

Lab 07 – Create a repo

Lab 07

In this lab we will create a private repo in Azure and deploy our image to it.

To do this we will be using the Azure CLI. Hopefully that is installed on your laptops.

We will start by building the resource group and then deploy a new container registry.

Step 1

Connect to Azure via the Azure CLI

1. Open a command prompt and type:

```
az login
```

This will prompt you to go to <https://microsoft.com/devicelogin>

2. Enter the device id you have been given.

Your session will be locked while you're logged in.

3. Find the Subscription id for your account. (Select subscriptions in Azure).
4. Run the following replacing the "<sub id>" for your id "599eb5a3-0fc5-4e18-a220-39ebaa74f940"

```
az account set -s "<sub id>"
```

Replace that number with the subscription id of your Azure.

Step 2

5. Create a new resource group

```
az group create -l northeurope -n mlworkshoprg
```

You can check that this has been created by logging in to the Azure Portal.

Step 3

6. Create an instance of ACR inside your new resource group.

```
az acr create --resource-group mlworkshoprg --name mlworkshopacr --sku Basic
```

Lab 07 – Create a repo

The name which you choose will need to be unique across all of Azure. You cannot use the one that I have listed, append your name to make it easier.

Once that has completed you can check in Azure if it has been created.

Step 4

Login to your new instance of ACR. This will authenticate that our connection is good and that we can deploy images.

7. Run the following:

```
az acr login --name mlworkshopacr
```

You should see the "Login Succeeded"

Step 5

We now need to get the full path to the ACR. We will need to deploy our models.

8. Run the following:

```
az acr list --resource-group mlworkshoprg --query  
"[].{acrLoginServer:loginServer}" --output table
```

This will return something that looks like **mlworkshopacr.azurecr.io**

Step 6

We now need to take our local docker image and deploy it. ACR can collect multiple version of the same image and source control them for you. Each new upload acts as a new version. We will first tag the image with the version is. This is V1 of our model so we will append v1 to the end. This is indicated by the :1.

9. Open a new command prompt

10. Run the following code

```
docker tag productiondiabetes mlworkshopterry.azurecr.io/productiondiabetes:v1
```

Now we need to push this model to ACR.

11. Run the following code

Lab 07 – Create a repo

```
docker push mlworkshopterry.azurecr.io/productiondiabetes:v1
```

Azure CLI should kick in and start pushing all the layers of your image in to Azure. Some parts are quite large so this may take a minute or so.

You can validate that it has been pushed by looking in Azure in ACR and selecting repositories.

12. Or run the following:

```
az acr repository list --name mlworkshopterry --output table
```

Once your image has been pushed, this **lab is complete**.