Akilesh K

k.akilesh123@gmail.com

Data engineering - Batch 1

Date: 24-02-24

DAY 25 – Azure DevOps-Pipelines

CI CD Pipelines

- CI/CD stands for Continuous Integration/Continuous Deployment, which is a process central to software development.
- Continuous Integration (CI) involves automatically testing new code changes as soon as they are added to the codebase.
- This testing ensures that new code additions do not introduce any issues or bugs into the system.
- Continuous Deployment (CD) involves automatically deploying code changes to the production environment once they have been tested and approved.
- The CI/CD process automates the testing and deployment of code changes, ensuring efficiency and reliability in software development.
- CI checks and tests every new piece of code, while CD ensures that approved code changes are deployed to the live system without manual intervention.

Deployment

- **Automated Deployment:** CD tools automate the deployment of code changes to production after passing CI checks, ensuring smooth data flow.
- **Monitoring and Alerts:** Monitoring tools track the performance and data quality of the data pipeline, with automated alerts to notify of any discrepancies or issues.
- **Rollbacks:** CD processes enable quick rollbacks to a previously stable state of the data pipeline in case issues are identified post-deployment.

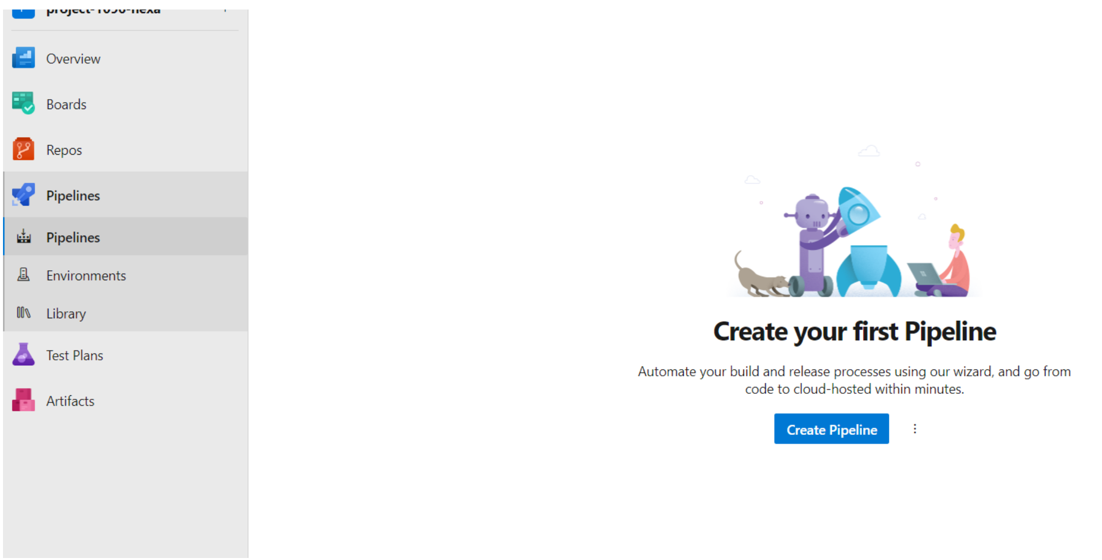
- **Infrastructure as Code (IaC):** Many CD tools support IaC, allowing for automatic provisioning of cloud resources such as storage or compute as part of the deployment process.

Data pipelines

- Data pipeline deployment with Git and CI/CD: Git's integration with CI/CD solutions enables the automation of deployment processes for data pipelines, ensuring seamless delivery to production environments.
- Automation of repetitive tasks: CI/CD tools execute scripts to validate data integrity, schedule analytical tasks, and catch anomalies or failures, minimizing risks and maintaining data quality.
- Novel workflow for data engineering: Unlike traditional software development, data engineering adopts a branch deployment approach, allowing for isolated testing and iteration of changes without impacting the production environment.
- CI/CD and ephemeral environments: CI/CD tools can automatically provision ephemeral environments upon branch creation in Git, facilitating safe and isolated testing of data pipelines, transformations, and integrations before deployment to production.
- Ephemeral environments: These temporary environments are spun up for development, testing, or experimentation purposes and are torn down after use, ensuring efficient resource utilization and cost management.

