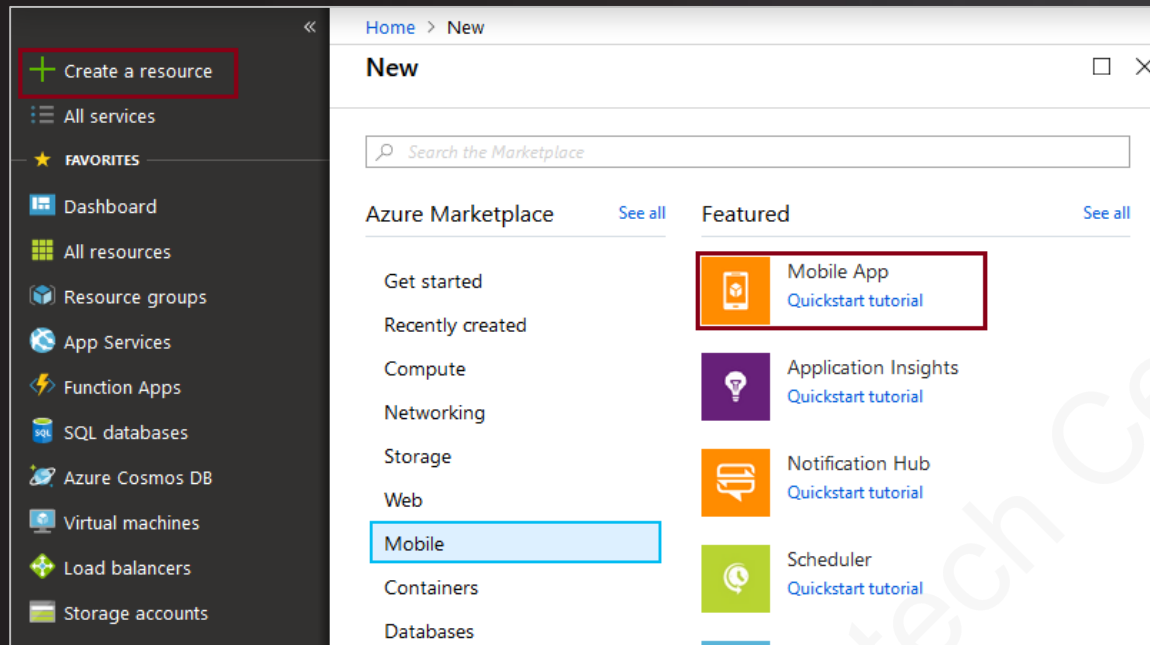
Creating Web App, API App, and Logic App [2-2]

The image displays three side-by-side screenshots of the Azure portal's 'Create' pages for different types of applications. Each panel shows a form with various configuration options.

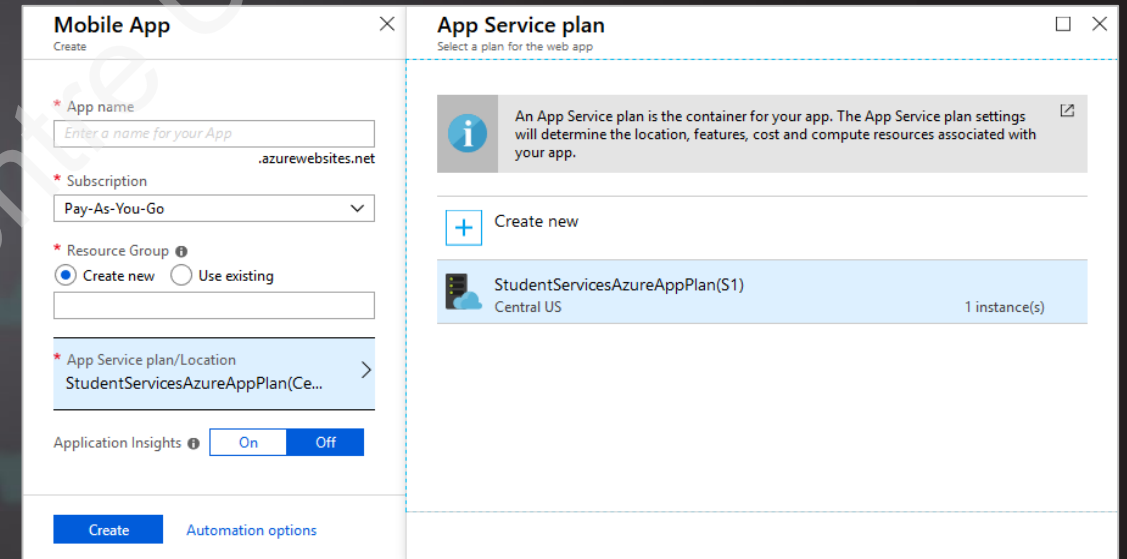
- Create logic app:** Includes fields for Name, Subscription (Pay-As-You-Go), Resource group (Create new), Location (Central US), and Log Analytics (On/Off). A note states: 'You can add triggers and actions to your Logic App after creation.' A 'Create' button and 'Automation options' link are at the bottom.
- API App:** Includes fields for App name, Subscription (Pay-As-You-Go), Resource Group (Create new), and App Service plan/Location (StudentServicesAzureAppPlan(Ce...)). It also features Application Insights (On/Off) and a 'Create' button with 'Automation options' link.
- Web App:** Includes fields for App name, Subscription (Pay-As-You-Go), Resource Group (Create new), OS (Windows/Linux), Publish (Code/Docker image), and App Service plan/Location (StudentServicesAzureAppPlan(Ce...)). It also features Application Insights (On/Off) and a 'Create' button with 'Automation options' link.

