

Cloud-based Development with Microsoft Azure and Office 365

Developing and Deploying Custom Solutions in the Microsoft Cloud

|  |  |
| --- | --- |
| Course Code | CBD365 |
| Audience | Professional Developers |
| Format | Instructor led training with hands-on labs |
| Length | 5 Days |
| Course Description | Cloud-based Development with Microsoft Azure and Office 365 is an intensive five-day training course that teaches students how to create custom software solutions in the Microsoft cloud using Microsoft Azure, Office 365 and SharePoint Online. Students will learn how to develop and deploy Azure applications, SharePoint-hosted add-ins, provider-hosted add-ins as well as custom software solutions for SharePoint Online using JavaScript Injection and Remote Provisioning.  This course examines developing client-side solutions using the single page application (SPA) model and emphasizes best practices such as using JavaScript, jQuery and the AngularJS framework. The course also examines the development of server-side applications using ASP.NET MVC and Web API 2. Students will learn how to authenticate and to execute calls against essential APIs such as the Azure Storage APIs, the SharePoint REST API, Client-side Object Model (CSOM) and the Azure Active Directory authorization (ADAL) library. Students will also learn how and when to use the Azure Graph API, the Microsoft Graph API and the Office Graph API.  Each student attending this course will require either a trial subscription or a paid subscription for Microsoft Azure. By working through the hands-on lab exercises of this course, students will become comfortable working in the Microsoft Azure environment developing and deploying custom solutions using Web Apps, Web Jobs, Virtual Machines and Cloud Services. Students will also learn how to develop custom solutions using essential Azure services such as Azure storage and Azure Active Directory Authentication Services.  The course is designed to assist students in preparing for the following Microsoft certification exam:   * 70-532: Developing Microsoft Azure Solutions |
| Student Prerequisites | Attendees should be in good health and should have professional development experience with Visual Studio, C#, the .NET Framework, ASP.NET, JavaScript and jQuery. It is also recommended that attendees have previous hands-on experience with the SharePoint platform as well as experience using AngularJS and ASP.NET MVC. |

Course Modules

1. Developer Roadmap into the Microsoft Cloud
2. Developing with Azure Web Apps and Web Jobs
3. Implementing Azure Virtual Machines and Cloud Services
4. Developing with Azure Storage and Azure SQL Databases
5. Developing SharePoint-hosted Add-ins for SharePoint Online
6. Programming the SharePoint REST API
7. Developing SharePoint Add-ins using AngularJS
8. Developing Provider-hosted Add-ins for SharePoint Online
9. Programming the Client-side Object Model (CSOM)
10. Extending Provider-hosted Add-ins with Remote Event Receivers
11. Publishing and Installing SharePoint Add-ins in SharePoint Online
12. Extending SharePoint Online using JavaScript Injection
13. Developing Secured Applications with Azure Active Directory
14. Developing Azure Applications using the Microsoft Graph API
15. Developing Secure Custom Web Services in Azure using Web API 2

Course Module Detailed Outline

Module 01: Developer Roadmap into the Microsoft Cloud

This module provides a developer introduction to the essential platform pieces of the Microsoft cloud including Microsoft Azure, Office 365 and SharePoint Online. The module begins with an overview of Microsoft Azure as a developer platform and explains the various types of Azure-based entities that are commonly created when developing and deploying custom applications in the Microsoft cloud. The module examines the integration between Microsoft Azure and Office 365 and explains how every Office 365 organization is implemented as an Office 365 tenancy which is backed by a directory created in Azure Active Directory. The module provides students with an overview of the SharePoint add-in model and discusses the various types of custom software projects that can be used to customize and extend sites in SharePoint Online. The module concludes with a walkthrough of how to set up a cloud-based development environment by creating a trial Office 365 tenancy account and signing up for a Microsoft Azure subscription.

Topics Covered

* Getting Started with Microsoft Azure
* Understanding Azure Service Management versus Azure Resource Manager (ARM)
* Azure Integration with Office 365 and Organizational Tenancies
* Developing Custom Solutions for SharePoint Online
* Developing Applications for Azure Active Directory
* Setting Up a Cloud-based Development Environment

Hands-on Lab: Getting Started with Microsoft Azure and Office 365 Development

* Exercise 1: Get Up and Running with the Student VM
* Exercise 2: Create an Office 365 Trial Tenant Account
* Exercise 3: Sign Up for an Azure Trial Subscription
* Exercise 4: Use PowerShell to Connect to Microsoft Azure and SharePoint Online
* Exercise 5: Write a PowerShell Script to Create New Users in an Office 365 Tenancy

