

**Web Application Identity and Access with Azure B2C**

**Course Number:** AZB2C-100  
**Duration:** 3 days

**Overview**

This Web Application Identity and Access with Azure B2C training course teaches attendees how to configure Azure B2C and provide authentication and authorization services to web applications. Students learn how to set up a new B2C tenant, register various web applications, configure permissions, and customize UI templates.

**Prerequisites**

There are no prerequisites for the class, but prior experience with the Azure Portal and web applications would be helpful.

**Materials**

All Azure B2C training students receive comprehensive courseware.

**Software Needed on Each Student PC**

* Windows, macOS, or Linux computer
* Latest LTS version of Node.js, .NET SDK, & Python
* Azure subscription (free trial account can be used)
* Modern web browser
* Text Editor (VS Code preferred)

**Objectives**

* Understand how Azure B2C can be used to authenticate and authorize users
* Create and configure a new Azure B2C tenant
* Integrate various kinds of web applications with Azure B2C
* Create and configure User Flows
* Customize User Flow UI templates
* Setup a workflow for custom UI template development
* Work with identity providers
* Secure and audit Azure B2C

**Outline**

* Introduction
  + What is Azure B2C?
  + Azure B2C compared to Azure Active Directory
  + Azure B2C compared to other authentication providers
* Create a B2C Tenant with the Azure Portal
  + Configure the Azure Active Directory resource provider
  + Create a new B2C Tenant
  + Importance of the organization name
  + Relationship between the B2C tenant and the main tenant
  + How to switch directories
  + Configure company branding
  + Configure user attributes
  + Controlling access to the B2C tenant
  + Configure Azure B2C identity providers
* Registering Applications
  + What does application registration accomplish?
  + Configure supported account types
  + Configure the redirect URI
  + Configure permissions
  + Application secrets
  + Securing APIs
* Example Application Registrations
  + React app
  + React Next.js app
  + Asp.net MVC
  + Blazor WASM
  + Blazor WASM ASP.NET hosted
  + Blazor Server
  + Python flask app
* User Flows
  + What is a User Flow?
  + Creating a User Flow
  + Sign up & Sign-In User Flow
  + Sign-in user flow
  + Profile editing user flow
  + Multi-factor authentication
  + Identity providers
  + User attributes
  + Application claims
  + Configure password settings
  + API connectors
* User Flow UI Templates
  + What are UI templates?
  + Configuring UI templates settings for user workflows
  + Enable JavaScript in the UI template
* Custom UI Template Storage
  + Set up a storage account to host custom templates
  + Configure CORS for the storage account
  + Configure permissions for the storage account
  + Uploading templates with Azure Storage Explorer
  + Uploading templates via an NPM script command
  + Setup a CDN to improve Custom Template hosting performance
* Custom UI Template Programming
  + Acquire the default templates provided by Microsoft
  + Template versions
  + How to set up an environment for modifying templates
  + Automate template building and deployment
  + File and page structure
  + Understand which UI functions are AJAX operations and which are post-back operations
  + Detecting AJAX operations
  + Applying CSS
  + Perform DOM manipulations with jQuery
* Security
  + Conditional access policies
  + Risky users
  + Risk detection
  + Audit logs
* Conclusion