Steps to Create Web App, API App, and Logic App

Creating Mobile App



Creating Mobile App Services

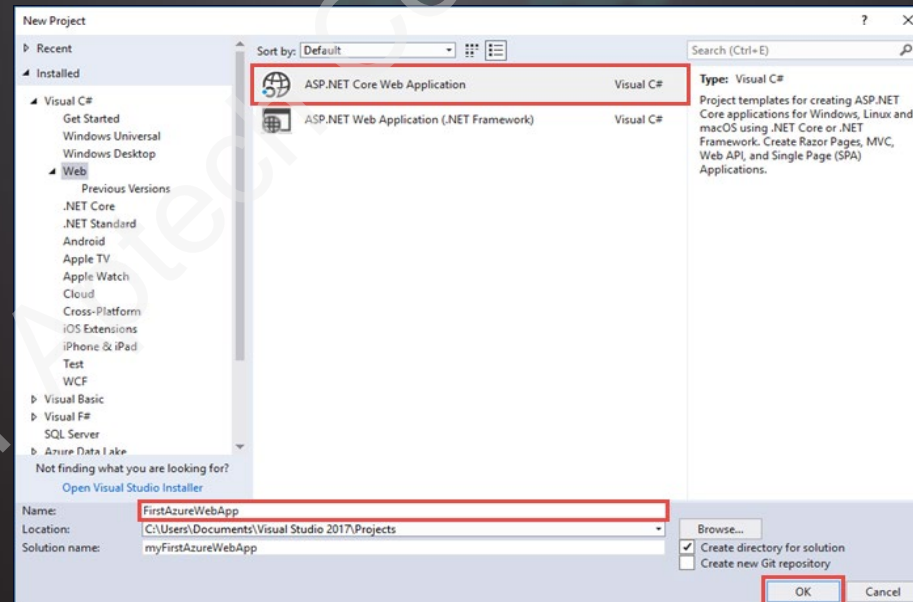


Mobile App Service Page

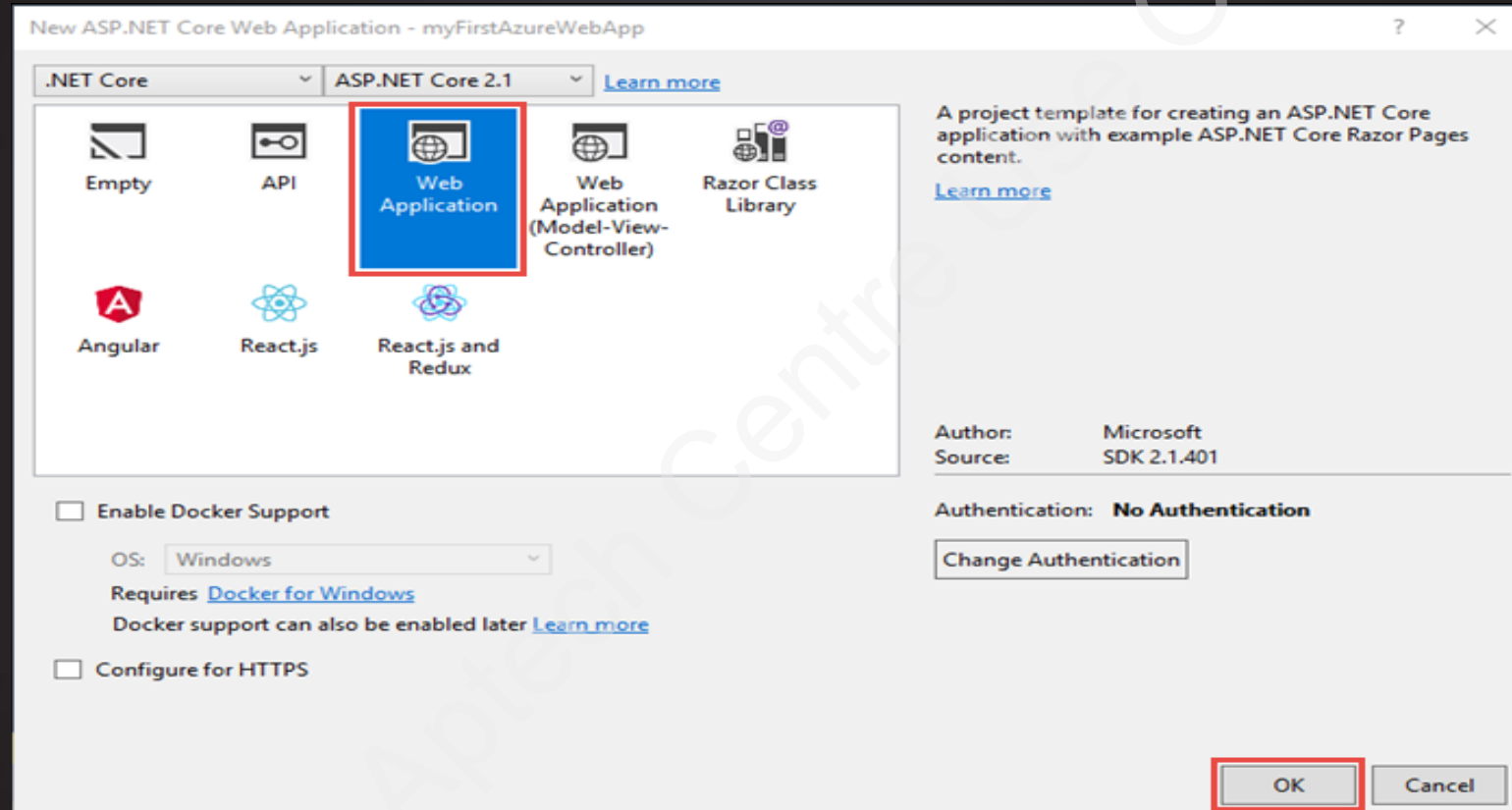
Deployment of Azure Web Applications [1-4]

- Once a Web application is built, it is required to deploy the application in production environment.
- It is recommended to use Microsoft Azure, to surpass the challenges of deployment like configuring databases, assemblies, IIS settings, certificates, and other configuration details.

Create ASP.NET Web App

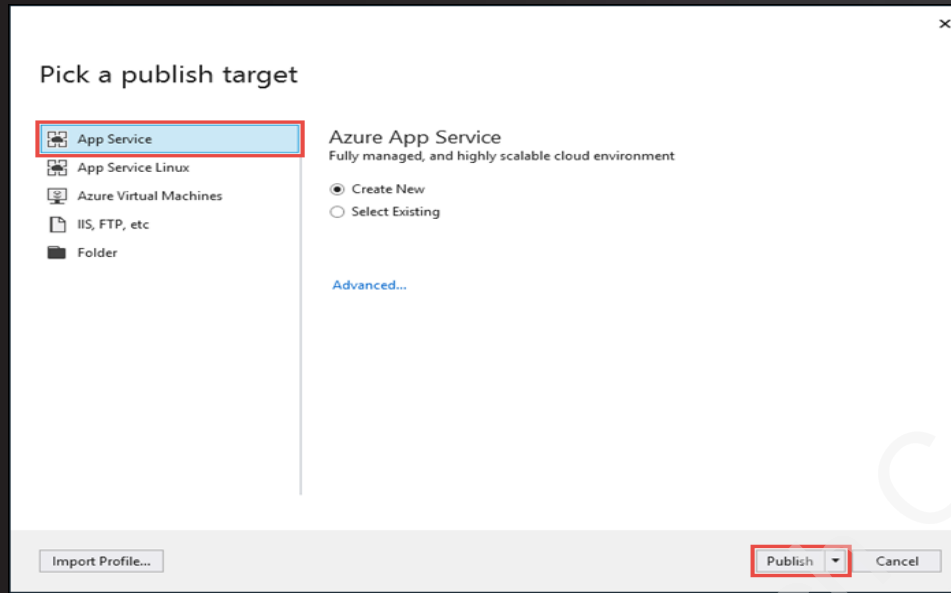


Deployment of Azure Web Applications [2-4]