Module 02: Developing with Azure Web Apps and Web Jobs

This module teaches students how to create and configure Azure Web Apps (aka Websites) to develop and deploy custom applications in the Azure environment. Students will learn how to create and configure Web Apps using both Azure Portals as well as with Visual Studio and with PowerShell. The module explains how to roll out staged deployments using Deployment Slots as well as how to configure service plans to scale Web Apps using either a scale-up strategy or a scale-out strategy. The module teaches students how to configure Web App support for custom domains, custom logging, diagnostics and monitoring. The module concludes by explaining how to create and implement Web Jobs which can execute custom logic from within the Microsoft cloud using a PowerShell script or a C# application that runs on a timer schedule or in a continuous or an on-demand fashion.

Topics Covered

* Creating and Configuring Web Apps
* Rolling Out Staged Deployments using Deployment Slots
* Configuring Web App for Custom Logging, Diagnostics & Monitoring
* Scaling Web Apps using a Scale-up Strategy versus a Scale-out Strategy
* Executing Custom Logic from the Microsoft Cloud using Web Jobs

Hands-on Lab: Working with Azure Web Apps and Web Jobs

* Exercise 1: Create and Configure Web Apps using the Azure Portal
* Exercise 2: Deploy an ASP.NET Application to a Web App using Visual Studio
* Exercise 3: Roll Out a Staged Deployment using Deployment Slots
* Exercise 4: Execute Custom Logic from inside the Microsoft Cloud using Web Jobs

Module 03: Implementing Azure Virtual Machines and Cloud Services

This module teaches students how to take advantage of the Infrastructure-as-a-Service (IaaS) support in Microsoft Azure for creating and connecting to Virtual Machines (VMs). Students will learn about the various Workloads supported by VM images available in the Microsoft Marketplace as well as how to convert and upload local Hyper-V images for use in Microsoft Azure. Students will learn how to configure support for VM management tools such as Custom Script Extension and PowerShell Desired State Configuration (DSC). Students will learn how to configure a virtual network to establish connections between VMs which need to communicate with one another in a custom solution. The final section of this module explains how Microsoft Azure provides Platform-as-a-Service (PaaS) support through Cloud Services which can be used abstract away many of the low-level details of configuring the security, connectivity and availability within a complex set of VM images which are being used to implement a single custom solution.

Topics Covered

* Understanding the Workloads Supported by VM Images in the Microsoft Marketplace
* Understanding How Azure VM Storage Works
* Creating and Connecting to an Azure Virtual Machine
* Converting and Uploading Local Hyper-V Images for use in Microsoft Azure
* Understanding Cloud Services, Load Balancing and Availability Sets
* Developing a Cloud Service with Web Roles and Worker Roles using Visual Studio

Hands-on Lab: Implementing Azure Virtual Machines and Cloud Services

* Exercise 1: Create a New VM Image from the Microsoft Marketplace
* Exercise 2: Access a Virtual Machine using PowerShell and Remote Desktop
* Exercise 3: Create a Cloud Service with Visual Studio
* Exercise 4: Deploy a Cloud Service to Microsoft Azure

Module 04: Developing with Azure Storage and Azure SQL Databases

This module explains how Microsoft Azure offers Platform-as-a-Service (PaaS) support for storing and retrieving data through Azure Storage and Azure SQL Databases. The module begins with an overview of Azure Storage and explains why it is required to create an Azure Storage account and to generate shared access keys. Students will learn how to store and retrieve unstructured data by creating containers in order to program using Azure Storage blobs. The module also explains how to develop applications which create and leverage Azure Storage tables and Azure Storage queues. The final section of the module teaches students how to create and develop custom applications against a SQL Azure Database as well as how to scale and manage SQL Azure Databases in a production scenario.

Topics Covered

* Overview of Azure Storage
* Programming with Azure Storage Blobs
* Programming with Azure Storage Tables
* Programming with Azure Storage Queues
* Creating and Configuring an Azure SQL Database
* Developing a Custom Application against an Azure SQL Database

Hands-on Lab: Developing with Azure Storage and Azure SQL Databases

* Exercise 1: Create an Azure Storage Account
* Exercise 2: Write Code to Read and Write Azure Storage Blobs
* Exercise 3: Program using Azure Storage Tables
* Exercise 4: Program using Azure Storage Queues
* Exercise 5: Develop an Application with an Azure SQL Database

Module 05: Developing SharePoint-hosted Add-ins for SharePoint Online

This module begins by examining the architecture of the SharePoint add-in model and discusses the differences between SharePoint-hosted add-ins and provider-hosted add-ins. After introducing the SharePoint add-in model, the module focuses on developing SharePoint-hosted add-ins and explains why SharePoint Online creates a new app web within its own isolated DNS domain each time a SharePoint-hosted add-in is installed. The module examines user interface design issues with SharePoint-hosted add-ins such as whether to create pages using ASPX files versus using HTML files and whether to design a SharePoint-hosted add-in project as a multipage application or a single page application (SPA). The module also demonstrates how to extend the user interface of the host web with a SharePoint-hosted add-in by creating custom add-in parts (formerly app parts) and user custom actions.