Create a pipeline in azure DevOps

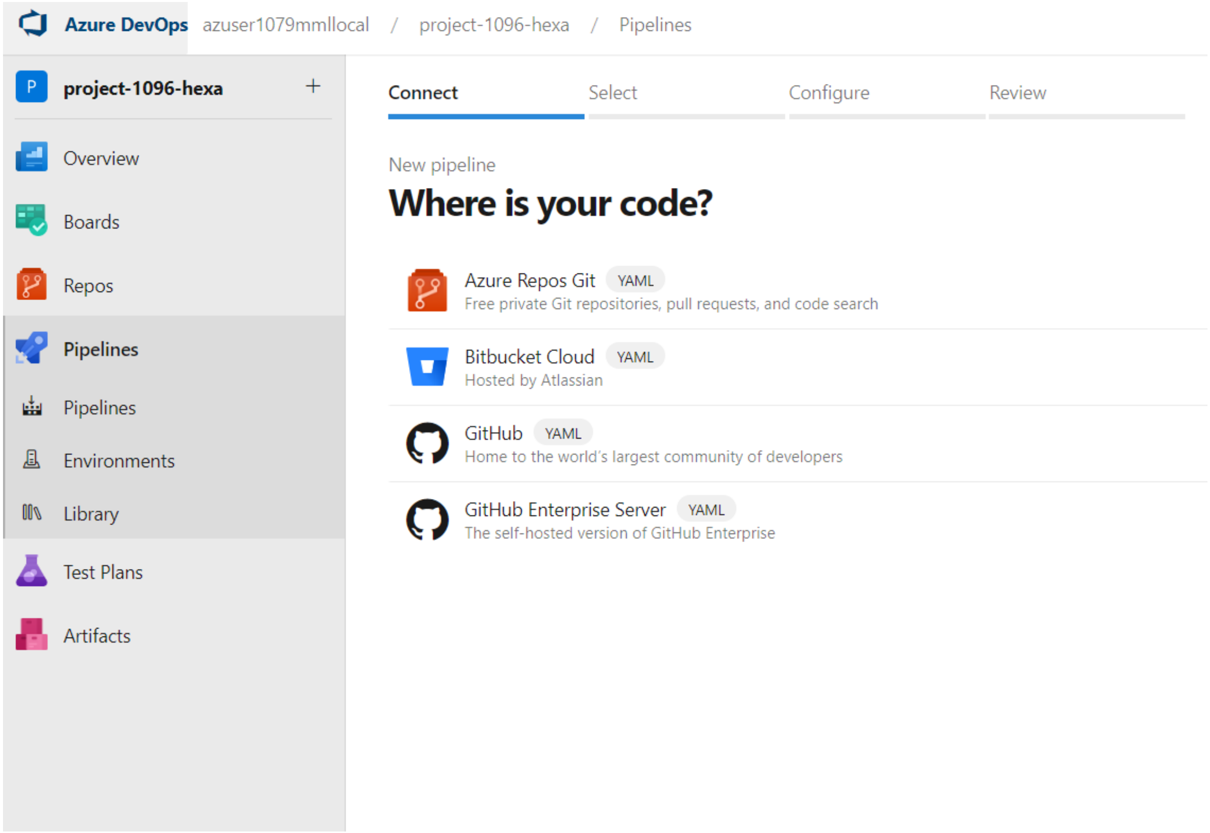
- **Create pipeline**



Create your first Pipeline

Automate your build and release processes using our wizard, and go from code to cloud-hosted within minutes.

- **Azure repos git**



New pipeline

Where is your code?

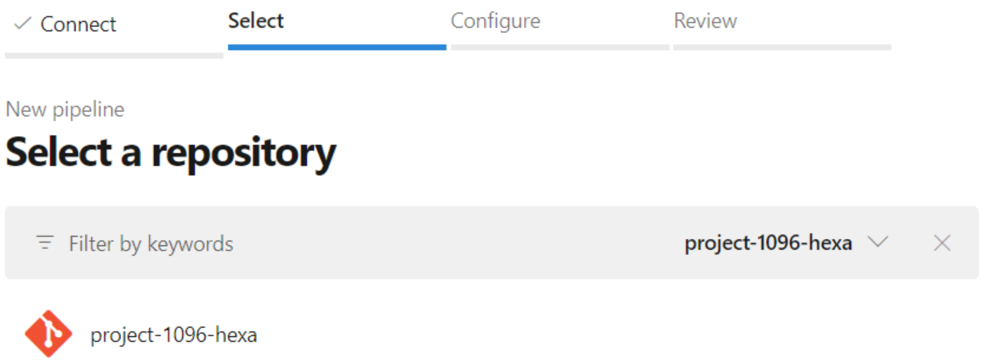
Azure Repos Git YAML
Free private Git repositories, pull requests, and code search