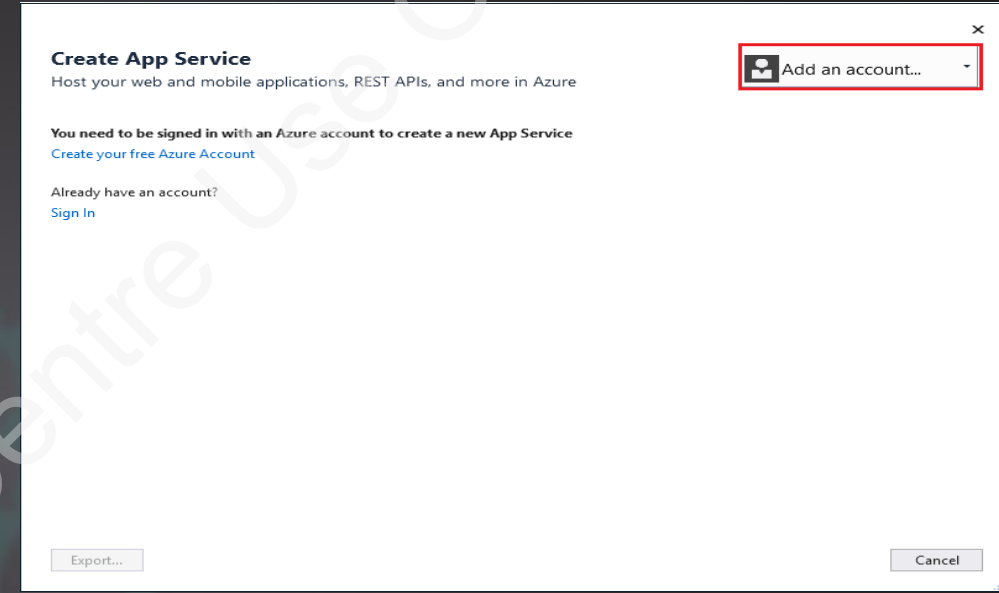


New ASP.NET Core Web Application

Deployment of Azure Web Applications [3-4]



Publish Wizard



Create App Service Dialog Box

Deployment of Azure Web Applications [4-4]

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

Microsoft account

App Name
FirstAzureWebApp2018

Subscription
Visual Studio Ultimate with MSDN

Resource Group
myResourceGroup (westeurope) [New...](#)

Hosting Plan
myAppServicePlan* (West Europe, F1) [New...](#)

Application Insights
West Europe

Export...

Create Cancel

Explore additional Azure services

- Create a SQL Database
- Create a storage account

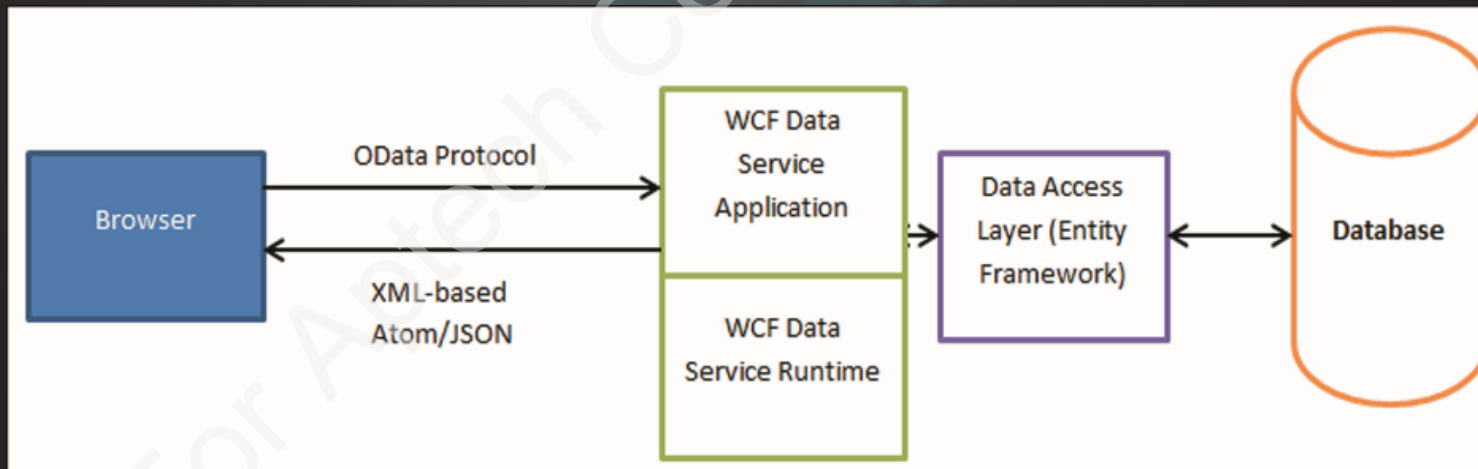
Clicking the Create button will create the following Azure resources

