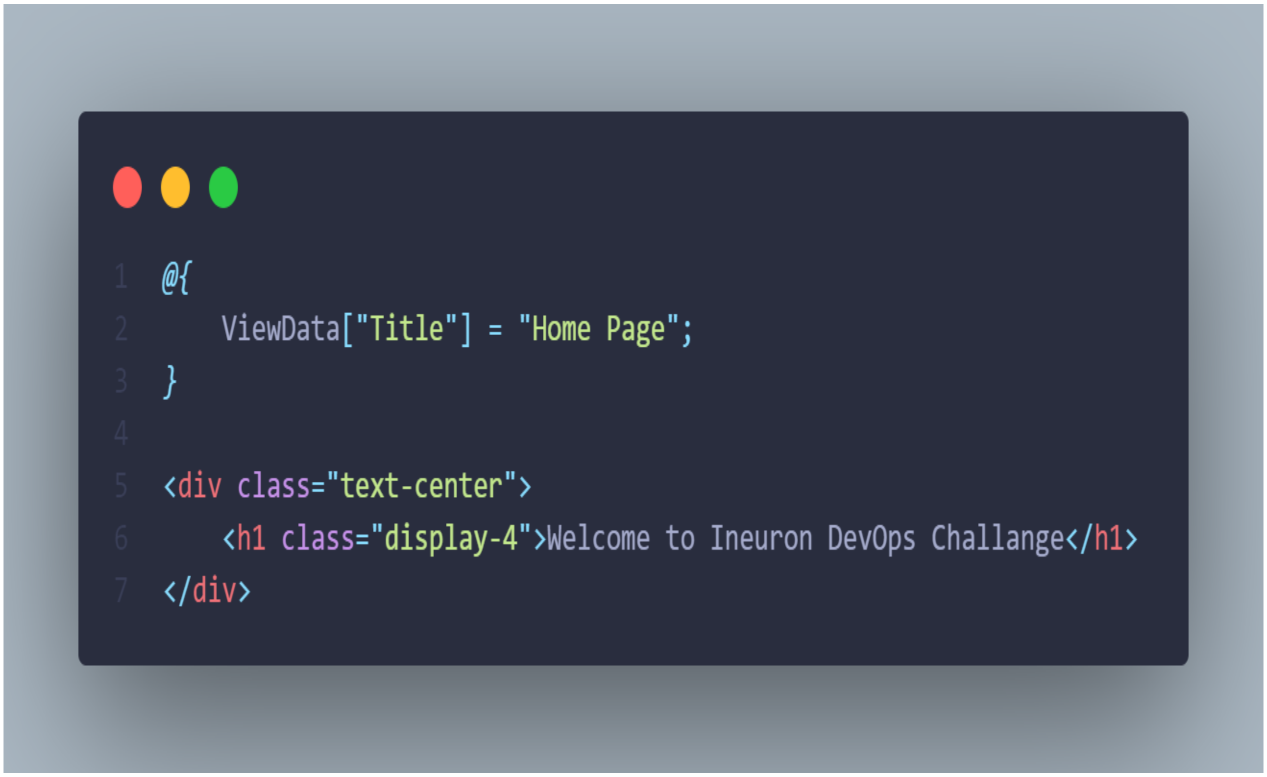
AZURE DevOps challenge

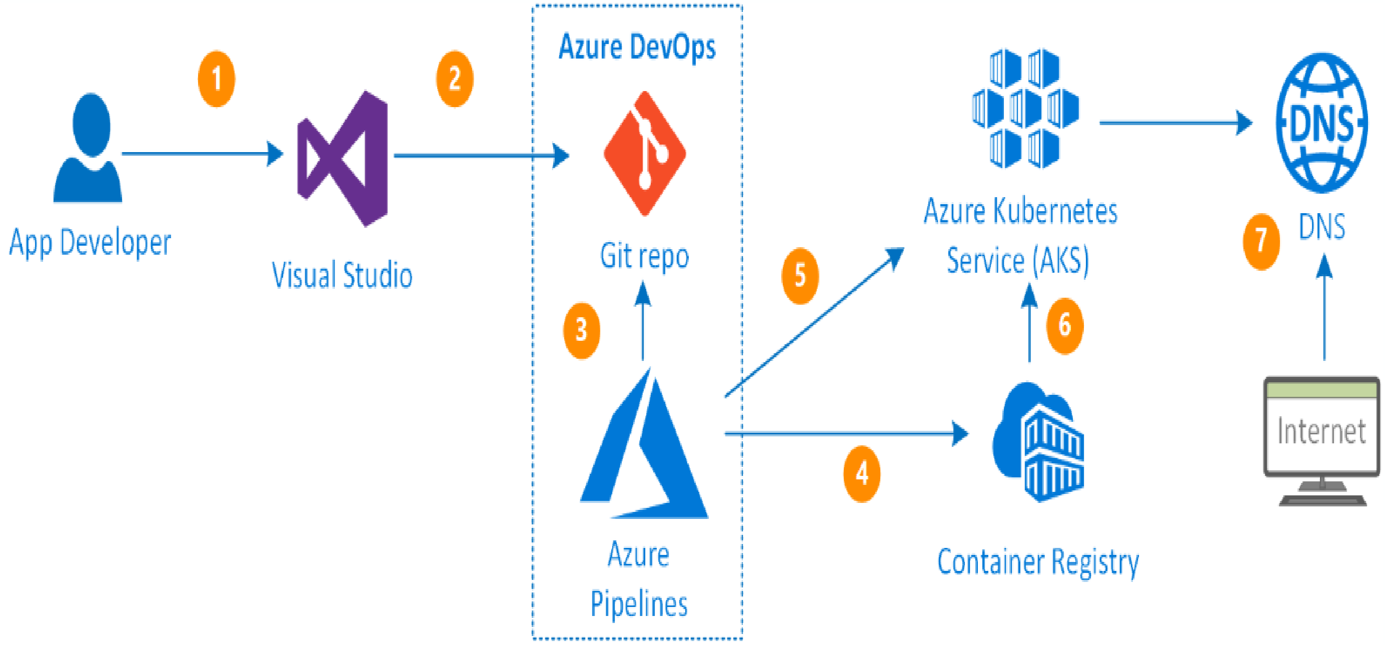
Problem Statement: Create a sample application using any technology along with microservices to demonstrate implementation of the DevOps principle using Azure services. Establish a pipeline for continuous integration, continuous testing, and continuous deployment.

Application : A very basic ASP.NET Web application which will show “Welcome to Ineuron DevOps Challenge” on web page.



```
@{
    ViewData["Title"] = "Home Page";
}

<div class="text-center">
    <h1 class="display-4">Welcome to Ineuron DevOps Challange</h1>
</div>
```

CI/CD Flow:



Benefits of using Azure AKS solutions:

- · Hosts your Kubernetes environment
- · Easy integration with Azure services such as Load balancing, Azure Blob Storage, Azure Active Directory, Application Gateway, Azure Traffic Manager etc.
- · Quick and easy to deploy
- · Hosted control plane
- · Easy and secure containerized applications management.
- · Continuous Integration by adopting Azure Pipeline concept for Docker images creation for faster deployments and reliability

- · Create resources and infrastructure inside the Azure Kubernetes cluster through Deployments and services manifest files
- · AKS management service is free of charge in Microsoft Azure

