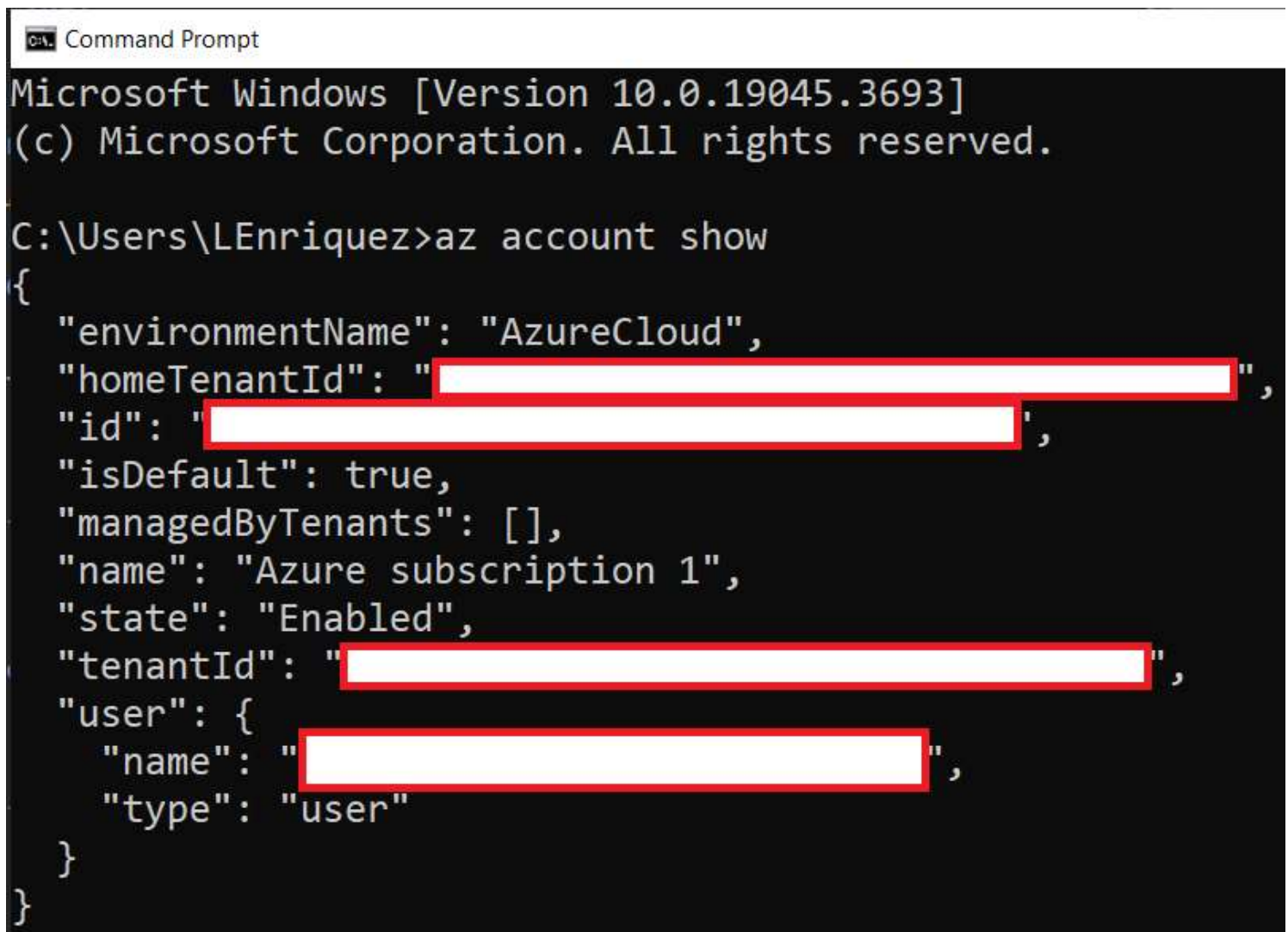


# How to create an Azure Resource Group with a .NET 8 console application and Azure SDK for .NET

## 0. Prerequisite

1. Create an **Azure Free** account: <https://azure.microsoft.com/en-us/free>
2. Install Azure CLI on Windows: <https://learn.microsoft.com/en-us/cli/azure/install-azure-cli-windows?tabs=azure-cli>
3. Confirme the Azure CLI installation running this command:

```
az account show
```



```
Microsoft Windows [Version 10.0.19045.3693]
(c) Microsoft Corporation. All rights reserved.

C:\Users\LEnriquez>az account show
{
  "environmentName": "AzureCloud",
  "homeTenantId": "[REDACTED]",
  "id": "[REDACTED]",
  "isDefault": true,
  "managedByTenants": [],
  "name": "Azure subscription 1",
  "state": "Enabled",
  "tenantId": "[REDACTED]",
  "user": {
    "name": "[REDACTED]",
    "type": "user"
  }
}
```

**NOTE:** For general info about Azure SDK for .NET navigate to the following github repo:

<https://github.com/Azure/azure-sdk-for-net>

## 1. Open VSCode and create a new C# console application with .NET 8

---

We first create the folder/directory where to place our console application.

```
md sampleCreatingResourceGroup
```

We navigate to the folder

```
cd sampleCreatingResourceGroup
```

We open VSCode running the command

```
code .
```

We create a .NET 8 console application with the command:

```
dotnet new console --framework net8.0
```

## 2. Load the dependencies

---

We load the libraries/dependencies running these commands

To load the library: **Azure.Identity**

<https://www.nuget.org/packages/Azure.Identity>

```
dotnet add package Azure.Identity --version 1.10.4
```

To load the library: **Azure.ResourceManager**

<https://www.nuget.org/packages/Azure.ResourceManager>

```
dotnet add package Azure.ResourceManager --version 1.9.0
```

For loading the library: **Azure.ResourceManager.Resources**

<https://www.nuget.org/packages/Azure.ResourceManager.Resources>

```
dotnet add package Azure.ResourceManager.Resources --version 1.7.0
```

