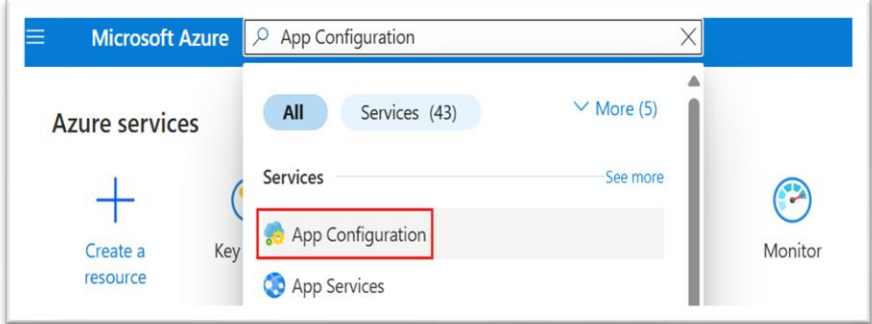
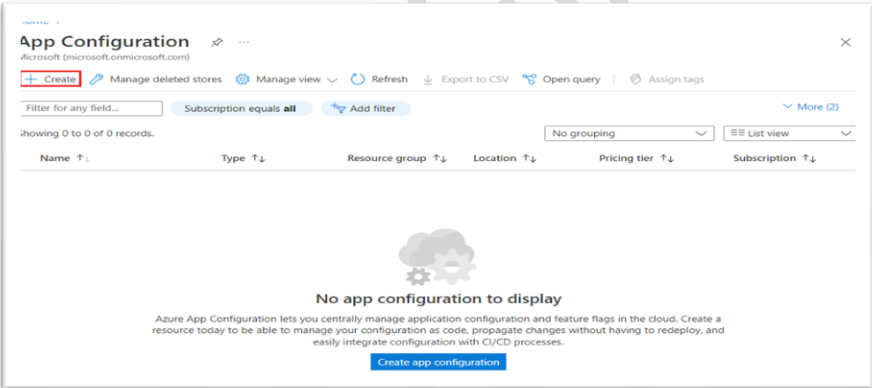


# App Configuration

## 1. Task 1: Create resource and create key-value

Steps	Demonstration
Search <i>App Configuration</i> in homepage search box	 The screenshot shows the Microsoft Azure homepage with a search bar at the top. The search results for 'App Configuration' are displayed, showing 'All' and 'Services (43)' tabs. The 'App Configuration' service is highlighted with a red box. Other services like 'App Services' are also visible.
Select <i>Create or Create app configuration</i>	 The screenshot shows the 'App Configuration' resource page in the Azure portal. The page title is 'App Configuration' with the provider 'Microsoft (microsoft.com/microsoft.com)'. There is a '+ Create' button highlighted with a red box. Below the title, there are tabs for 'Manage deleted stores', 'Manage view', 'Refresh', 'Export to CSV', 'Open query', and 'Assign tags'. A filter bar shows 'Subscription equals all' and 'Add filter'. Below the filter bar, there is a table with columns: Name, Type, Resource group, Location, Pricing tier, and Subscription. The table is currently empty, showing '0 to 0 of 0 records'. A message at the bottom states 'No app configuration to display' and provides instructions on how to create a resource. A 'Create app configuration' button is visible at the bottom.

Fill information and  
create the resource

home > App Configuration >

## Create App Configuration

Basics Networking Tags Review + create

Azure App Configuration provides a service to centrally manage application settings and feature flags. Modern programs, especially programs running in a cloud, generally have many components that are distributed in nature. Spreading configuration settings across these components can lead to hard-to-troubleshoot errors during an application deployment. Use App Configuration to store all the settings for your application and secure their accesses in one place. [Learn more](#)

**Project Details**

Subscription \*

Resource group \*  [Create new](#)

**Instance Details**

Location \*

Resource name \*   
Resource names are reserved for a period of time after deletion. [Learn more](#)