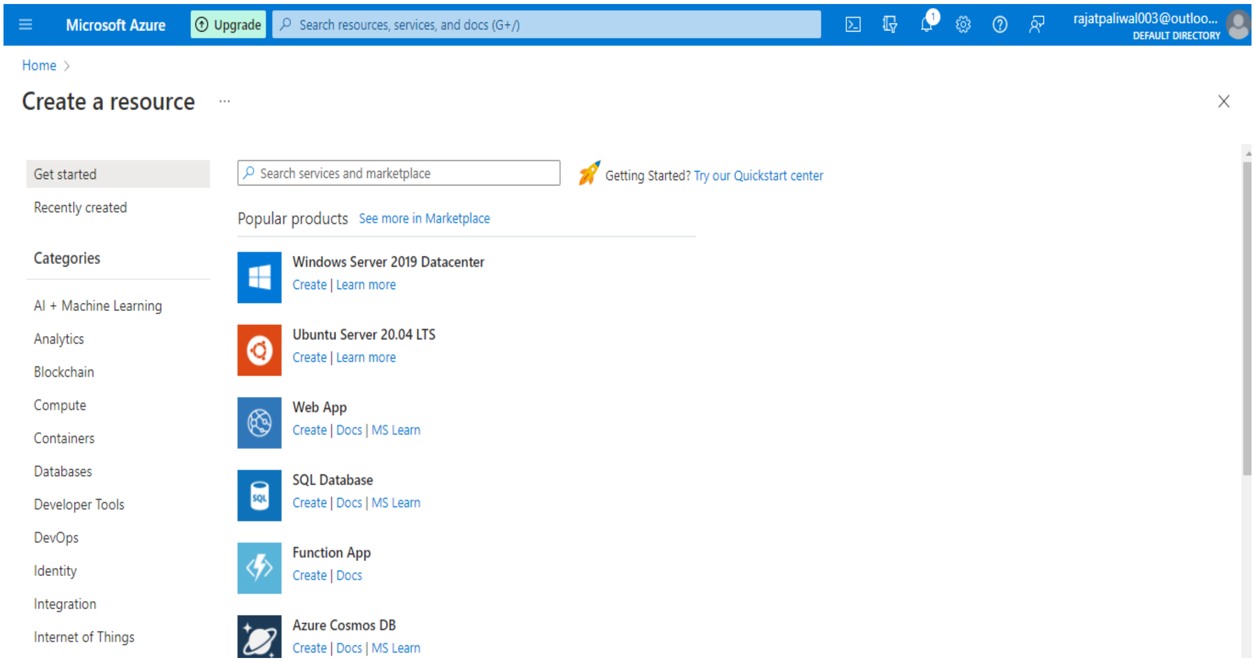
Scheme:

- Create Azure Resource Group
- Create Azure Container Registry
- Create Azure Kubernetes Cluster
- Create a Pipeline For Deployment to Kubernetes

Let's get started,

Step 1 - Create an Azure Resource Group

Now we will create a resource group named “DevOps” for our docker images and Kubernetes cluster. We will keep our all resources in this resource group that we are creating.



Create a resource

Getting Started? Try our Quickstart center

Popular products See more in Marketplace

Windows Server 2019 Datacenter
Create | Learn more

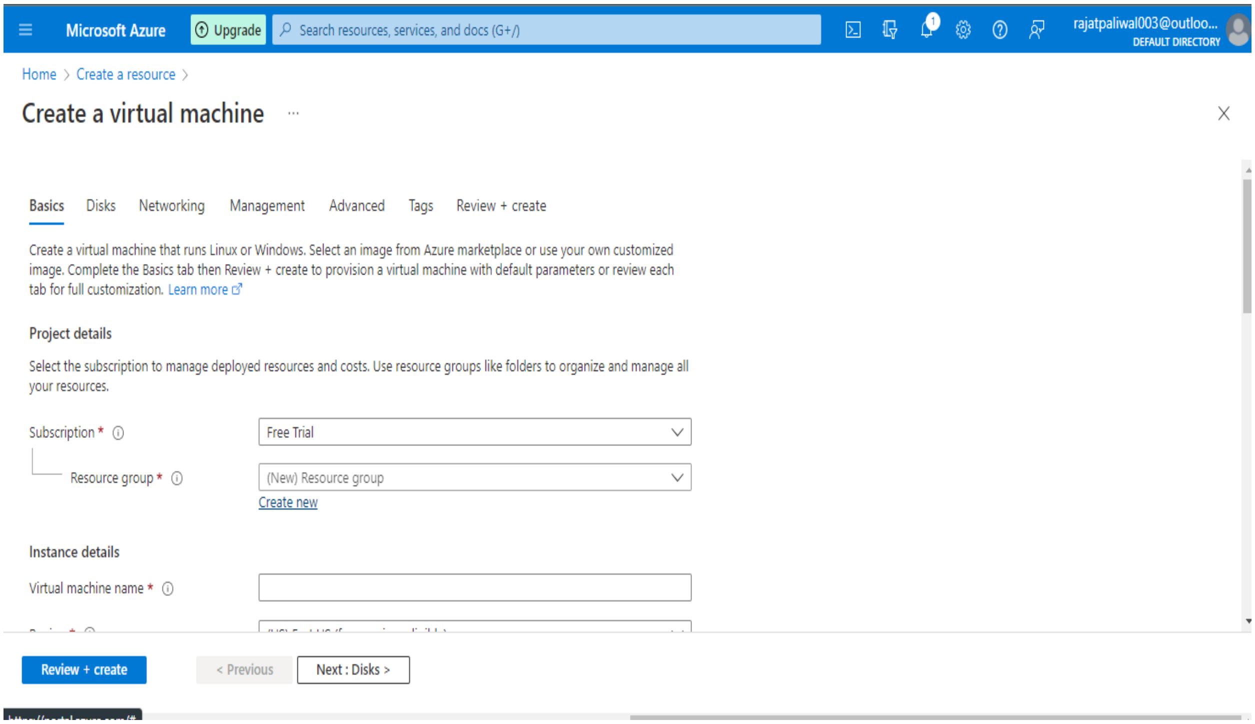
Ubuntu Server 20.04 LTS
Create | Learn more