Bitbucket Cloud YAML
Hosted by Atlassian

GitHub YAML
Home to the world's largest community of developers

GitHub Enterprise Server YAML
The self-hosted version of GitHub Enterprise

- **Select repository**



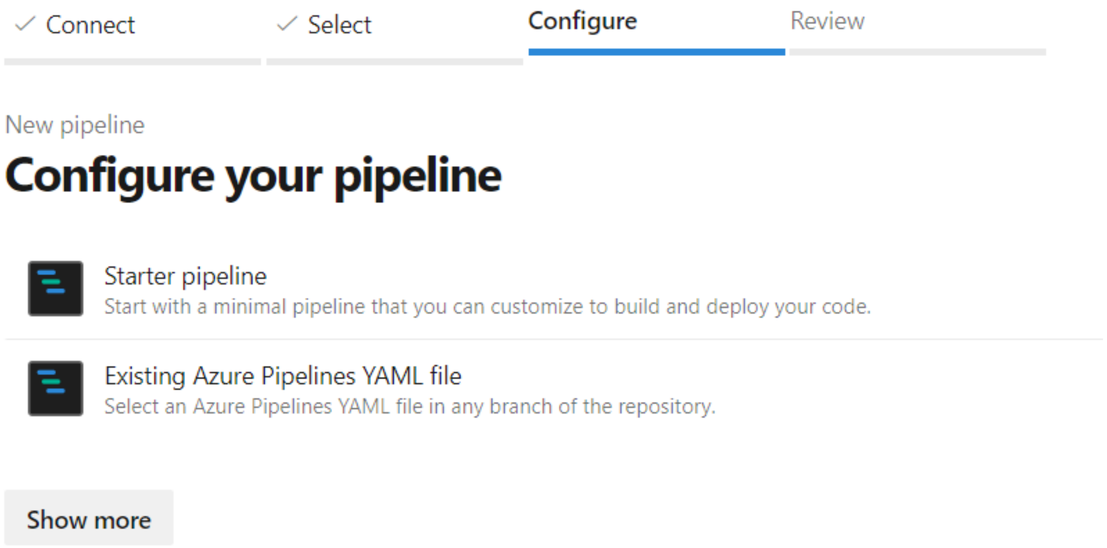
New pipeline

Select a repository

Filter by keywords   project-1096-hexa

project-1096-hexa

- **Start a pipeline**



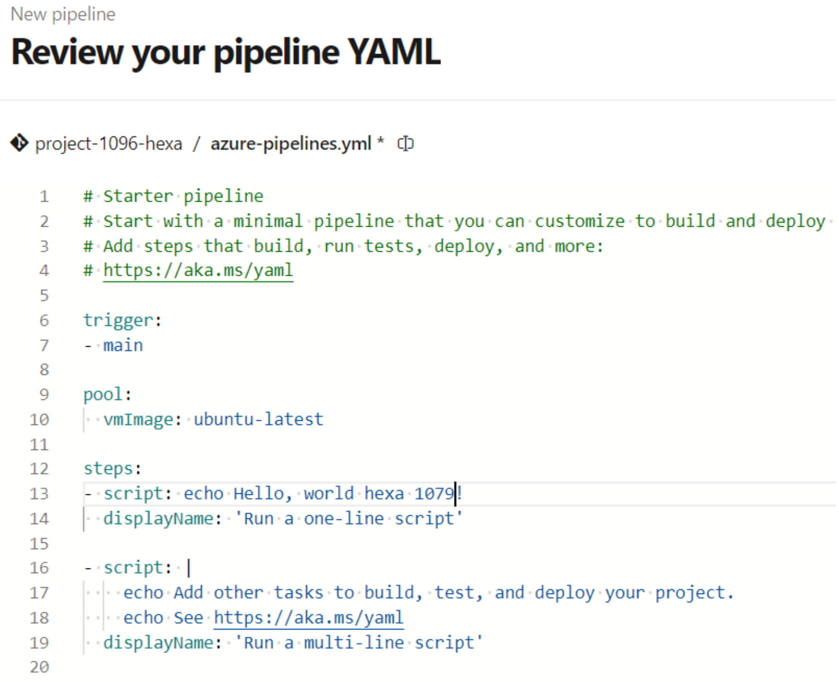
New pipeline

Configure your pipeline

Starter pipeline
Start with a minimal pipeline that you can customize to build and deploy your code.

