Infrastructure Repository and Pipeline

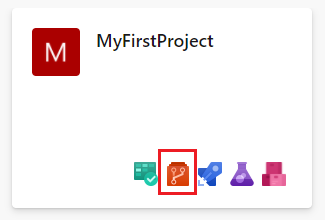
Content

[1. Initialize the repository 2](#_Toc66790278)

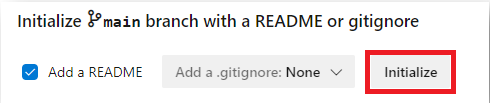
[2. Set up the variable group 5](#_Toc66790284)

[3. Set up and run the pipeline 8](#_Toc66790288)

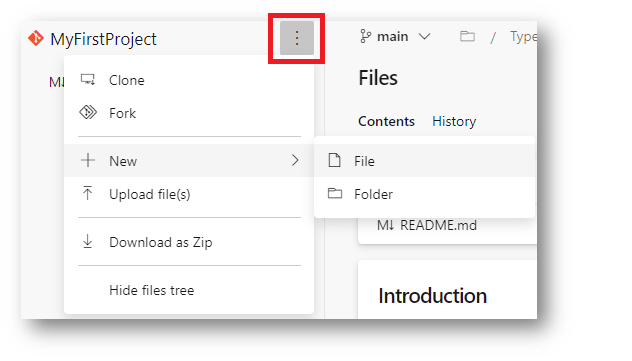
1. Initialize the repository
   1. Navigate to your repository in your project



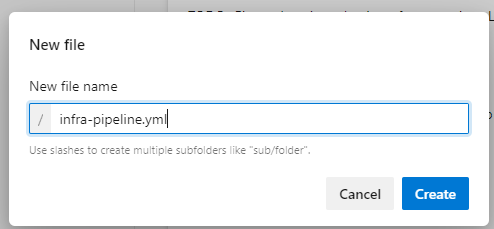
* 1. Initialize your repository with a generic readme file by clicking on Initialize



* 1. Add a new file to your repository



* 1. Give it a meaningful name with the suffix .yml



* 1. Now start writing the infrastructure pipeline to deploy an **App Service Plan** and a **WebApp** in Azure step by step

*Warning: The formatting of YAML (yml) files is based on spaces and tabs and therefor the following lines should be copied with care.  
It is advised to use Visual Studio Code to validate the copied file.*

Starting with making the variable group available to the pipeline. From now on we can reference any variable that is held in the variable group under libraries (described under 2):

variables:  
- group: infravars

Next, we need to set a trigger which will run the pipeline after every commit on this repository:  
(we set it to none to avoid unwanted pipeline runs)  
  
trigger: none  
  
Then we need to set an operating system in which our pipeline will run:

pool:  
 vmImage: ubuntu-latest  
  
After that we define the smallest building block of a pipeline which is called **step** and

can be a task or a script.

This Task that we want to use is a special Azure CLI task that will execute our commands like would do it on the Azure Portal in the cloud shell:

steps:

- task: AzureCLI@2

inputs:

azureSubscription: '$(so)'

scriptType: 'bash'

scriptLocation: 'inlineScript'

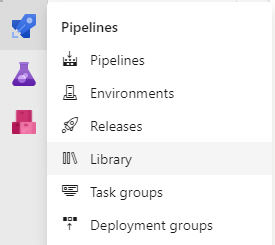
inlineScript: |

az appservice plan create -g $(rg) -n $(asp) --is-linux --number-of-workers 1 --sku B1

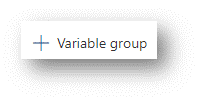
az webapp create -g $(rg) -p $(asp) -n $(wa) --runtime "node|10.14"

If you want to learn more about the concept of a pipeline you can do it here:  
  
https://docs.microsoft.com/de-de/azure/devops/pipelines/get-started/key-pipelines-concepts?view=azure-devops

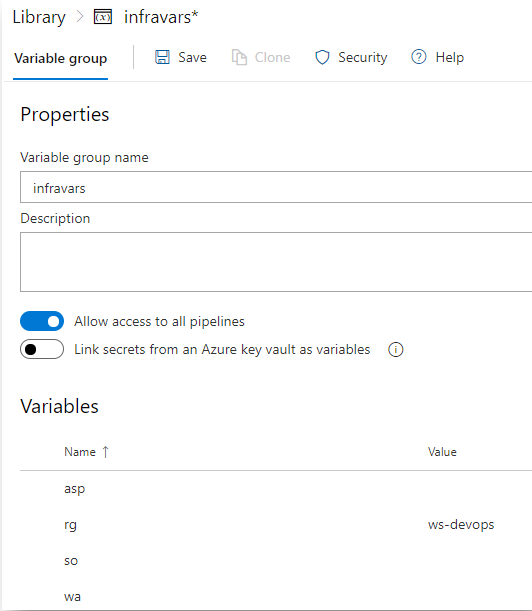
1. Set up the variable group
   1. Navigate to Libraries



* 1. Click on Variable group



* 1. Name it **infravars** and define the variables that you want to use at the pipeline



We all are use the same Subscription and same Resource Group but

everybody uses his own App Service Plan and his own Webapp.

Therefore, pick unique names e.g. robsplan21 or robswebapp21 for your App Service Plan

and Webapp variables.