Web App
Create | Docs | MS Learn

SQL Database
Create | Docs | MS Learn

Function App
Create | Docs

Azure Cosmos DB
Create | Docs | MS Learn



Create a virtual machine

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. Learn more

Project details

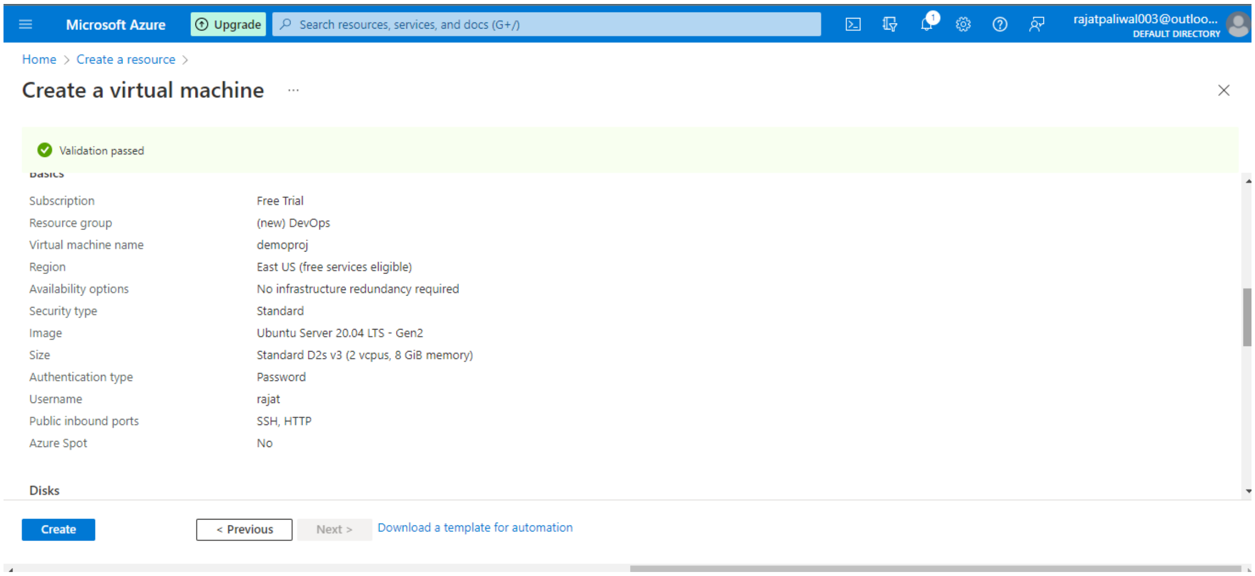
Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ Free Trial

Resource group * ⓘ (New) Resource group
Create new

Instance details

Virtual machine name * ⓘ


Microsoft Azure
Upgrade
Search resources, services, and docs (G+/)
rajatpaliwal003@outloo...
DEFAULT DIRECTORY
Home > Create a resource >