Existing Azure Pipelines YAML file
Select an Azure Pipelines YAML file in any branch of the repository.

- **YAML code for pipelines**



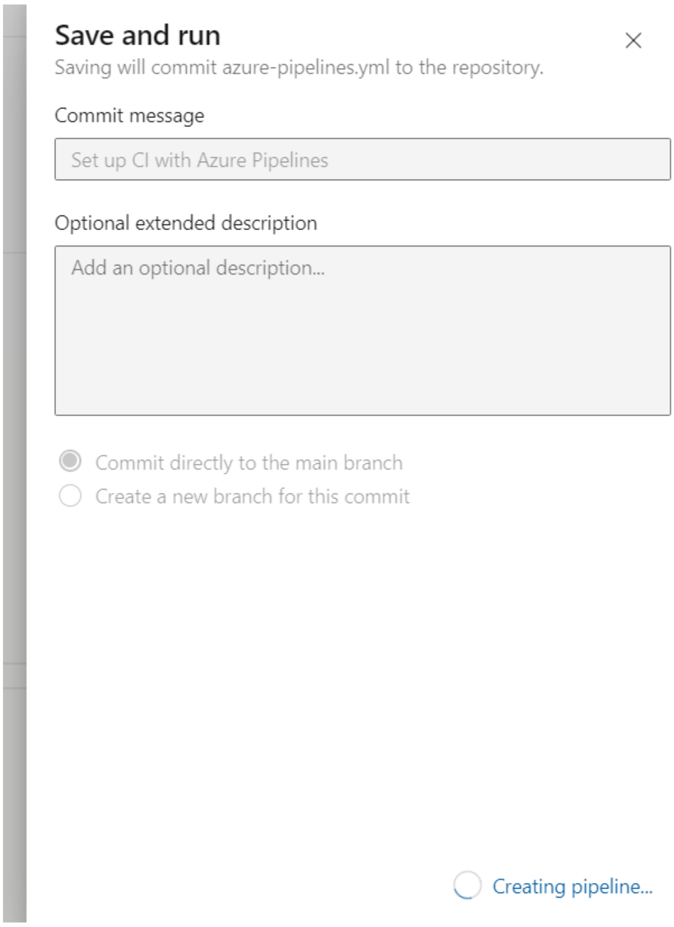
New pipeline

Review your pipeline YAML

project-1096-hexa / **azure-pipelines.yml** *

```yaml
# Starter pipeline
# Start with a minimal pipeline that you can customize to build and deploy
# Add steps that build, run tests, deploy, and more:
# https://aka.ms/yaml

trigger:
- main

pool:
  vmImage: ubuntu-latest

steps:
- script: echo Hello, world hexa 1079!
  displayName: 'Run a one-line script'

- script: |
    echo Add other tasks to build, test, and deploy your project.
    echo See https://aka.ms/yaml
  displayName: 'Run a multi-line script'

```

- **Save and run**



Save and run

Saving will commit azure-pipelines.yml to the repository.

Commit message

Set up CI with Azure Pipelines

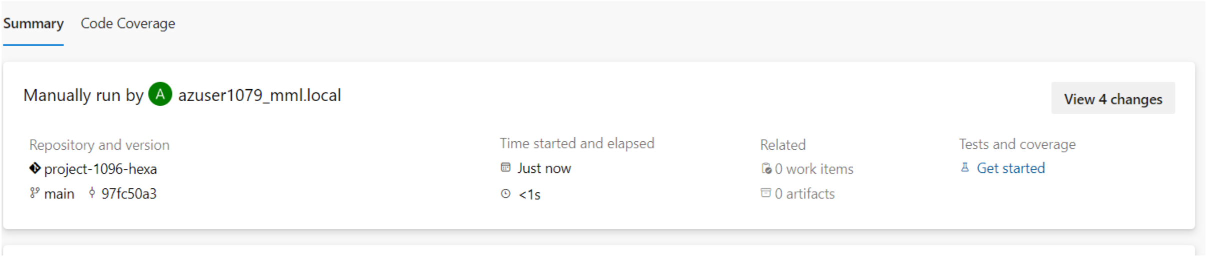
Optional extended description

Add an optional description...

● Commit directly to the main branch

○ Create a new branch for this commit

- **Summary of the pipeline**



Manually run by A azuser1079_mml.local

Repository and version	Time started and elapsed	Related	Tests and coverage
project-1096-hexa	Just now	0 work items	Get started
main 97fc50a3	<1s	0 artifacts	