- Hosting Plan - myAppServicePlan
- App Service - FirstAzureWebApp2018

Hosting Options in Create App Service Dialog Box

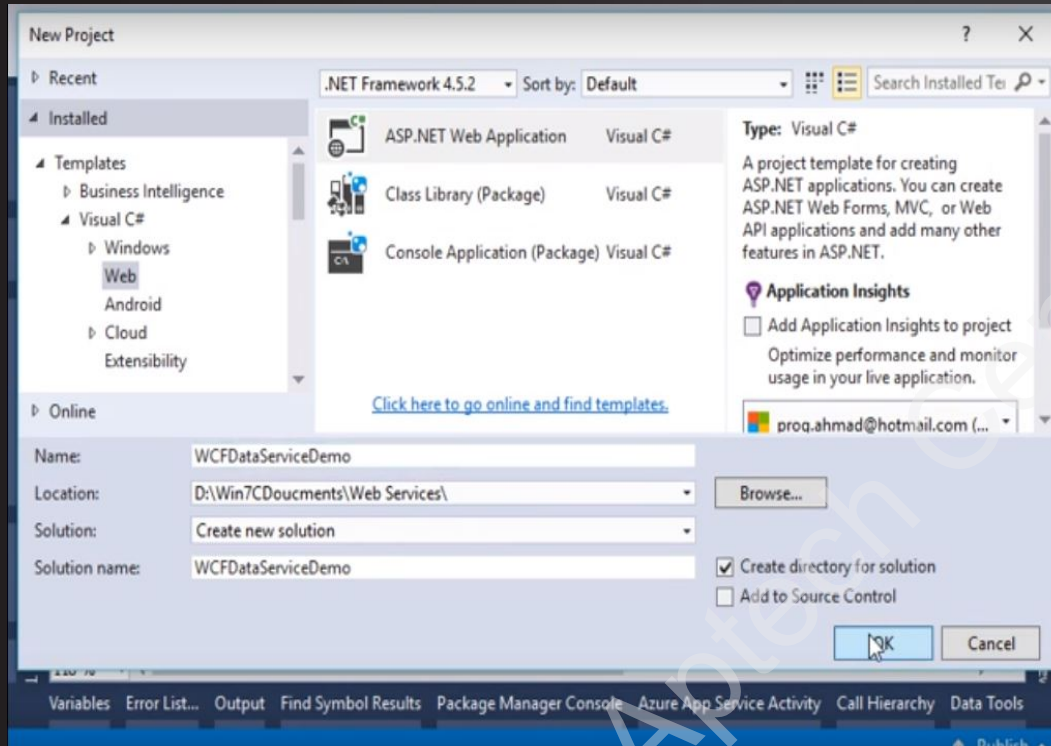
Deploying WCF Data Services [1-10]

- WCF data services are used to create services using the Open Data (OData) protocol.
- OData is a REST-based protocol that allows to perform CRUD operations on resources through URLs using standard HTTP methods.
- The result returned by the WCF data service can be in XML-based Atom and JSON formats.

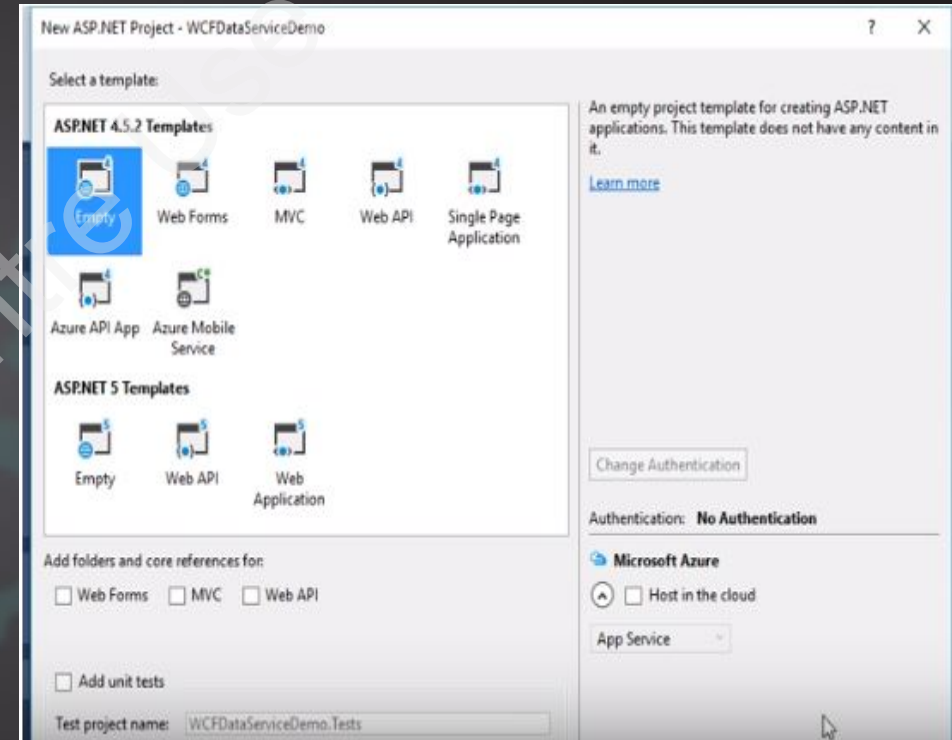


WCF Data Services

Deploying WCF Data Services [2-10]