Topics Covered

* Understanding the Architecture of the SharePoint Add-in Model
* Developing SharePoint-hosted Add-in with Visual Studio
* Designing the User Interface of a SharePoint-hosted Add-in
* Developing Add-in Parts
* Adding User Custom Actions

Hands-on Lab: Developing SharePoint-hosted Add-ins

* Exercise 1: Create a New Developer Site for Testing
* Exercise 2: Create and Debug a SharePoint-hosted Add-in
* Exercise 3: Create the Hello World App Part
* Exercise 4: Create an App Part with Custom Properties

Module 06: Programming the SharePoint REST API

This module begins with a quick primer on the fundamentals of REST and the OData protocol. Next, the module examines the architecture and the functionality of the SharePoint REST API provided by SharePoint Online. Students will learn how to formulate REST URIs which target SharePoint objects such as sites, lists and list items and how to execute asynchronous REST API calls using the jQuery library. The module steps through how to use the SharePoint REST API to implement the full range of CRUD behavior in a SharePoint-hosted add-in by creating, reading, updating and deleting items in a SharePoint list. The module demonstrates how to implement paging with SharePoint list items using skip tokens returned by the SharePoint REST API. Students will also learn advanced OData programming techniques in SharePoint Online for dealing with the request digest and using verbose metadata and eTags to implement optimistic concurrency when performing updates.

Topics Covered

* Understanding REST and ODATA
* Constructing URLs for The SharePoint REST API
* Programming the SharePoint REST API
* Paging SharePoint List Items
* Modifying SharePoint List Items

Hands-on Lab: Programming with the SharePoint REST API

* Exercise 1: Get the SharePointCRM Starter Project Up and Running
* Exercise 2: Query a SharePoint List using the SharePoint REST API
* Exercise 3: Add and Delete Items with the SharePoint REST API
* Exercise 4: Update Existing Items with the SharePoint REST API

Module 07: Developing SharePoint Add-ins using AngularJS

This module provides a quick and intensive introduction to development using the AngularJS framework. Students will learn how to use AngularJS to develop a SharePoint-hosted add-in that is designed as a single page applications (SPA). The module explains what features the AngularJS framework provides as well as in which scenarios it makes sense to use them. Students will learn essential AngularJS concepts and techniques such as creating a custom Angular routing scheme using custom controllers and custom views. Students will also learn the AngularJS best practice of separating data access code which calls to the SharePoint REST API by creating a custom AngularJS service. The module concludes by discussing the forthcoming releases of AngularJS 2.0 and ECMAScript 6.0 and what developers can expect when migrating development projects created using AngularJS 1.0.

Topics Covered

* Introduction to AngularJS
* Angular Modules and Code Injection
* Angular Routes, Views and Controllers
* Programming with the Built-in Angular Services
* Creating a Custom Angular Service
* What's Coming Next with AngularJS 2.0

Hands-on Lab: Creating a SharePoint-hosted App using AngularJS

* Exercise 1: Creating a SharePoint-hosted App using Bootstrap and AngularJS
* Exercise 2: Working with Views, Controllers and Routing
* Exercise 3: Extending the AngularCRM Project with a Custom Service

Module 08: Developing Provider-hosted Add-ins with MVC

The module begins by examining the provider-hosted add-in architecture and discussing the differences between implementing a provider-hosted add-in with ASP.NET in Visual Studio using WebForms versus using the MVC framework. The module discusses how to design and implement a provider-hosted add-in as a SPA which deploys client-side JavaScript code to the remote web which is able to program against the host web using the Cross Domain Library. The module demonstrates how to develop provider-hosted add-ins with ASP.NET MVC which contain server-side code which implement external authentication against Azure ACS using the SharePointContext class and the TokenHelper class to acquire and manage access tokens.

