**Lab 20: Azure DevOps Pipeline**

Creating a new pipeline

The steps to create a new pipeline are straightforward. The challenge is to have the necessary preparations in place to call the Azure Machine Learning pipeline from Azure DevOps.

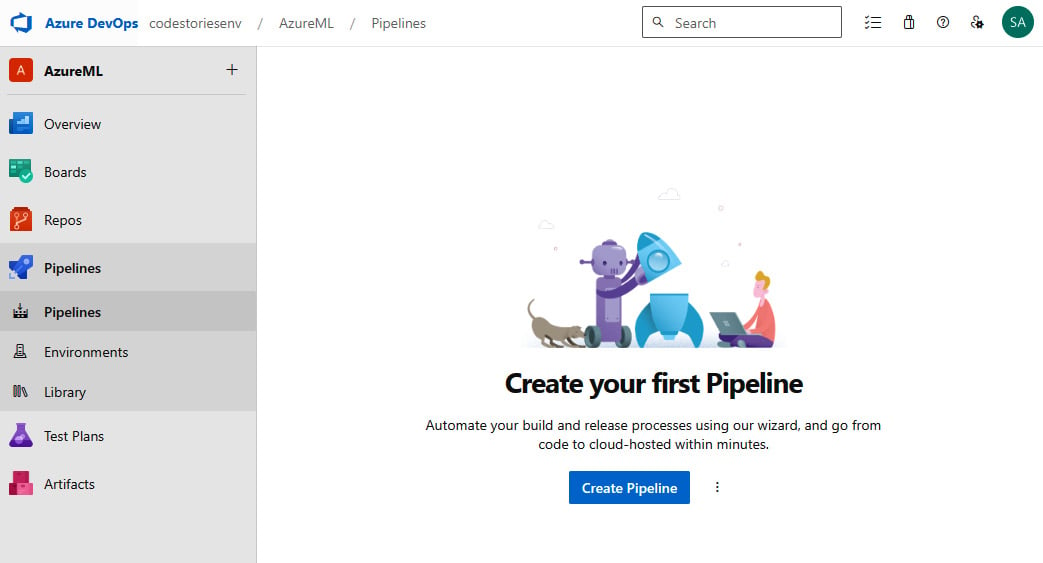
Use the following repository, which includes pipeline examples and code.

Azure Machine Learning example repository

If you don’t have a pipeline and would like to explore some ML examples in Azure, you can fork and run code from this repository: https://github.com/Azure/mlops-v2-ado-demo.

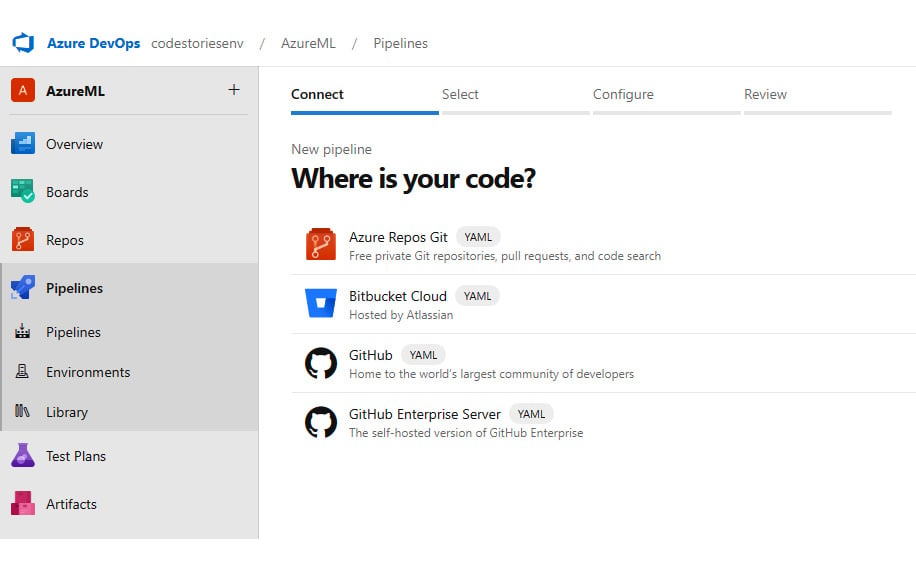
In Azure DevOps, you will need to build a YAML pipeline to run your tasks. Here are the steps necessary to create a new pipeline:

