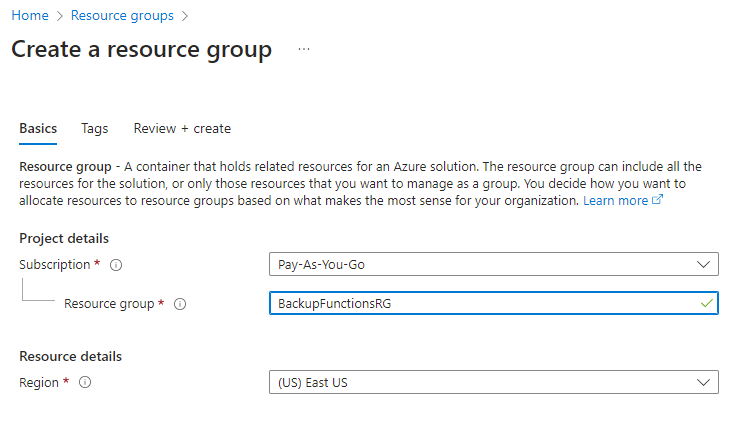
**Step 1: Sign in to the Azure Portal**

1. Open your web browser and navigate to [Azure Portal](https://portal.azure.com).
2. Sign in with your Azure account credentials.

**Step 2: Create a Resource Group**

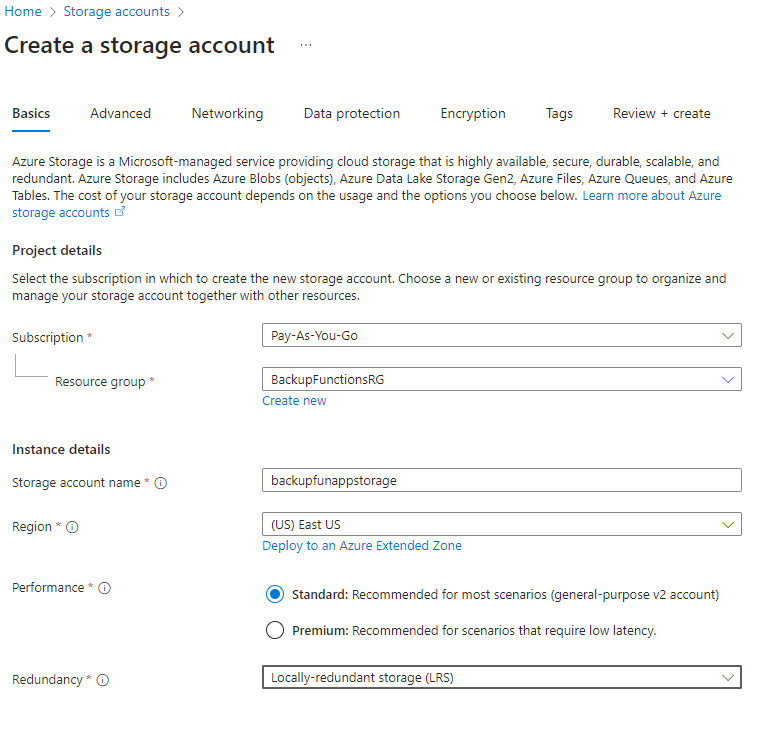
1. In the left-hand navigation pane, click on **Resource groups**.
2. Click on **+ Create** to create a new resource group.
3. Fill in the following details:
   * **Subscription**: Select your subscription.
   * **Resource group**: Enter a name for your resource group, e.g., BackupFunctionsRG.
   * **Region**: Select a region close to your location or where you want your resources to be deployed.



