(iFinance SourceCode Migration From Github to Azure Devops)

**Overview:**

Infrastructure is deployed/ setup on AWS cloud.

Pipeline workflow is running via AWS Code Build.

Source Code is managed via Github.

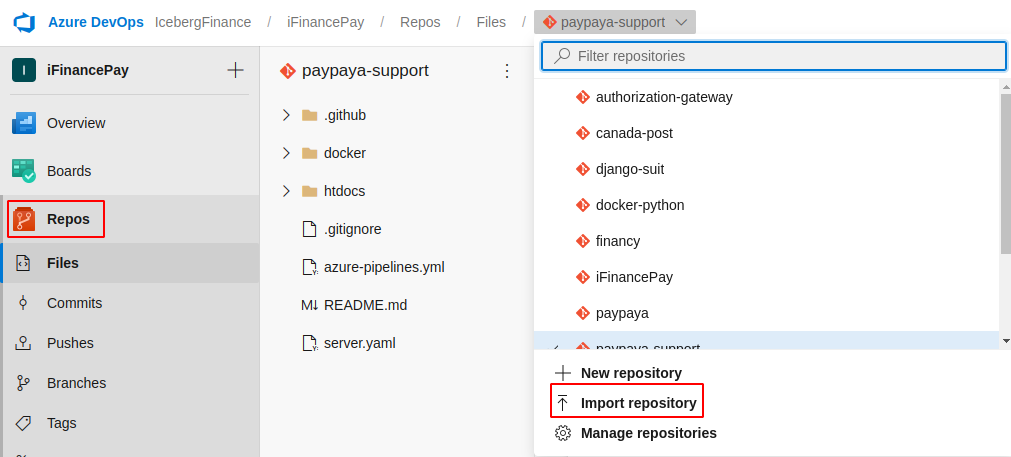
**Challenging tasks:**

* To move the source code repo from github to azure devops
* What is the alternative for github actions in azure devops (On hold now)
* Ansible level modifications for whitelisting azure repos and git config
* How to install python packages via ssh from private git repository in azure devops
* Modify existing pipeline to work with azure devops
* No, webhook option in source code other than github to automate the build flow

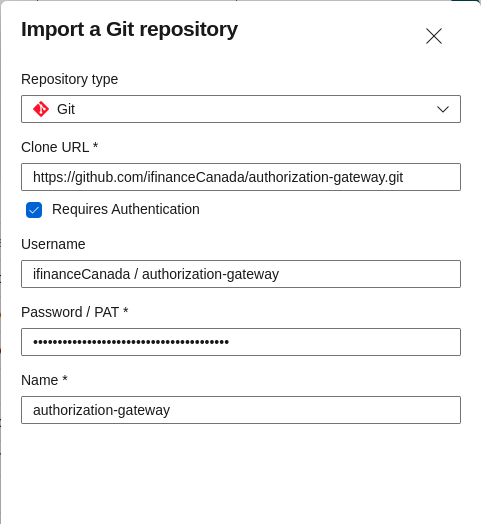
Steps Performed in this Migration

**Import a Repo on Azure from Github:**

1. Importing into a new repo, select Repos, Files and from the repo drop-down, select Import repository.

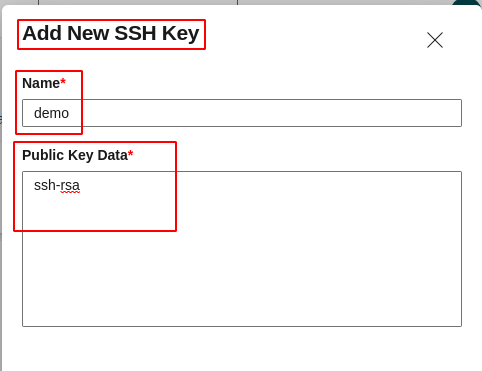
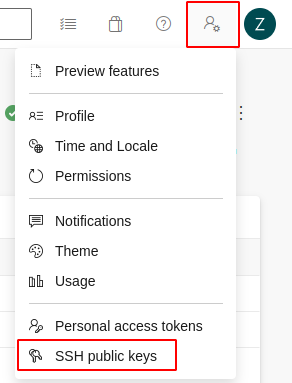


1. If the source repository is private but can be accessed using basic authentication (username-password, personal access token, etc.), select Requires authorization and enter your credentials. SSH authentication is not supported, but you can manually import a repository that uses SSH authentication by following the steps in Manually import a repo using git CLI.