Pricing tier (View pricing details) \*

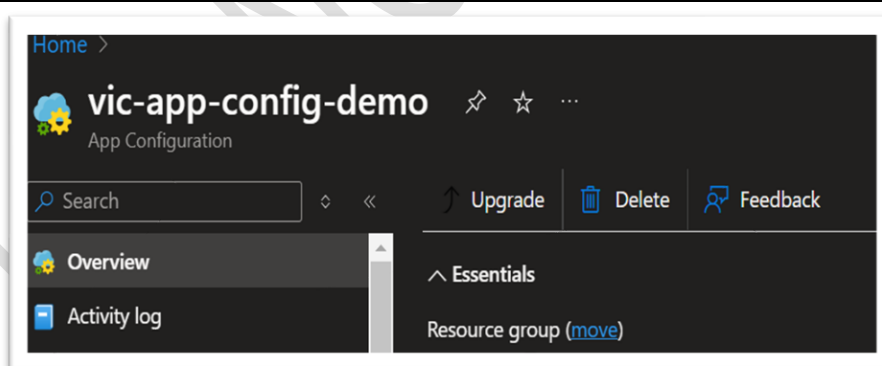
**Geo-replication (preview)**

Use the geo-replication feature to create replicas in other locations of your current configuration store for enhanced resiliency and availability. Additionally, having multi-region replicas lets you better distribute load, lower latency, protect against datacenter outages, and compartmentalize globally distributed workloads.

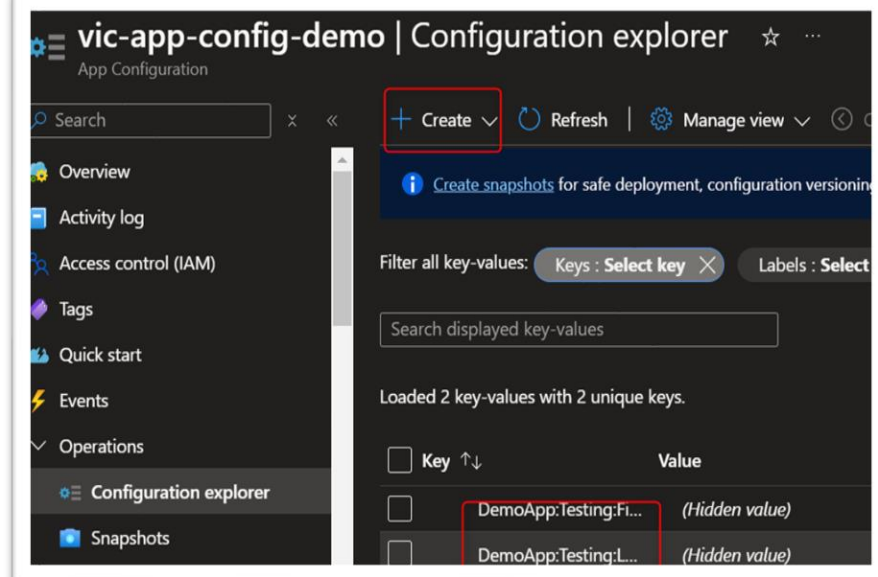
Create replicas (View pricing details) ☐

[Review + create](#) < Previous Next: Networking >

Resources created



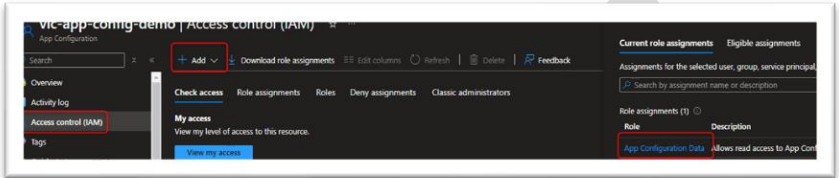
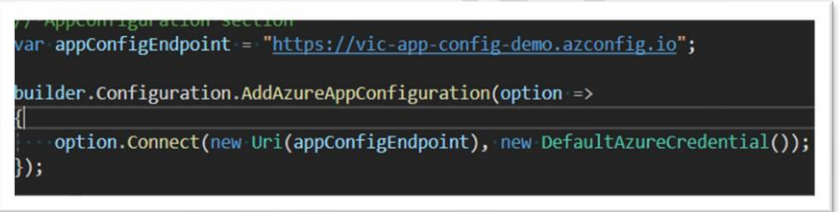
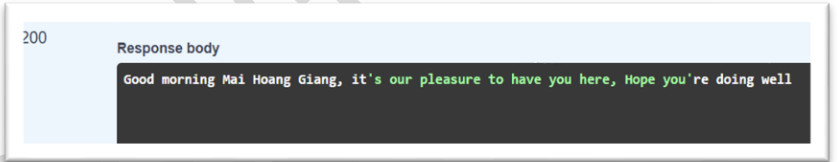
Create Key Value:  
Select *Operations* >  
*Configuration explorer* > *Create* >  
*Key-value* to add a  
key-value to a store