Create a virtual machine

Validation passed

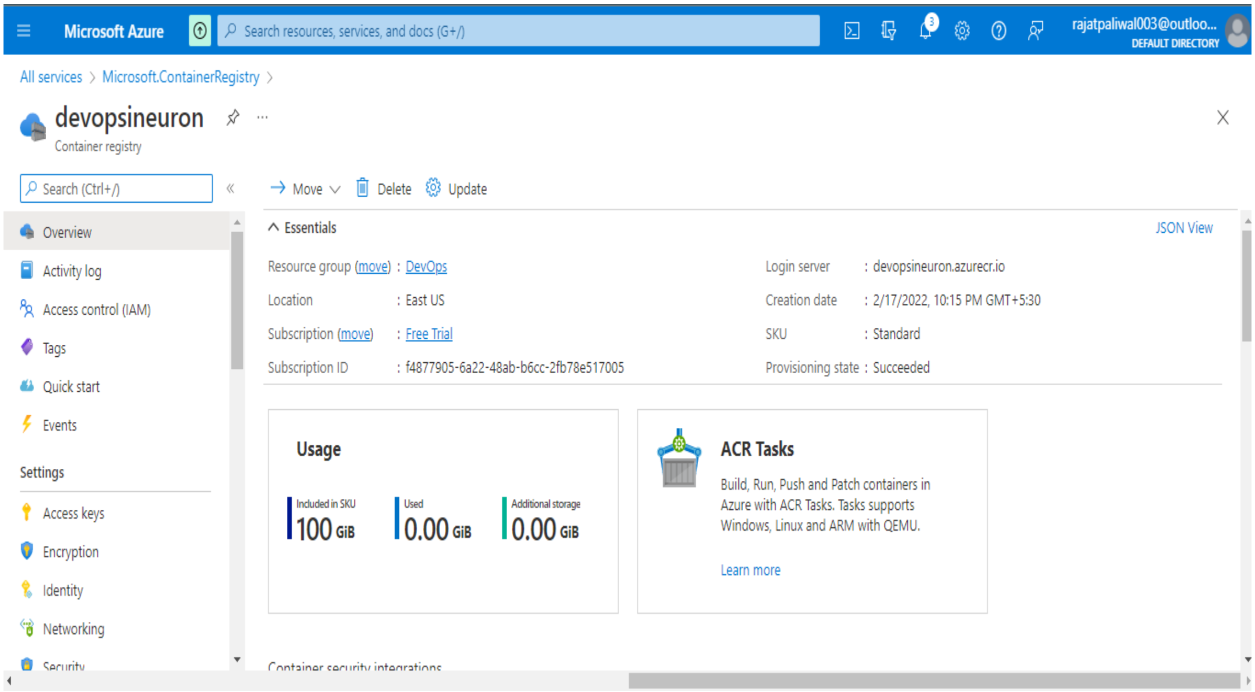
Basics

Subscription	Free Trial
Resource group	(new) DevOps
Virtual machine name	demoproj
Region	East US (free services eligible)
Availability options	No infrastructure redundancy required
Security type	Standard
Image	Ubuntu Server 20.04 LTS - Gen2
Size	Standard D2s v3 (2 vcpus, 8 GiB memory)
Authentication type	Password
Username	rajat
Public inbound ports	SSH, HTTP
Azure Spot	No

Disks

Step 2 - Create Azure Container Registry

We have created a resource group. Now for every new resource, we will add to this resource group.



devopsineuron

Container registry

Essentials

Resource group (move)	: DevOps		Login server	: devopsineuron.azurecr.io
Location	: East US		Creation date	: 2/17/2022, 10:15 PM GMT+5:30
Subscription (move)	: Free Trial		SKU	: Standard
Subscription ID	: f4877905-6a22-48ab-b6cc-2fb78e517005		Provisioning state	: Succeeded

Usage

Included in SKU	Used	Additional storage
100 GiB	0.00 GiB	0.00 GiB

ACR Tasks

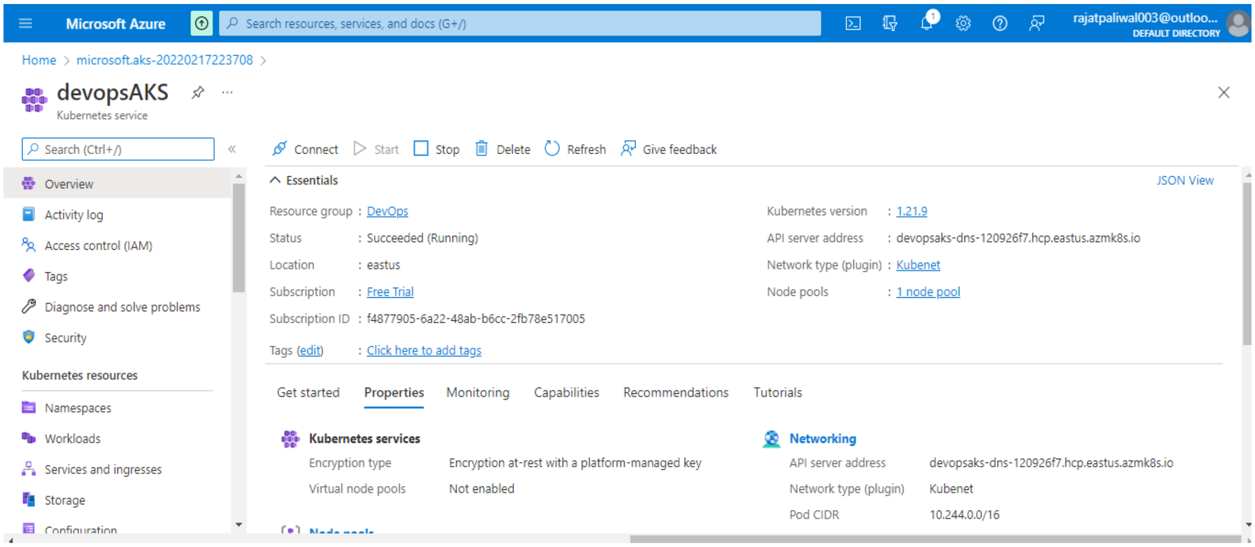
Build, Run, Push and Patch containers in Azure with ACR Tasks. Tasks supports Windows, Linux and ARM with QEMU.