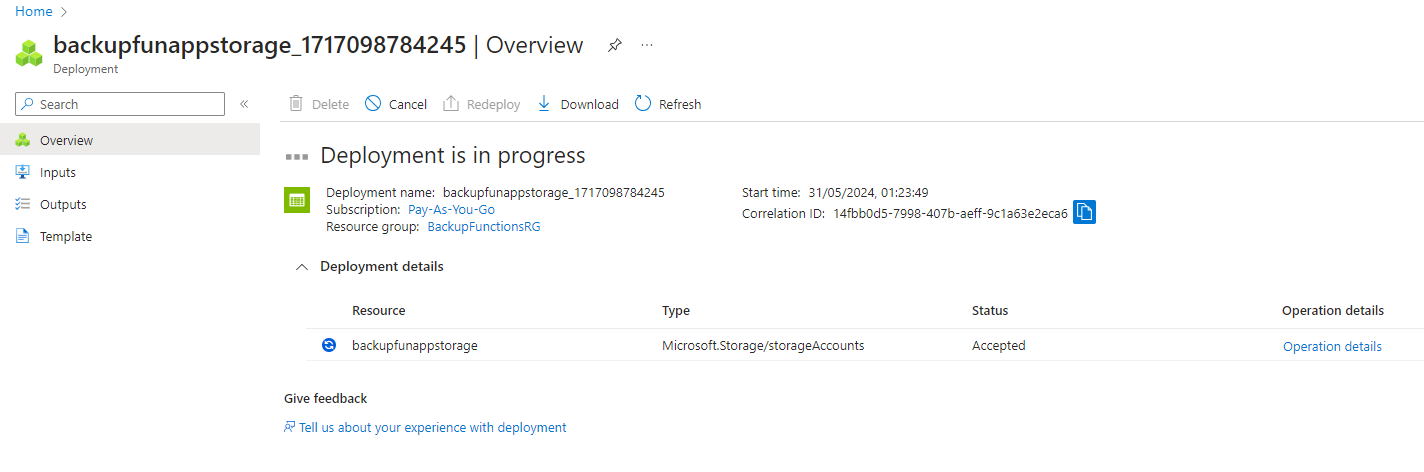
1. Click **Review + create** and then **Create**.

**Step 3: Create a Storage Account**

1. In the left-hand navigation pane, click on **Storage accounts**.
2. Click on **+ Create** to create a new storage account.
3. Fill in the following details:
   * **Subscription**: Select your subscription.
   * **Resource group**: Select BackupFunctionsRG.
   * **Storage account name**: Enter a unique name, e.g., backupfunappstorage.
   * **Region**: Select the same region as your resource group.
   * **Performance**: Select Standard.
   * **Replication**: Select Locally-redundant storage (LRS).