**Adding public key into azure to clone repo via ssh**:

User settings > SSh public keys > New Key > Enter Name and Public Key Data



**Changes made in Ansible configurations to use Azure as source repo:**

1. *configuration/playbooks/roles/paypaya/tasks/main.yml*

**From:**

shell: if [ ! -n "$(grep "^github.com " ~/.ssh/known\_hosts)" ]; then ssh-keyscan github.com >> ~/.ssh/known\_hosts 2>/dev/null; fi

**To:**

shell: if [ ! -n "$(grep "^ssh.dev.azure.com " ~/.ssh/known\_hosts)" ]; then ssh-keyscan ssh.dev.azure.com >> ~/.ssh/known\_hosts 2>/dev/null; fi

**And From:**

shell: git config --global url."git@github.com:".insteadOf "https://github.com/"

**To:**

shell: git config --global url."git@ssh.dev.azure.com:v3".insteadOf "https://dev.azure.com"

1. *configuration/playbooks/overrides/production.yml* and *staging.yml*

**Added** a parameter to override the config:

PAYPAYA\_GIT\_REPO:

"git@ssh.dev.azure.com:v3/IcebergFinance/iFinancePay/paypaya"

**To install python packages via ssh from private azure repository:**

We have a private repo url instead of a package name in requirements file to install the python package so in order to install it we can do below:

**From:**

git+ssh://git@github.com/ifinanceCanada/django-suit.git

**To:**

git+ssh://git@ssh.dev.azure.com/v3/IcebergFinance/iFinancePay/django-suit

**Pipeline Workflow:**

Our pipeline is running on CodeBuild with buildspec.yml file

Old Flow running with Github uses two source providers:

1. Github (app code)
2. CodeCommit (configurations)

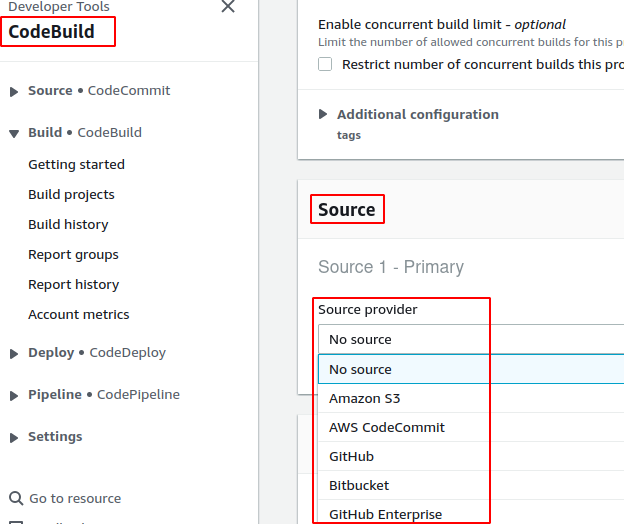
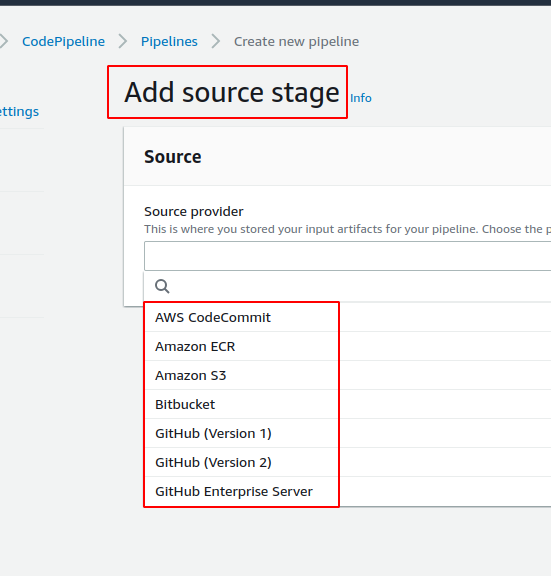
New flow created with only one source provider:

1. AWS CodePipeline

And in CodePipeline we added two source’s both from CodeCommit one for application and second for configuration.