Topics Covered

* Architecture of a Provider-hosted Add-in
* Developing with Web Forms Versus the MVC Framework
* Working with the Cross Domain Library
* Developing Provider-hosted Add-ins using ASP.NET MVC
* Implementing External Authentication using SharePointContext and TokenHelper

Hands-on Lab: Developing Provider-hosted Add-ins using MVC

* Exercise 1: Create a Provider-Hosted Add-in that uses the MVC Framework
* Exercise 2: Track SharePoint Session State in a Provider-Hosted Add-in
* Exercise 3: Create a Custom Database using the Entity Framework
* Exercise 4: Create a Strongly-typed Controller using the MVC Framework

Module 09: Programming the Client-side Object Model (CSOM)

This module introduces students to programming with the Client-side Object Model (CSOM) in SharePoint Online. Emphasis will be placed on using CSOM when writing server-side C# code to implement the remote web of a provider-hosted add-in. Student will learn the various ways in which CSOM can be used to authenticate the user and to authenticate the SharePoint add-in itself. Students will learn CSOM programming techniques for provisioning site columns, content types, lists and document libraries in the host web. Along the way, students will also learn how to optimize CSOM calls across the network and how to utilize CSOM's support for remote exception handling when executing CSOM commands in batches.

Topics Covered

* Understanding CSOM Architecture
* Writing Code to Authenticate Users and Add-ins
* Optimizing CSOM Code
* Remote Exception Handling
* Creating Site Columns, Content Types and Lists

Hands-on Lab: Programming a Provider-hosted App using the CSOM

* Exercise 1: Create a Provider-hosted App that Uses CSOM
* Exercise 2: Program CSOM to Query the Set of Lists in the Host Web
* Exercise 3: Using CSOM to Create a List in the Host Web

Module 10: Extending Provider-hosted Add-ins with Remote Event Receivers

This module discusses when and how to use remote event receivers in the design of a provider-hosted add-in. The module discusses the architectural differences between "before events" which are modeled as two-way events and "after events" which are modeled as one-way events. Students will learn how to configure and implement an App-Installed event handler which can be used to provision lists, document libraries and other types of assets in the host web during the installation of a provider-hosted add-in. Students will learn how to configure debugging support for remote event receivers in the SharePoint Online environment by registering an Azure service bus endpoint and integrating that service bus endpoint with a Visual Studio project. The module also demonstrates a CSOM programming technique which can be used to register a remote event receiver dynamically so it's possible to wire up remote event handlers to lists and document libraries in the host web.

Topics Covered

* Reviewing the SharePoint Classic Event Model
* Understanding Remote Event Receiver Architecture
* Configuring Debugging Support
* Handling App Lifecycle Events
* Registering Event Receivers in the Host Web

Hands-on Lab: Developing SharePoint Add-ins with Remote Event Receivers

* Exercise 1: Create an Azure Service Bus Endpoint for Debugging
* Exercise 2: Create A Remote Event Receiver for the AppInstalled event

Module 11: Publishing and Installing SharePoint Add-ins in SharePoint Online

This module discusses how to manage the lifecycle of Office add-ins and SharePoint add-ins in the Office 365 environment. Students will learn how to publish Office add-ins and SharePoint add-ins using the App Catalog site. Students will also learn the various ways in which add-ins can be installed by standard users and by tenant administrators. The module explains the differences between installing a SharePoint add-in at site scope versus tenancy scope. Students will also learn how to upgrade SharePoint-hosted add-ins using feature upgrade techniques to replace files in the add-in web containing HTML, CSS and JavaScript. The module concludes by discussing the steps involved with deploying a provider-hosted add-in and its associated Entity Framework database in the Office 365 environment which involves creating a new Azure Web App and a SQL Azure Database to host the remote web and a custom database in the Azure cloud.

Topics Covered

* Creating the App Catalog Site in SharePoint Online
* Publishing SharePoint Add-ins to the App Catalog
* Installing SharePoint Add-ins at Site Scope versus Tenancy Scope
* Upgrading SharePoint-hosted Add-ins
* Deploying Provider-hosted Add-ins using an azure Web App

Hands-on Lab: App Deployment, Installation and Upgrade

* Exercise 1: Setup Lab Environment
* Exercise 2: Create the App Catalog Site Collection
* Exercise 3: Package and Deploy a SharePoint-Hosted App
* Exercise 4: Install a SharePoint Hosted App at the Tenancy Scope
* Exercise 5: Deploy an App Update

Module 12: Extending SharePoint Online using JavaScript Injection

The module begins by explaining the central concept of JavaScript injection where scripts containing custom JavaScript code are uploaded to SharePoint Online and executed using the permissions of the current user. Students will learn to get started with JavaScript injection using the Script Editor Web Part. Next, students will learn how to extend a SharePoint site by adding new pages and custom JavaScript code which makes use of the JavaScript Object Model (JSOM) and the SharePoint REST API. The module then explains how to use remote provisioning to effectively deploy a custom solution which uses JavaScript injection. Along the way, the module discusses loading JavaScript library dependencies as well as how to execute custom JavaScript code with user custom actions such as ribbon buttons, ECB menu items and ScriptLinks as well as how to use client-side rendering rendering techniques and JSLink to create customized views for SharePoint lists.