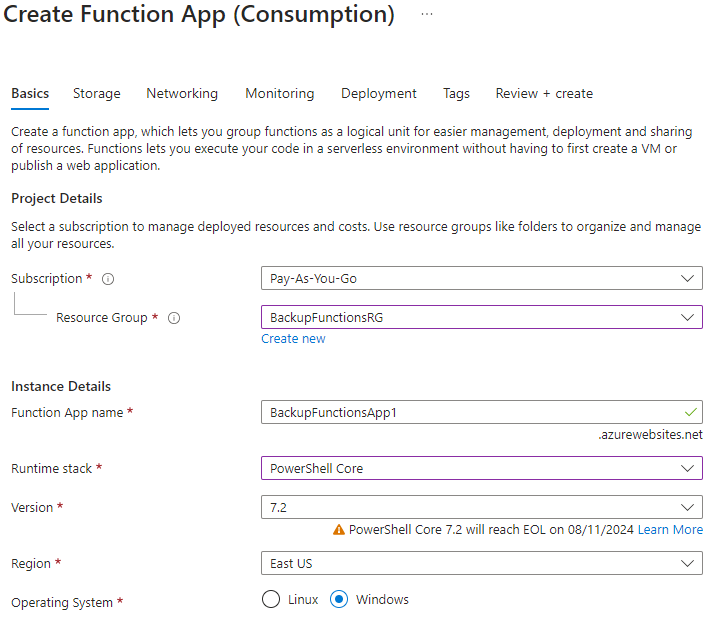


1. Click **Review + create** and then **Create**.

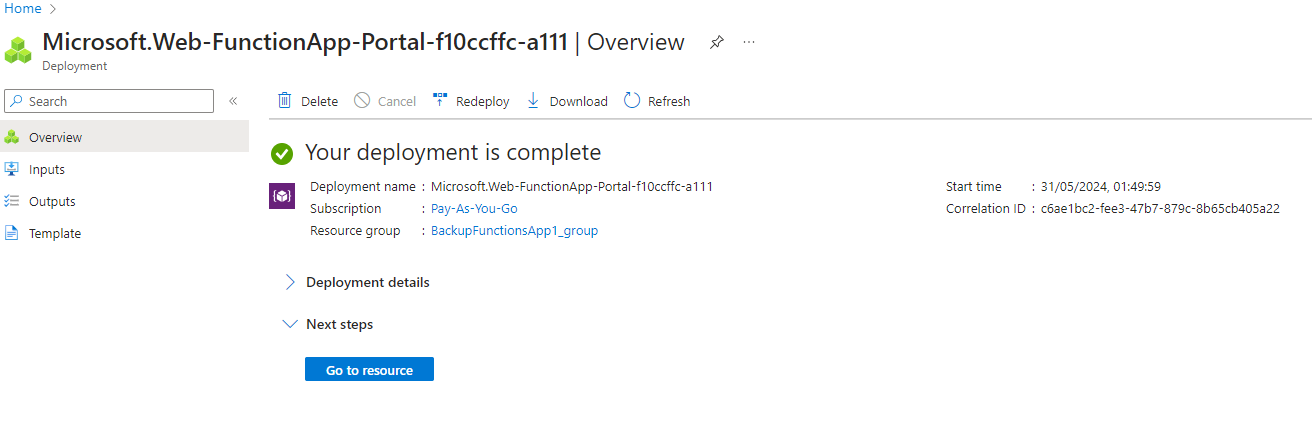


**Step 4: Create a Function App**

1. In the left-hand navigation pane, click on **Function Apps**.
2. Click on **+ Create** to create a new Function App.
3. Select Consumption plan as per your need.
4. Fill in the following details:
   * **Subscription**: Select your subscription.
   * **Resource group**: Select BackupFunctionsRG.
   * **Function App name**: Enter BackupFunctionsApp1. (name must be unique)
   * **Region**: Select the same region as your resource group.
   * **Publish**: Select Code.
   * **Runtime stack**: Select PowerShell Core.
   * **Version**: Select the latest version available.
   * **Operating System**: Select Windows.

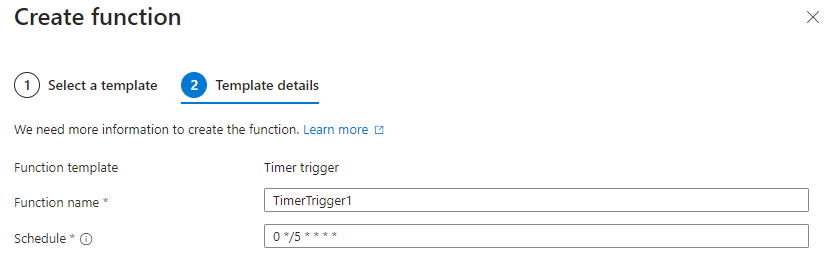


1. Click **Next: Storage**.
2. Under the **Storage** tab, fill in the following details:
   * **Storage account**: Select backupfunappstorage.
3. Click **Review** and then **Create**.



**Step 5: Configure the Function App for PowerShell Scripts**

1. Once the Function App is created, navigate to **Function Apps** in the left-hand navigation pane.
2. Click on BackupFunctionsApp1 to open it.
3. In the **Overview** section, click on **Functions** in the left-hand menu.
4. Click on **Create Function** to create a new function.
5. Select **In-portal** for the development environment.
6. Select **PowerShell** as the language.
7. Choose a template, such as **Timer trigger** or **HTTP trigger**, depending on how you want your backup script to be triggered. For Demo we are using TimerTrigger.



1. Click **Create**.

**Step 6: Add PowerShell Backup Script**

1. Once the function is created, navigate to the function's code editor.
2. Replace the default script with your PowerShell backup script.
3. If your script has dependency related to any module which is present in Powershell Gallery then you can take advantage of managed dependency feature by enabling it from host.json file and add your required modules in requirments.psd1 file but if your dependent module is not present in Powershell Gallery then you need to use KUDU under Developer Tools option and create a new folder called Modules inside site\wwwroot in KUDU option and copy your locally installed module to site\wwwroot\Modules section.
4. Click **Save** to save your script.