Finally we load the library **Azure.ResourceManager.Storage**

<https://www.nuget.org/packages/Azure.ResourceManager.Storage/1.2.0-beta.2>

```
dotnet add package Azure.ResourceManager.Storage --version 1.2.0-beta.2
```

After installing the libraries we run the command:

```
dotnet restore
```

### 3. We input the application C# source code

---

We open the **program.cs** file and we input the application source code:

```
using System;
using System.Threading.Tasks;
using Azure.Identity;
using Azure;
using Azure.Core;
using Azure.ResourceManager;
using Azure.ResourceManager.Resources;
using Azure.ResourceManager.Storage.Models;

ArmClient armClient = new ArmClient(new DefaultAzureCredential());
SubscriptionResource subscription = await armClient.GetDefaultSubscriptionAsync();

string rgName = "myRgNameLUISCOCO";
AzureLocation location = AzureLocation.WestEurope;
ArmOperation<ResourceGroupResource> operation = await subscription.GetResourceGroups().CreateOrUp
ResourceGroupResource resourceGroup = operation.Value;
Console.WriteLine(resourceGroup.Data.Name);
```

### 4. Build and run the application

---

For executing the application we run the command:

```
dotnet run
```

## 5. See the created Resource Group in Azure portal

After creating a new Resource Group we see the following picture:

The screenshot shows the Microsoft Azure portal interface. The main heading is 'Resource groups' with a subheading 'particular'. Below this, there are filters for 'Subscription equals all' and 'Location equals all'. A table lists the resource groups, with 'myRgNameLUISCOCO' selected. The table has columns for Name, Subscription, and Location. The 'myRgNameLUISCOCO' group is located in 'West Europe' and is associated with 'Azure subscription 1'.

Name	Subscription	Location
DefaultResourceGroup-WEU	Azure subscription 1	West Europe
LCE_VM1	Azure subscription 1	France Central
luisocovm2_group	Azure subscription 1	France Central
myRgNameLUISCOCO	Azure subscription 1	West Europe
NetworkWatcherRG	Azure subscription 1	France Central