**We now have a Limitation here !**

In Source of both CodeBuild and CodePipeline we have a limited Source providers:



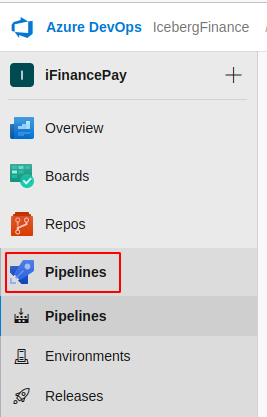
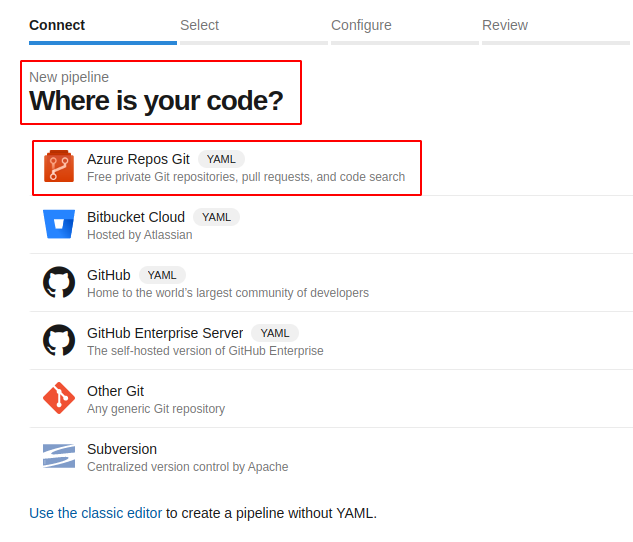
But our Source Code is on Azure Devops Repo which is not listed. So we need a workaround in this case.

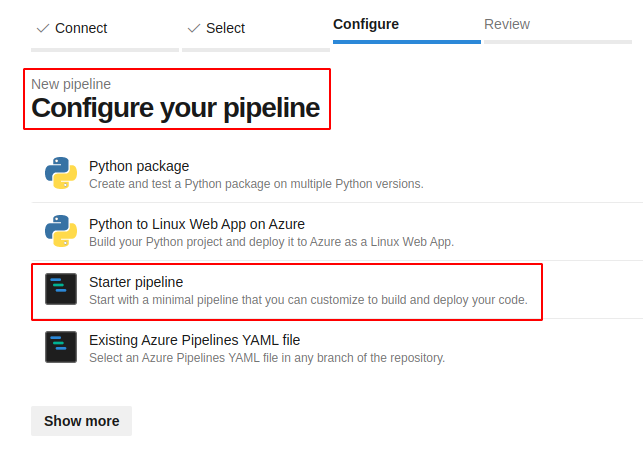
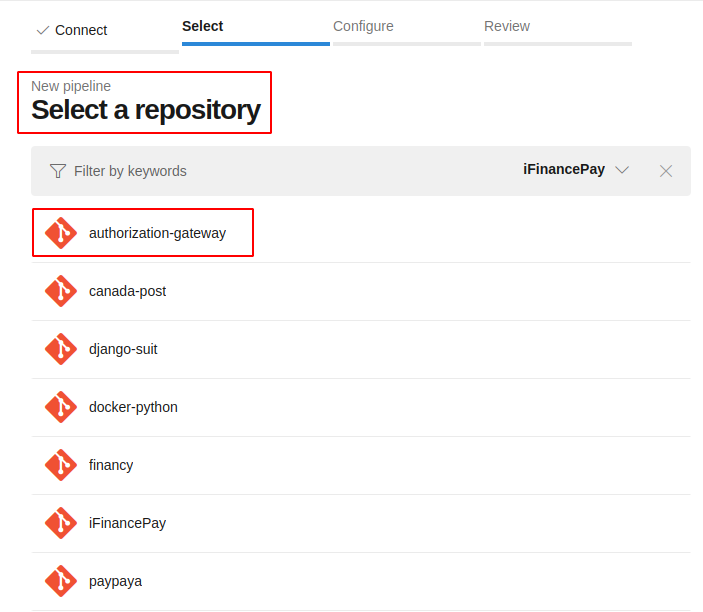
**Workaround of Limitation:**

We have decided to choose the Source as AWS CodeCommit. But yet source code is on azure how this will work?

So then we created a pipeline on Azure Devops that will clone/ mirror the code and its changes to AWS CodeCommit repo.