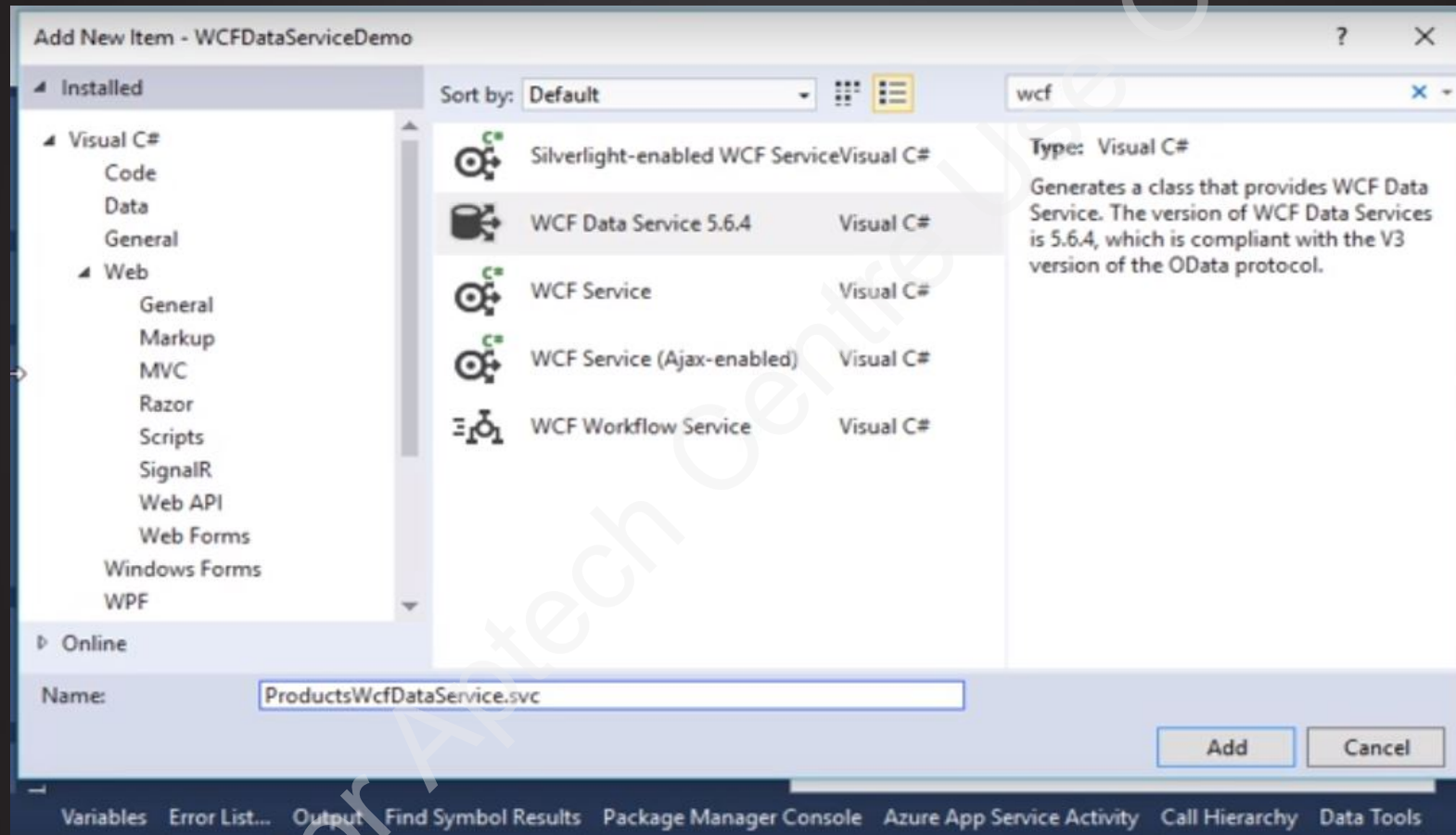


Create ASP.NET Web Application

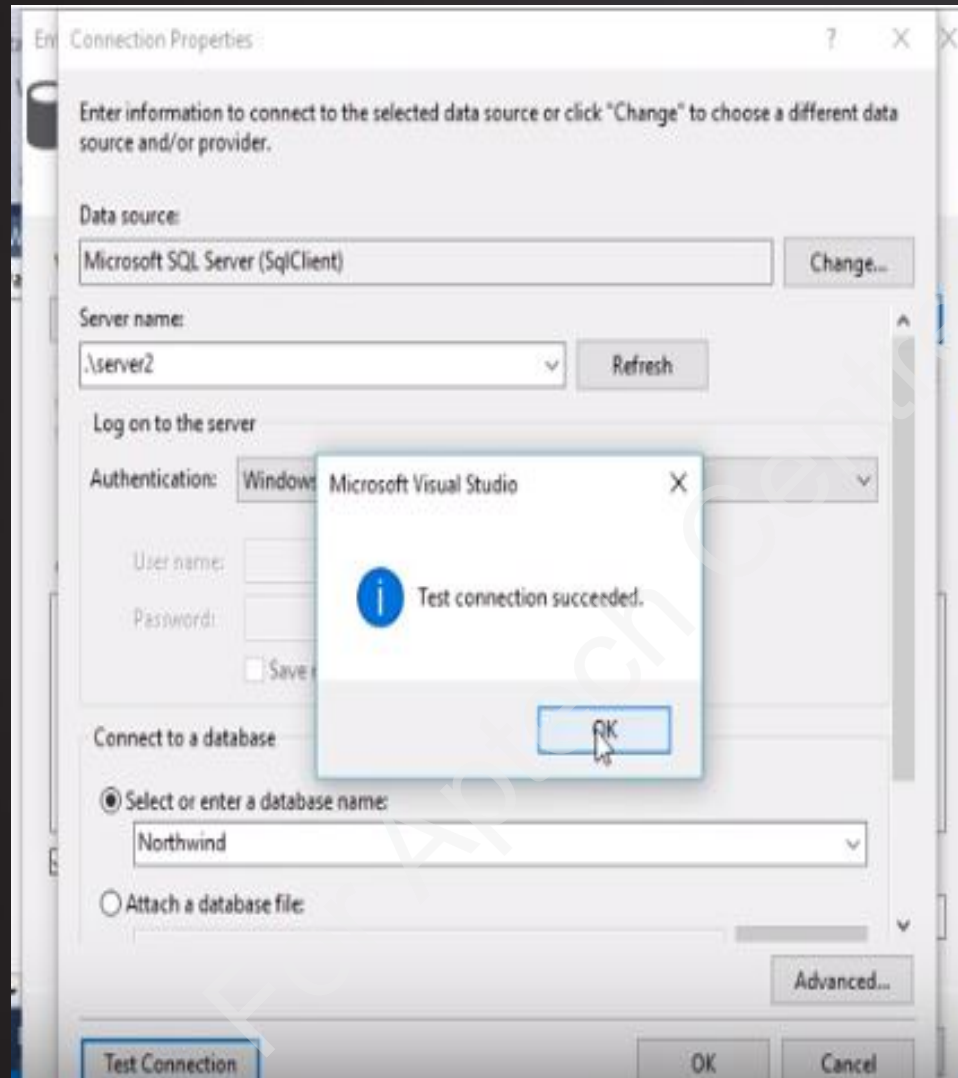


New Project Template Wizard

Deploying WCF Data Services [3-10]

