

Lab 1: Connecting to Azure from GitHub Actions

Objective

1. Create a Service Principal.
2. Create a Repo on GitHub

Note:

1. Steps to log into VM (Each participant will have a separate user/pwd)
 - a. Open in a private window <https://training.datacouch.io/pluralsight>
 - b. Enter the provided username and password.

All the following steps are to be done within the Virtual Machine.

- c. Steps to Log into Azure Portal (4-5 participants will be in each group)
 - i. Go to <https://portal.azure.com>
 - ii. Login with the supplied credentials (username and password).
 1. Each group has a unique integer for their login [1-4] eg. **usergroup[1-4]** that will remain same for the duration of the course.
 2. Complete username and Password will be provided in the class.
 3. Each usergroup has an associated resource group which is **rg-usergroup[1-4]**

Section 1: Create a Service Principal

Steps

1. Login into the Virtual Machine
2. Open a stand-alone terminal window or within VS Code.
3. Type **"az login"** on the terminal window.
 - a. A separate browser window will open. Enter your Azure usergroup and password to confirm login and once prompted, close the browser window, and go back to the terminal.
 - b. You should be able to see some details in JSON format about the name, user, tenantId etc.
4. Type **"az group list"** on the terminal window.
 - a. Copy the id value which should be like this
"/subscriptions/<subscription-id>/resourceGroups/rg-usergroup[1-4]"

- b. The above value is required for creating the service principal.
5. Type the following command:
az ad sp create-for-rbac -n <anyname> --scope "<ID value from above step 4>" --sdk-auth --role contributor
Replace both <anyname> and <ID Value> before running this command.
 - a. Wait for the command to complete.
 - b. Copy and paste the entire JSON output on notepad. These values are used for credentials.

Section 2: Create a GitHub Repo

Steps

1. Login to your GitHub account and create a new repo with the defaults.
2. Go to settings and click on **"secrets and variables"**. Click on **Actions**.
 - a. On the right side, click on **"New repository secret"**.
 - i. Name: **AZURE_CREDENTIALS**
 - ii. Secret: Paste the entire JSON that you copied from the last section and click on **"Add Secret"**
 - b. Staying on the same screen, click on **"Variables"** tab and click on **"New repository variable"**
 - i. Name: **RG_NAME**
 - ii. Value: Enter the name of your resource group and click on **"Add variable"**
3. Go to Actions and click on **"set up a workflow yourself"**
4. In the **main.yml** file, copy and paste the below contents and commit and save the file

```
on: workflow_dispatch

name: AzureLoginGitHub

jobs:
  azure-github:
    runs-on: ubuntu-latest
    steps:
      - name: Log in with Azure
        uses: azure/login@v1
        with:
          creds: '${{ secrets.AZURE_CREDENTIALS }}'

      - name: Running a sample AZ CLI command to view the account
        uses: azure/CLI@v1
        with:
          azcliversion: latest
          inlineScript: |
            echo "Show Account information"
            az account show
```

5. We have configured this workflow to run manually. Click on Actions again. Then click on the workflow name “AzureLoginGitHub” and click on “**Run workflow**”.
6. Wait for the workflow to start the job and you can click on the job name to see the progress.

End of Lab.