**Task 4: Create a multi-stage pipeline for deployments across Dev, Test, and Prod environments with specific configurations.**

**Expected Output**: multi-stage-pipeline.yml, environment-specific parameter files.

**# multi-stage-pipeline.yml**

trigger:

- main

pool:

name: 'self-hosted-server'

stages:

- stage: setup\_aks

jobs:

- job: configure\_AKS\_cluster

steps:

- script: |

az aks get-credentials --resource-group my-node-rg --name aks-demo-cluster

displayName: "configure AKS access"

- stage: Dev

displayName: "deploy to Dev"

jobs:

- job: dev\_Deploy

displayName: "deploy to dev env"

steps:

- script: |

kubectl apply -f dev/dev-namespace.yaml

kubectl apply -f dev/nginx-deployment.yaml

kubectl apply -f dev/nginx-service.yaml

displayName: "Deploy to Dev"

- stage: Test

displayName: "deploy to test"

dependsOn: Dev

jobs:

- job: test\_deploy

displayName: "deploy to test env"

steps:

- script: |

kubectl apply -f test/test-namespace.yaml

kubectl apply -f test/nginx-test-deployment.yaml

kubectl apply -f test/nginx-test-service.yaml

displayName: "Deploy to Test"

- stage: Prod

displayName: "deploy to prod"

dependsOn: Test

jobs:

- job: prod\_deploy

displayName: "deploy to prod env"

steps:

- script: |

kubectl apply -f prod/prod-namespace.yaml

kubectl apply -f prod/nginx-prod-deployment.yaml

kubectl apply -f prod/nginx-prod-service.yaml

displayName: "Deploy to Prod"

* **Dev:**

**# dev-namespace.yaml**

apiVersion: v1

kind: Namespace

metadata:

name: dev-ns

**# nginx-deployment.yaml**

apiVersion: apps/v1

kind: Deployment

metadata:

name: nginx-dev-deployment

namespace: dev-ns

spec:

replicas: 2

selector:

matchLabels:

app: nginx

template:

metadata:

labels:

app: nginx

spec:

containers:

- name: nginx

image: nginx:latest

ports:

- containerPort: 80

**# nginx-service.yaml**

apiVersion: v1

kind: Service

metadata:

name: nginx-dev-service

namespace: dev-ns

spec:

type: LoadBalancer

selector:

app: nginx

ports:

- protocol: TCP

port: 80

targetPort: 80

* **Test:**

**# test-namespace.yaml**

apiVersion: v1

kind: Namespace

metadata:

name: test-ns

**# nginx-test-deployment.yaml**

apiVersion: apps/v1

kind: Deployment

metadata:

name: nginx-test-deployment

namespace: test-ns

spec:

replicas: 2

selector:

matchLabels:

app: nginx

template:

metadata:

labels:

app: nginx

spec:

containers:

- name: nginx

image: nginx:latest

ports:

- containerPort: 80

**# nginx-test-service.yaml**

apiVersion: v1

kind: Service

metadata:

name: nginx-test-service

namespace: test-ns

spec:

type: LoadBalancer

selector:

app: nginx

ports:

- protocol: TCP

port: 80

targetPort: 80

* **PROD:**

**# prod-namespace.yaml**

apiVersion: v1

kind: Namespace

metadata:

name: prod-ns

**# nginx-prod-deployment.yaml**

apiVersion: apps/v1

kind: Deployment

metadata:

name: nginx-prod-deployment

namespace: prod-ns

spec:

replicas: 2

selector:

matchLabels:

app: nginx

template:

metadata:

labels:

app: nginx

spec:

containers:

- name: nginx

image: nginx:latest

ports:

- containerPort: 80

**# nginx-prod-service.yaml**

apiVersion: v1

kind: Service

metadata:

name: nginx-prod-service

namespace: prod-ns

spec:

type: LoadBalancer

selector:

app: nginx

ports:

- protocol: TCP

port: 80

targetPort: 80