**Creating a Pipeline on Azure Devops:**

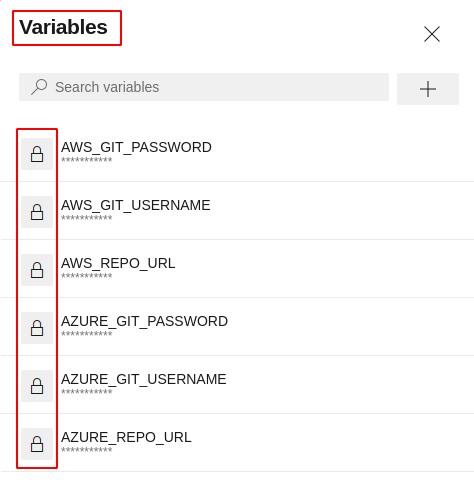
Pipelines > New pipeline > Azure Repos Git Select a repository > Starter pipeline 



azure-pipelines.yml

| # Pipeline to automatically mirror  # an Azure DevOps repository in AWS CodeCommit  # Trigger only on staging branch changes  trigger:  - staging  # Use latest Ubuntu image  pool:  vmImage: 'ubuntu-latest'  # Pipeline  steps:  - checkout: none  - script: |  # Install urlencode function to encode reserved characters in passwords  sudo apt-get install gridsite-clients  # Create local mirror of Azure DevOps repository  git clone --mirror https://${AZURE\_GIT\_USERNAME}:$(urlencode ${AZURE\_GIT\_PASSWORD})@${AZURE\_REPO\_URL} repo-mirror  # Sync AWS CodeCommit repository  cd repo-mirror  git push --mirror https://${AWS\_GIT\_USERNAME}:$(urlencode ${AWS\_GIT\_PASSWORD})@${AWS\_REPO\_URL}  displayName: 'Sync repository with AWS CodeCommit'  env:  AZURE\_REPO\_URL: $(AZURE\_REPO\_URL)  AZURE\_GIT\_USERNAME: $(AZURE\_GIT\_USERNAME)  AZURE\_GIT\_PASSWORD: $(AZURE\_GIT\_PASSWORD)  AWS\_REPO\_URL: $(AWS\_REPO\_URL)  AWS\_GIT\_USERNAME: $(AWS\_GIT\_USERNAME)  AWS\_GIT\_PASSWORD: $(AWS\_GIT\_PASSWORD) |
| --- |



All variables used in the pipeline above are kept secret here and stored in Pipeline > Variables as shown below:

**AWS\_REPO\_URL**:

Https Repo url on CodeCommit

**AWS\_GIT\_USERNAME**:

HTTPS Git credentials for AWS CodeCommit

**AWS\_GIT\_PASSWORD**:

HTTPS Git credentials for AWS CodeCommit

**AZURE\_REPO\_URL**:

Https Repo url on Azure

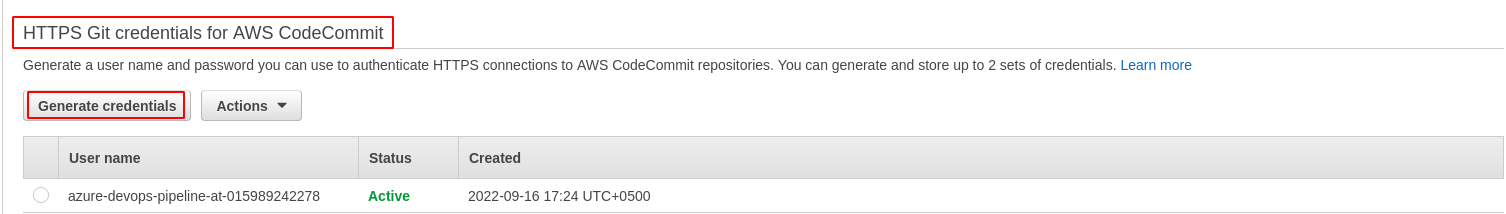
**AZURE\_GIT\_USERNAME**:

HTTPS Git credentials for Azure

**AZURE\_GIT\_PASSWORD**:

HTTPS Git credentials for Azure

AWS HTTPS Git credentials for AWS CodeCommit Generated as below:



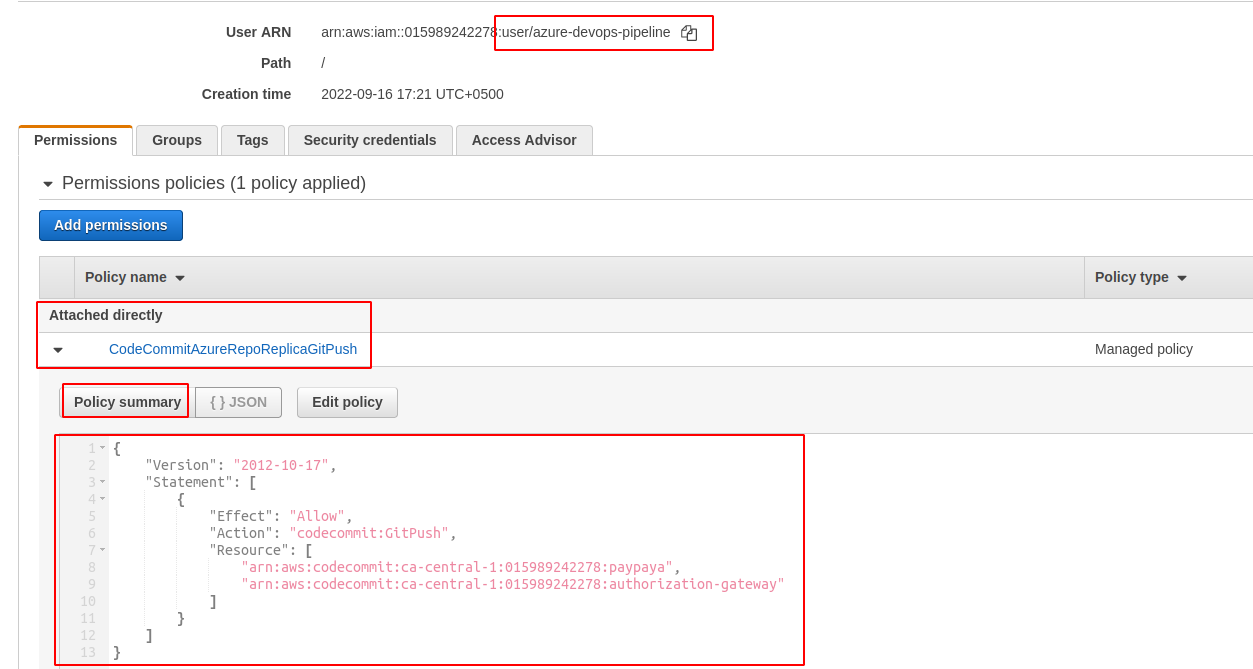
Azure HTTPS Git credentials for Azure DevOps Repo Generated as below:



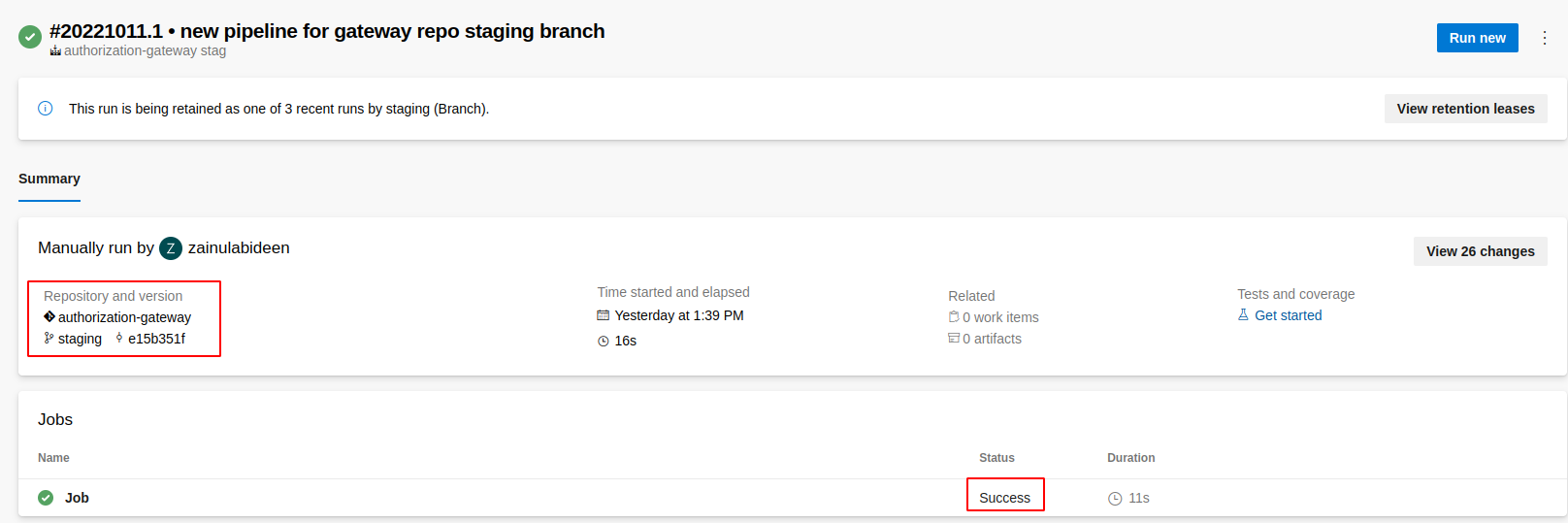
IAM User is created on AWS with custom policy and will be used by the azure pipeline to push code on aws codecommit.