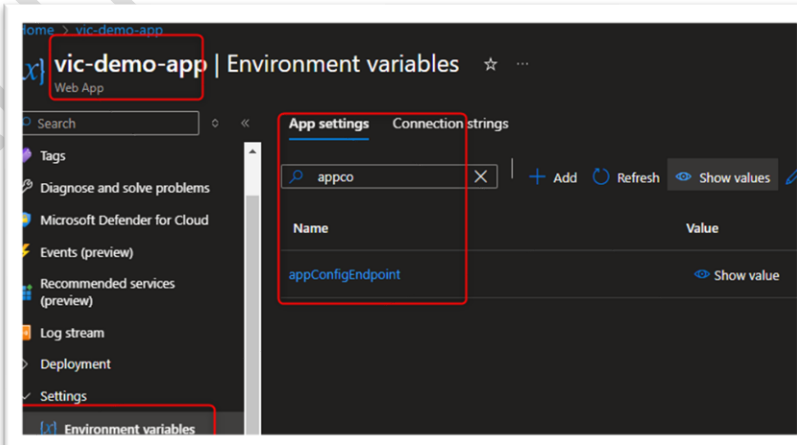
## 2. Task 2: Connect to App Configuration store by using connection string (ASP .net core API)

Steps	Demonstration
Packages: <i>Microsoft.Azure.AppConfigu</i> <i>ration.AspNetCore</i>	
Add code in <i>Program.cs</i> file	<pre>// Load configuration from Azure App Configuration var connectionString = builder.Configuration.GetValue&lt;string&gt;("appConfigConnectionString"); builder.Configuration.AddAzureAppConfiguration(connectionString);</pre>
Read config from the IConfiguration	<pre>[HttpGet] [Route("Greeting")] 0 references   GiangHM, 4 hours ago   1 author, 2 changes public ActionResult&lt;string&gt; Greet1() {     logger.LogInformation("Use Keyed DI 1");     var firstName = _configuration["DemoApp:Testing:FirstName"];     var lastName = _configuration["DemoApp:Testing:LastName"];     var fullName = \$"{lastName} {firstName}";     return Ok(_welcoming.WelcomeFormalWay(fullName, "Hope you're doing well")); }</pre>
Run app to see the result:	<p>200</p> <p>Response body</p> <p>Good morning Mai Hoang Giang, it's our pleasure to have you here, Hope you're doing well</p>

### 3. Task 3: Improve security by using Microsoft Entra ID

Steps	Demonstration
Packages: <i>Microsoft.Azure.AppConfiguration.AspNetCore</i> <i>Azure.Identity</i>	
Add Role Assignment to App Configuration	
Add code in <i>Program.cs</i> file	
Run app to see the result:	

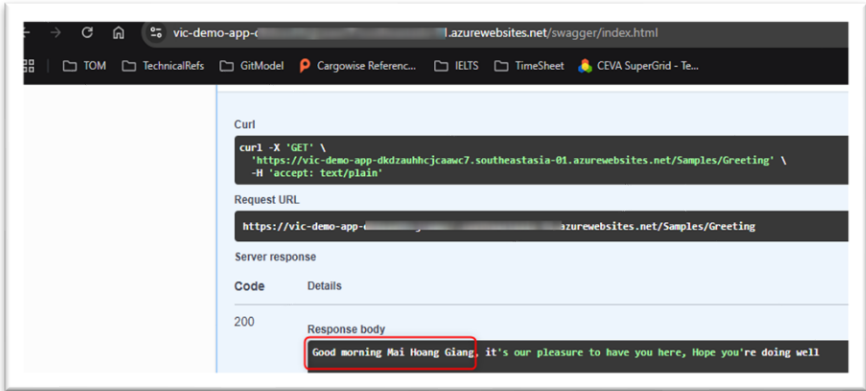
### 4. Task 4: Deploy to Azure App Service

Steps	Demonstration
Deploy to <i>App Service</i> and Add the app configuration endpoint	

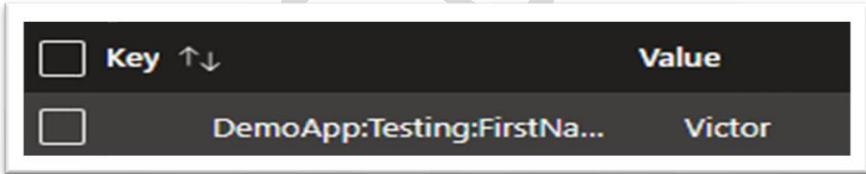
Grant access permission to App Service in order to access App Configuration



Results



Update config value



Don't restart my app, the app can get the new value