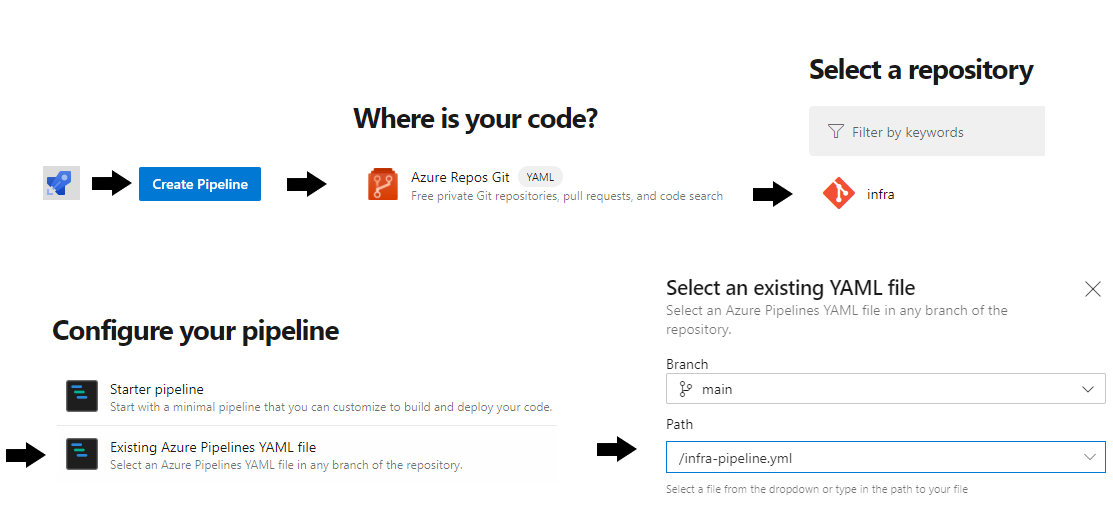
Use for Resource Group (rg) **ws-devops**.

asp = App Service Plan must be unique in each resource group (your own)  
rg = Resource Group must be unique in each subscription **(**use **ws-devops)**  
so = Service Connection Name which you have set in your project (Settings -> Service connections)

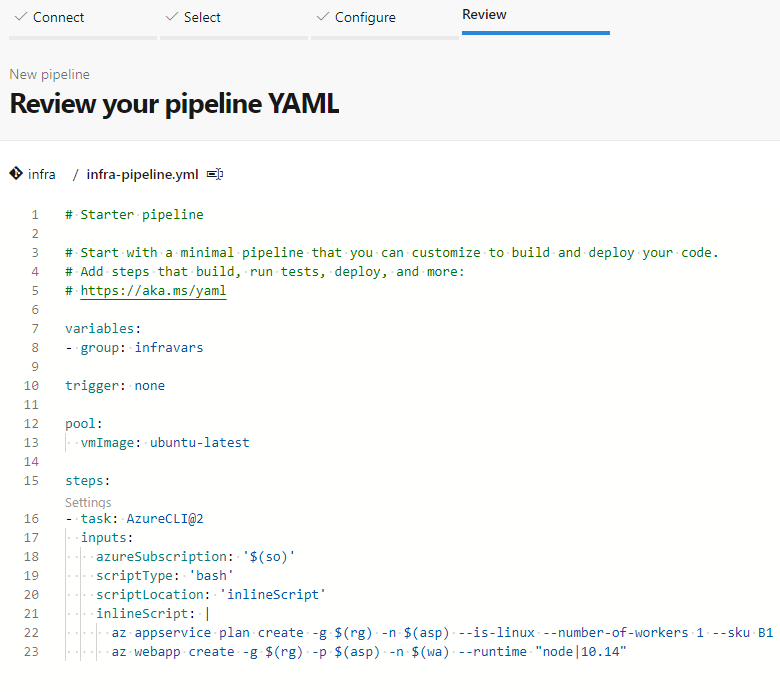
wa = Webapp must be unique worldwide (your own)

Of course, you can use your own variables names, they must correspond with your pipeline.

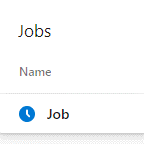
1. Set up and run the pipeline  
   1. Now it is time for some rocket science, therefore click on the rocket and reference it to your pipeline



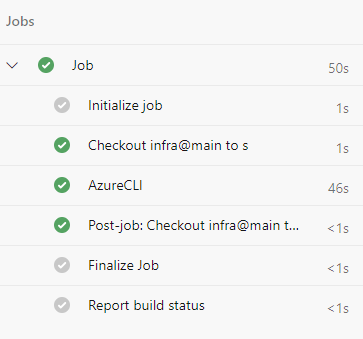
* 1. After all you can admire your work and launch the pipeline by clicking on RUN (upper right corner)



* 1. Now you can watch your pipeline running by clicking on the Job



* 1. After the pipeline has run you should see only green lights



* 1. Check your WebApp is online after approx. 5 minutes

https://<yourWebAppName>.azurewebsites.net/

You should see a “Hey, Node developers” welcome screen.

Congratulations, you have deployed your first WebApp infrastructure.   
Now, you can go ahead and deploy some code to your WebApp.