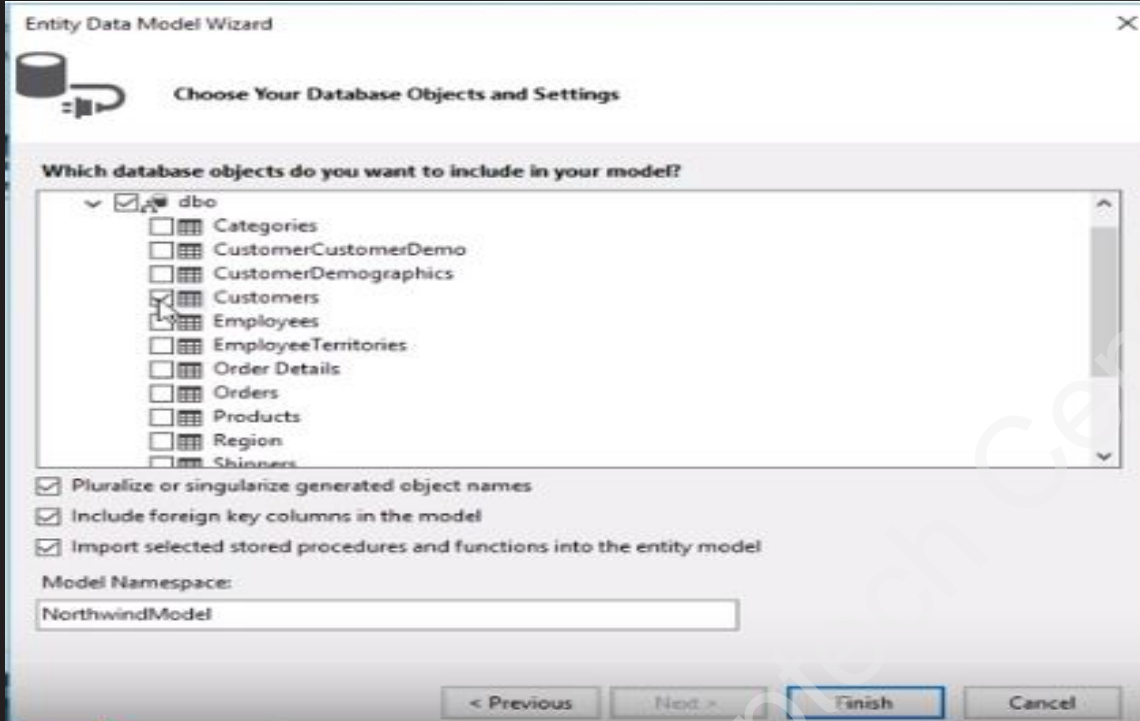


Deploying WCF Data Services [4-10]

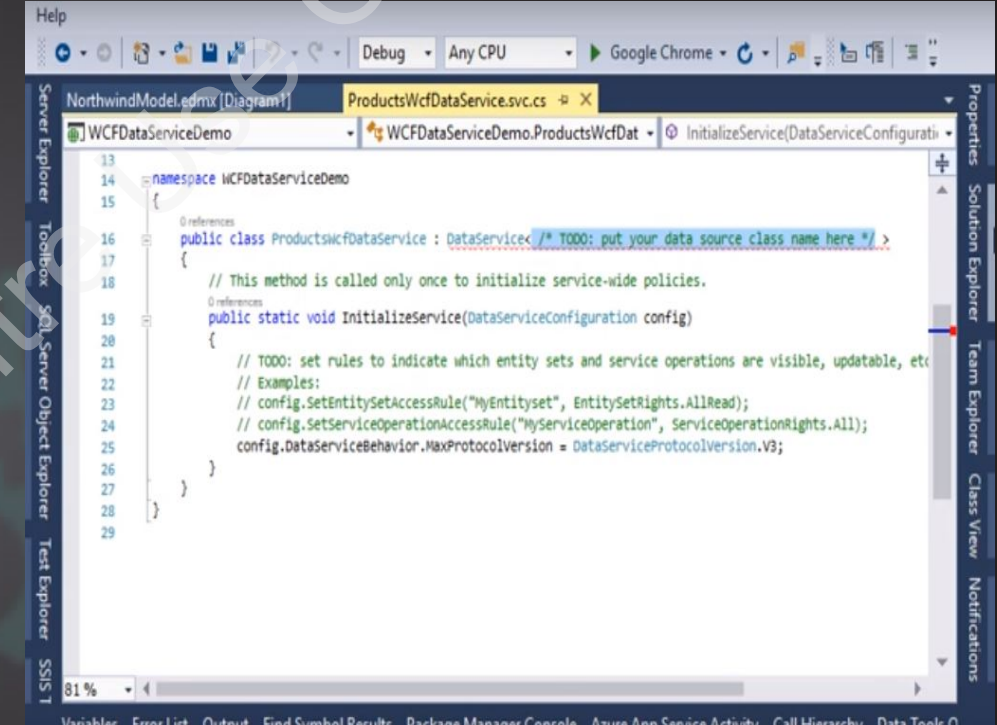


Testing the Connection

Deploying WCF Data Services [5-10]



Choose Database Objects

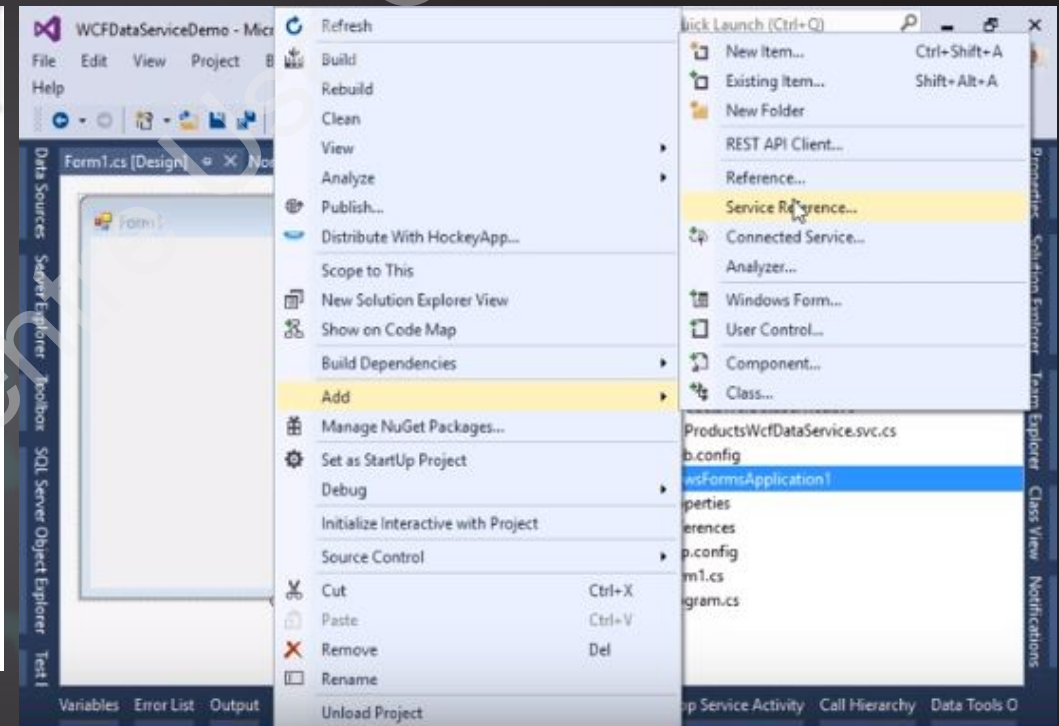


Change Service Model Name in Code

Deploying WCF Data Services [6-10]