This user also has git credentials generated that was used in the above pipeline.

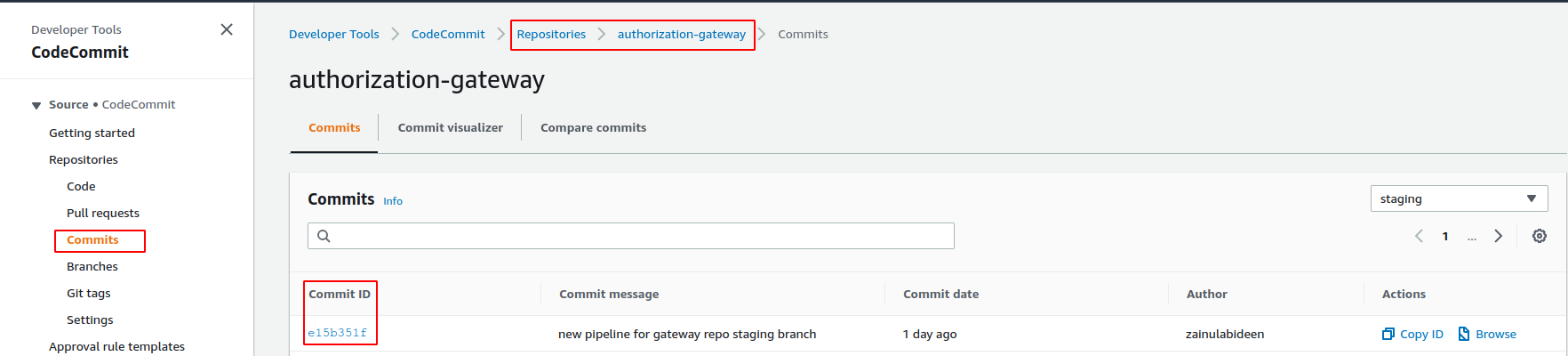
IAM user with custom policy is as below:



After a successful pipeline running it looks like below:



Now we have a repo mirrored on AWS CodeCommit. We can verify it from the commit of azure pipeline and the latest commit on CodeCommit



Notes related to pipeline:

* Every branch have its own pipeline which will not affect the other branch
* Pipeline is committed to default branch when creating from scratch
* We can use same pipeline into multiple branches if the functionality is same
* Variables are not global means it will be specific to each pipeline
* Triggers only when change occurs in that branch where pipeline exists

**Creating a pipeline on AWS:**

**Helpful Links:**

<https://learn.microsoft.com/en-us/azure/devops/repos/git/use-ssh-keys-to-authenticate?view=azure-devops#questions-and-troubleshooting>

<https://learn.microsoft.com/en-us/azure/devops/repos/git/go-get?view=azure-devops#go-get-with-private-projects>

<https://seb-nyberg.medium.com/using-go-modules-with-private-azure-devops-repositories-4664b621f782>

<https://stackoverflow.com/questions/57500028/azure-build-pipeline-with-dependency-on-private-az-repo>

<https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_policies_elements_resource.html>

<https://aws.amazon.com/blogs/devops/use-aws-codecommit-to-mirror-an-azure-devops-repository-using-an-azure-devops-pipeline>