Topics Covered

* Enabling Scripting in Office 365.
* Understanding JavaScript Injection
* JSOM Programming
* Remote Provisioning using CSOM
* Designing MDS-enabled Pages
* JSLink and Client-side Rendering

Hands-on Lab: Extending SharePoint Online using JavaScript Injection

* Exercise 1: Use CSOM to Achieve Remote Provisioning
* Exercise 2: Create a Custom Solution which uses JavaScript Injection
* Exercise 3: Display SharePoint List Data using JSLink and Client-side Rendering

Module 13: Developing Secured Applications with Azure Active Directory

This module provides an overview of developing applications which leverage Azure Active Directory (AD) authentication services and the Azure AD security model which involves user authentication, application authentication and an authorization scheme based on configurable permissions. The module explains the differences between application permissions and delegated permissions as well as how permissions are granted to an application using the Common Consent framework. The module discusses when to use single-tenant applications versus multitenant applications and demonstrates how to develop Azure AD applications which support common OAuth 2.0 authentication flows using authorization codes, client credentials and implicit flows.

Topics Covered

* Understanding OAuth 2.0 and OpenID Connect
* Developing Applications for Azure Active Directory
* Understanding Azure Active Directory Authentication Flows
* Developing with the Active Directory Authentication Library for .NET
* Implementing OpenID Connect Authentication using the OWIN Framework
* Developing SPAs with the Active Directory Authentication Library for JavaScript

Hands-on Lab: Security Programming with Azure Active Directory

* Exercise 1: Create an Azure Application using Visual Studio and ASP.NET MVC
* Exercise 2: Implement the Authorization Grant Flow using the OWIN Framework
* Exercise 3: Program Against Azure AD using the Azure Graph API
* Exercise 4: Create an SPA using the Active Directory Authentication Library for JavaScript

Module 14: Developing Azure Applications using the Microsoft Graph API

This module introduces the Microsoft Graph API and discusses how this API serves to abstract away the divisions between Azure Active Directory, Exchange Online and SharePoint Online to provide a single unified endpoint for general application development. Students will learn how to execute REST calls against the Microsoft Graph API to access data associated with the current Office 365 user such as mail messages and calendar events in Exchange Online and files in OneDrive for Business. The module also demonstrates how to develop .NET applications which leverage the Microsoft Graph client library which provides a productivity-oriented alternative to programming directly against the underlying REST API. The module concludes by demonstrating how to develop pure client-side SPAs which access the Microsoft Graph API with the assistance of the Active Directory Authentication Library for JavaScript.

Topics Covered

* Introduction to the Microsoft Graph API
* Acquiring and Managing Access Tokens for the Microsoft Graph API
* Reading and Write Mail Messages and Calendar Events for the Current User
* Managing Documents in OneDrive for Business for the Current User
* Manage Users and Groups in the Current Office 365 tenancy

Hands-on Lab: Developing with the Microsoft Graph API

* Exercise 1: Create an Azure Application using Visual Studio and ASP.NET MVC
* Exercise 2: Acquire and Manage Access Tokens for the Microsoft Graph API
* Exercise 3: Write Code to Access Mail Messages and Calendar for the Current User
* Exercise 4: Write Code to Manage Users and Groups in the Current Office 365 tenancy

Module 15: Developing Secure Custom Web Services in Azure using Web API 2

This module teaches students how to create, test and debug custom web services using Web API 2 and Visual Studio. Students will learn how to implement RESTful web services as well as how to implement ODATA-compatible web services which support common ODATA query options such as filtering, sorting and inline count. The module examines how to leverage the Web API 2 support for Cross-Origin Resource Sharing (CORS). The module concludes by teaching students how to secure a custom web service created with Web API 2 by integrating it with the identity management features of Azure Active Directory.

Topics Covered

* Understanding Web API Controllers and Call Routing
* Implementing RESTful Web Services
* Implementing OData Web Services
* Adding Support for Cross-Origin Resource Sharing (CORS)
* Secure a Custom Web Service using Azure Active Directory

Hands-on Lab: Developing Custom RESTful Services with Web API

* Exercise 1: Create and Populate the Wingtip CRM Database in SQL Server
* Exercise 2: Create an OData Service using Web API 2
* Exercise 3: Enable CORS Support in the WingtipCRM Service
* Exercise 4: Secure an ODATA Service using Azure Active Directory