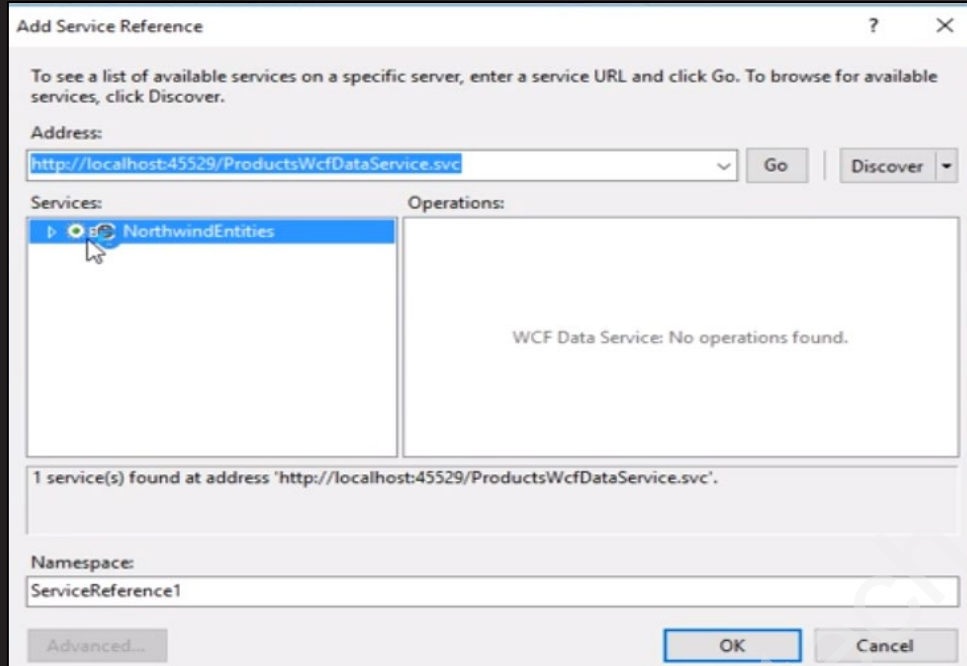


View WCF Data in Browser

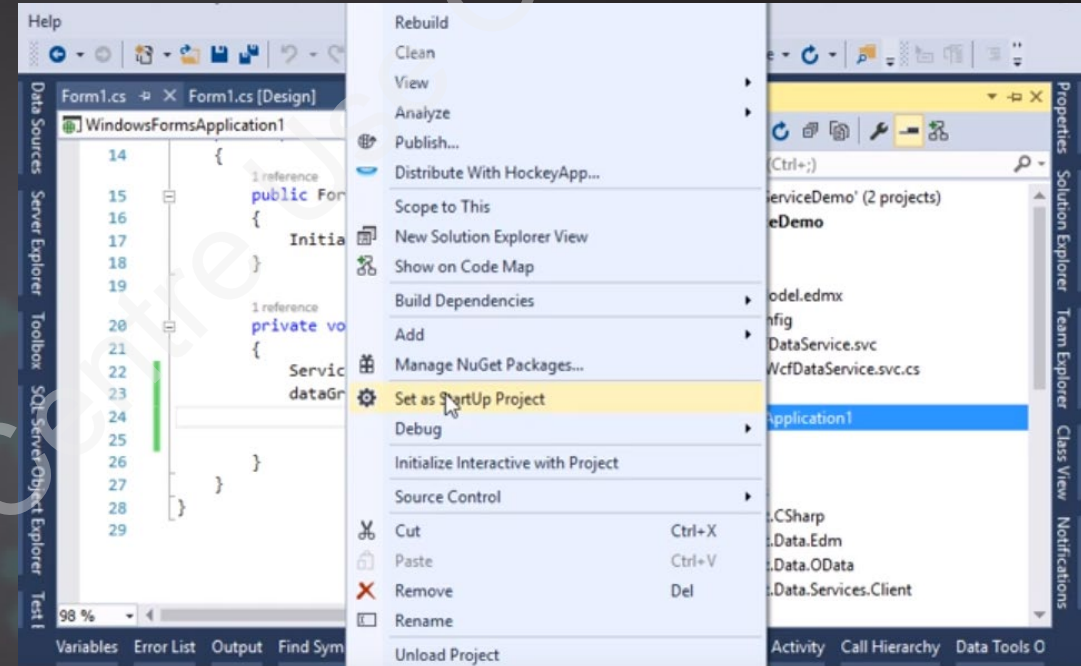


Add New Service Reference

Deploying WCF Data Services [7-10]