1. Start by creating a new pipeline, as shown in the following screenshot:



New pipeline

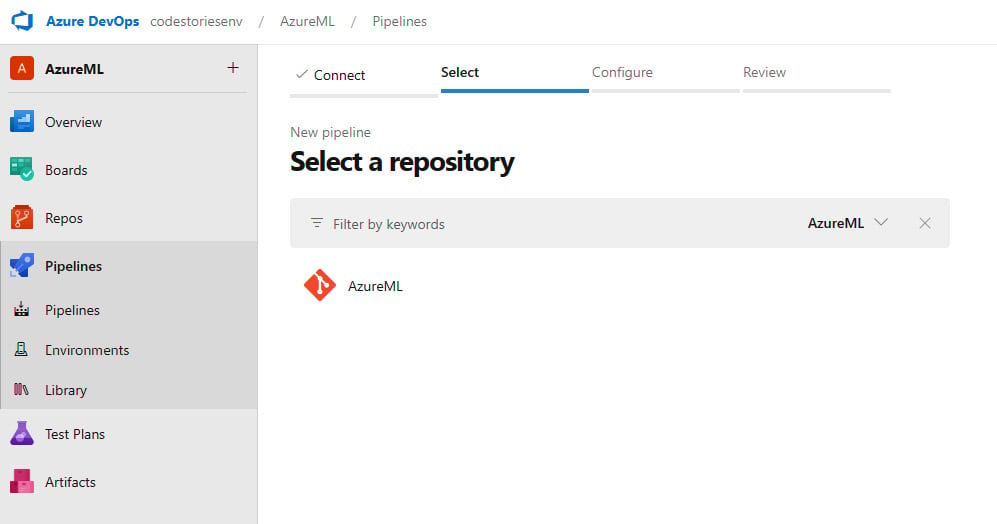
1. You will need to provide the repository where your scripts are stored. You can use the repository from your project or an external one such as Bitbucket or GitHub:



Choosing your code repository

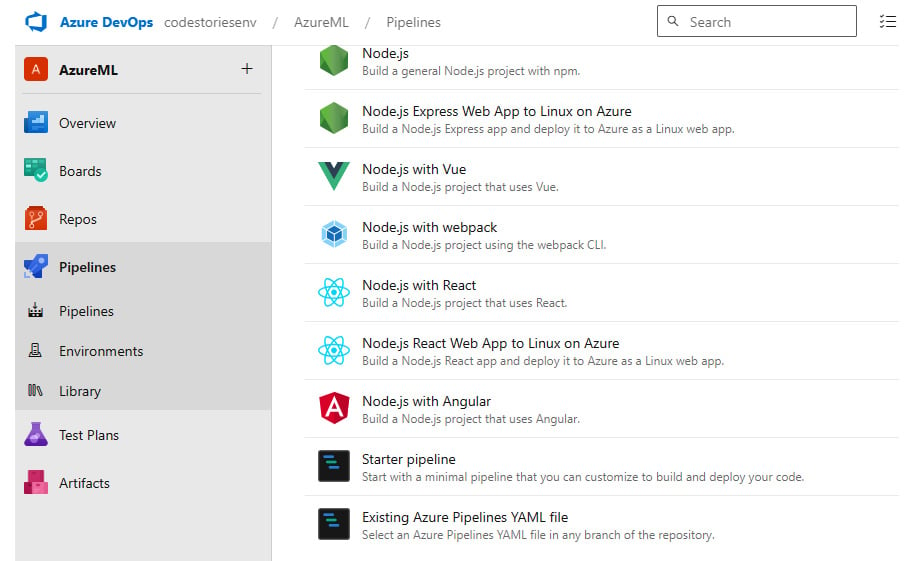
1. I selected the one in the repository in the same project, so all I have to do is select the name of the repository, as shown in the next screenshot:

f



Selecting the repository from the list

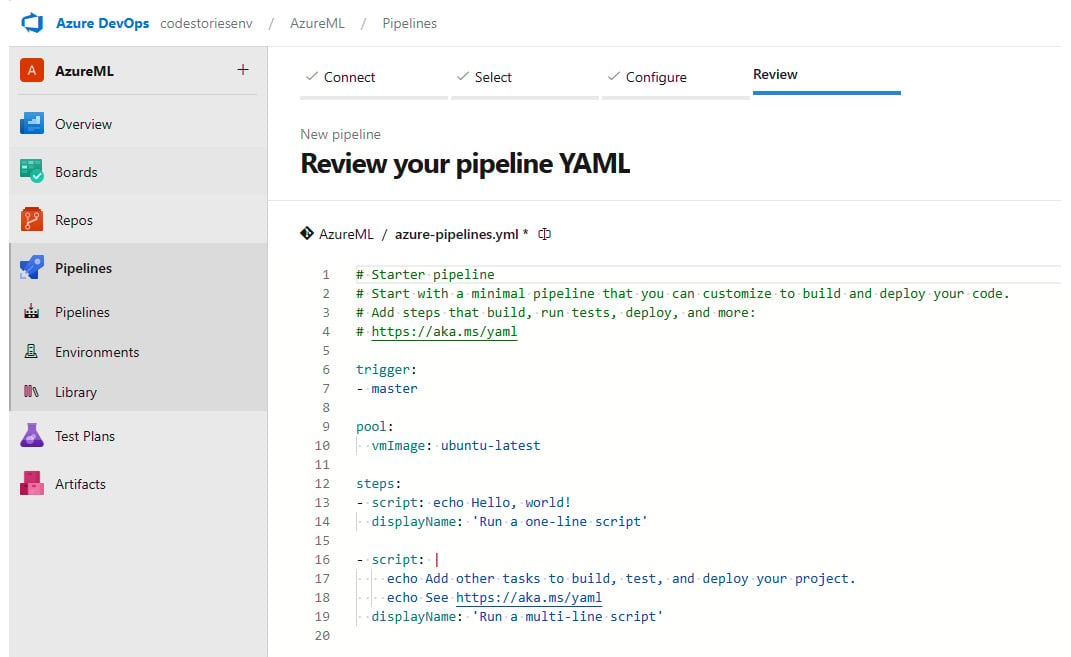
1. Here, select the **Starter pipeline** template to load a simple pipeline YAML file:



Choosing the Starter pipeline template (You can choose any – the point of this lab is just to show you where and how pipelines are created)

Now, here is the big challenge. You need to define the triggers and start up your Azure Machine Learning pipeline from here. An effective way to get started is to use the CLI to submit any ML jobs.

The starter pipeline does provide some sample code, as shown in the next screenshot, to get started and fill in the details. For several things, such as variables, you can also add them via an integrated designer:



Filling in the code to call your pipeline

If you chose to clone the repository suggested in the previous section, you can find the pipeline YAML code in the mlops/devops-pipelines folder. Copy and paste the file of your choice and submit the code.

Once you submit the code, you can run your pipeline. If anything goes wrong, you can see what happened with the associated error messages and you can always go back to fix the pipeline code and resubmit the job.

YAML pipeline editor

For a brief introduction to the YAML editor in Azure DevOps, take a look at the information at this link: [https://learn.microsoft.com/en-us/azure/devops/pipelines/get-started/yaml-pipeline-editor?view=azure-devops](https://learn.microsoft.com/en-us/azure/devops/pipelines/get-started/yaml-pipeline-editor?view=azure).

Azure DevOps provides logs and output that can help you monitor the run. Additionally, you can also monitor the run directly from the Azure Machine Learning workspace. Consider using *triggers* in Azure DevOps to automate the ML pipeline run – for example, every time there is a change to your training script or dataset.