Add Service Reference Dialog

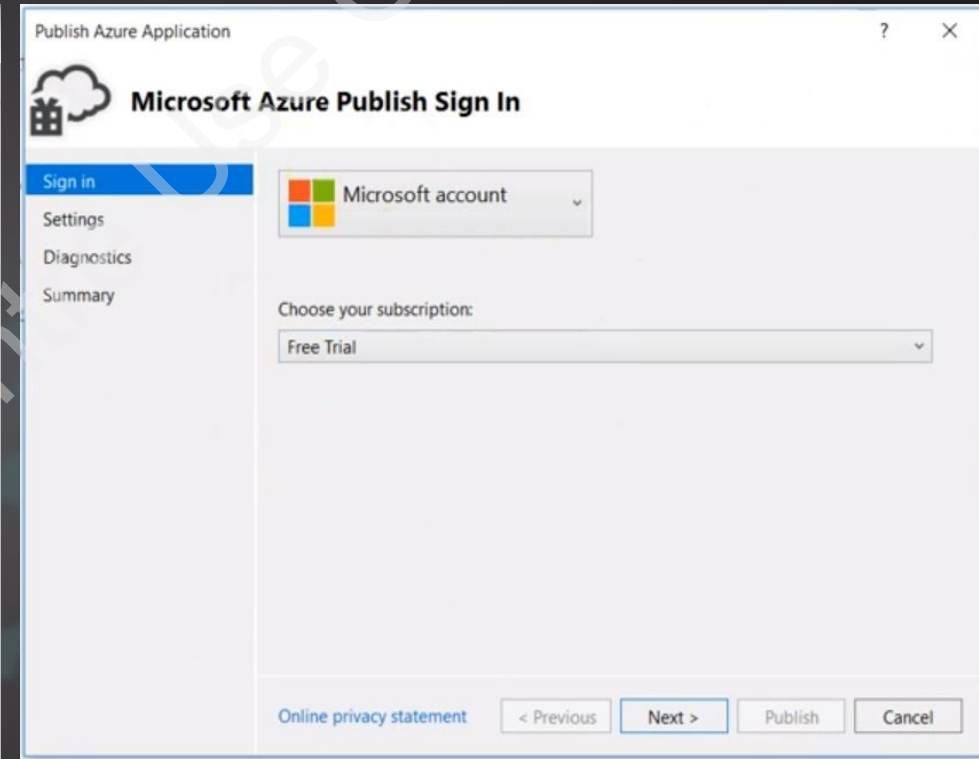


Setting Up As Startup Project

Deploying WCF Data Services [8-10]

```
<?xml version="1.0" encoding="utf-8"?><feed xml:base="http://localhost:45529/ProductsWcfDataService.svc/"
xmlns="http://www.w3.org/2005/Atom" xmlns:d="http://schemas.microsoft.com/ado/2007/08/dataservices"
xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata">
<id>http://localhost:45529/ProductsWcfDataService.svc/Products</id><title type="text">Products</title>
<updated>2017-08-21T06:57:19Z</updated><link rel="self" title="Products" href="Products" /><entry>
<id>http://localhost:45529/ProductsWcfDataService.svc/Products(1)</id><category
term="NorthwindModel.Product" scheme="http://schemas.microsoft.com/ado/2007/08/dataservices/scheme" /><link
rel="edit" title="Product" href="Products(1)" /><title /><updated>2017-08-21T06:57:19Z</updated><author>
<name /></author><content type="application/xml"><m:properties><d:ProductID
m:type="Edm.Int32">1</d:ProductID><d:ProductName>Test T</d:ProductName><d:SupplierID
m:type="Edm.Int32">1</d:SupplierID><d:CategoryID m:type="Edm.Int32">1</d:CategoryID><d:QuantityPerUnit>10
boxes x 20 bags</d:QuantityPerUnit><d:UnitPrice m:type="Edm.Decimal">18.0000</d:UnitPrice><d:UnitsInStock
m:type="Edm.Int16">34</d:UnitsInStock><d:UnitsOnOrder m:type="Edm.Int16">0</d:UnitsOnOrder><d:ReorderLevel
m:type="Edm.Int16">10</d:ReorderLevel><d:Discontinued m:type="Edm.Boolean">>false</d:Discontinued>
</m:properties></content></entry><entry><id>http://localhost:45529/ProductsWcfDataService.svc/Products(2)
</id><category term="NorthwindModel.Product"
scheme="http://schemas.microsoft.com/ado/2007/08/dataservices/scheme" /><link rel="edit" title="Product"
href="Products(2)" /><title /><updated>2017-08-21T06:57:19Z</updated><author><name /></author><content
type="application/xml"><m:properties><d:ProductID m:type="Edm.Int32">2</d:ProductID>
<d:ProductName>Chang</d:ProductName><d:SupplierID m:type="Edm.Int32">1</d:SupplierID><d:CategoryID
m:type="Edm.Int32">1</d:CategoryID><d:QuantityPerUnit>24 - 12 oz bottles</d:QuantityPerUnit><d:UnitPrice
m:type="Edm.Decimal">19.0000</d:UnitPrice><d:UnitsInStock m:type="Edm.Int16">12</d:UnitsInStock>
<d:UnitsOnOrder m:type="Edm.Int16">40</d:UnitsOnOrder><d:ReorderLevel
m:type="Edm.Int16">25</d:ReorderLevel><d:Discontinued m:type="Edm.Boolean">>false</d:Discontinued>
</m:properties></content></entry><entry><id>http://localhost:45529/ProductsWcfDataService.svc/Products(3)
</id><category term="NorthwindModel.Product"
scheme="http://schemas.microsoft.com/ado/2007/08/dataservices/scheme" /><link rel="edit" title="Product"
href="Products(3)" /><title /><updated>2017-08-21T06:57:19Z</updated><author><name /></author><content
```

Output of WCF Data Service



Azure Publish Sign in Wizard

Deploying WCF Data Services [9-10]

The screenshot shows the 'Microsoft Azure Publish Settings' dialog box. The 'Settings' tab is selected. A sub-dialog titled 'Create Cloud Service and Storage Account' is open, showing the following fields:

- Microsoft account:** A dropdown menu with the Microsoft logo.
- Subscription:** A dropdown menu with 'Free Trial' selected.
- Name:** A text box containing 'wcf'.
- Region or Affinity Group:** A dropdown menu.
- Replication:** A dropdown menu with 'Geo-Redundant' selected.

At the bottom of the sub-dialog are 'Create' and 'Close' buttons. The main dialog has a sidebar with 'Sign in', 'Settings', 'Diagnostics', and 'Summary'. At the bottom of the main dialog are 'Online privacy statement', '< Previous', 'Next >', 'Publish', and 'Cancel' buttons.

Setting Dialog Box

The screenshot shows the 'Microsoft Azure Publish Settings' dialog box. The 'Settings' tab is selected. A sub-dialog titled 'Remote Desktop Configuration' is open, showing the following fields:

- User name:** A text box.
- Password:** A text box.
- Confirm password:** A text box.
- Account expiration date:** A date picker showing '29-11-2018'.

At the bottom of the sub-dialog are 'More Options', 'OK', and 'Cancel' buttons. The main dialog has a sidebar with 'Sign in', 'Settings', 'Diagnostics', and 'Summary'. At the bottom of the main dialog are 'Online privacy statement', '< Previous', 'Next >', 'Publish', and 'Cancel' buttons.

Remote Desktop Configuration

Summary

- Azure App Services provides enterprise-grade development and deployment environment for both Windows-based environments and Linux-based environments.
- Azure App Services can be used to develop variety of services such as Web App, Mobile App, Logic App, and API App.
- An App Service plan provides resources for a Web app to run in Azure App Service.
- Azure App can be subscribed in different compute plans such as Shared compute, dedicated compute, and Isolated.
- Azure App Service offers everything that is required for creating mobile back-ends, Websites, and Web APIs for any device or platform.
- The .NET Framework provides WCF data services that developers can use to create services using OData protocol.