Learn more

Container security integrations

Step 3 - Create Azure Kubernetes Cluster

Click on create a resource and choose Kubernetes Service.



Resource group : DevOps

Status : Succeeded (Running)

Location : eastus

Subscription : Free Trial

Subscription ID : f4877905-6a22-48ab-b6cc-2fb78e517005

Tags (edit) : Click here to add tags

Kubernetes version : 1.21.9

API server address : devopsaks-dns-120926f7.hcp.eastus.azmk8s.io

Network type (plugin) : Kubenet

Node pools : 1 node pool

Kubernetes services

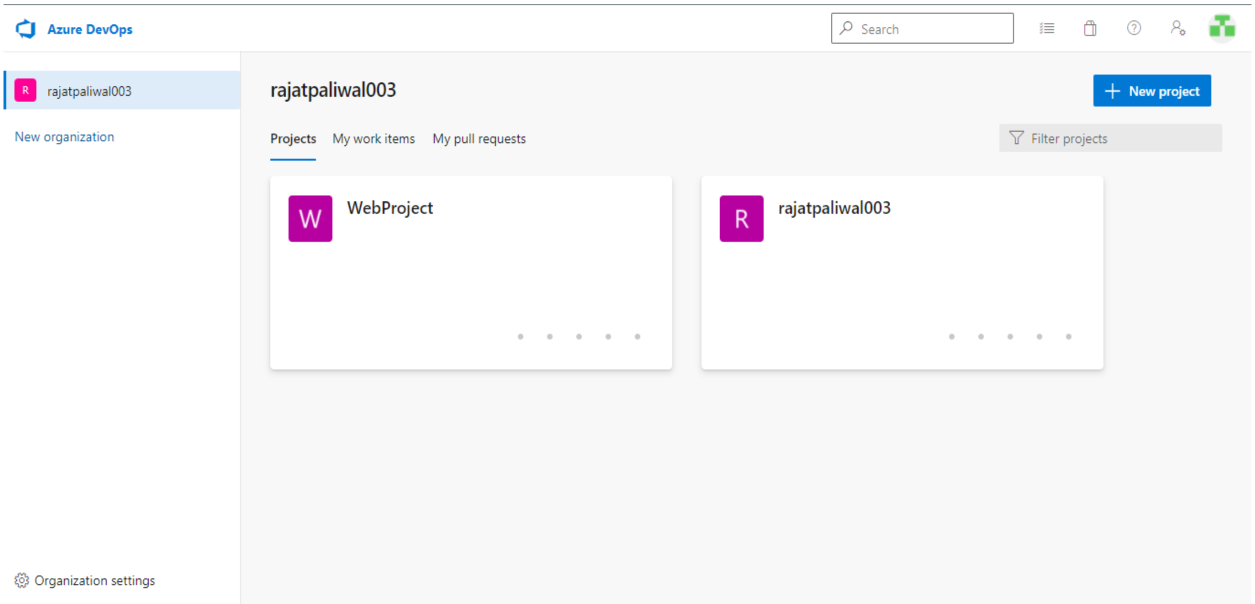
Encryption type	Encryption at-rest with a platform-managed key
Virtual node pools	Not enabled

Networking

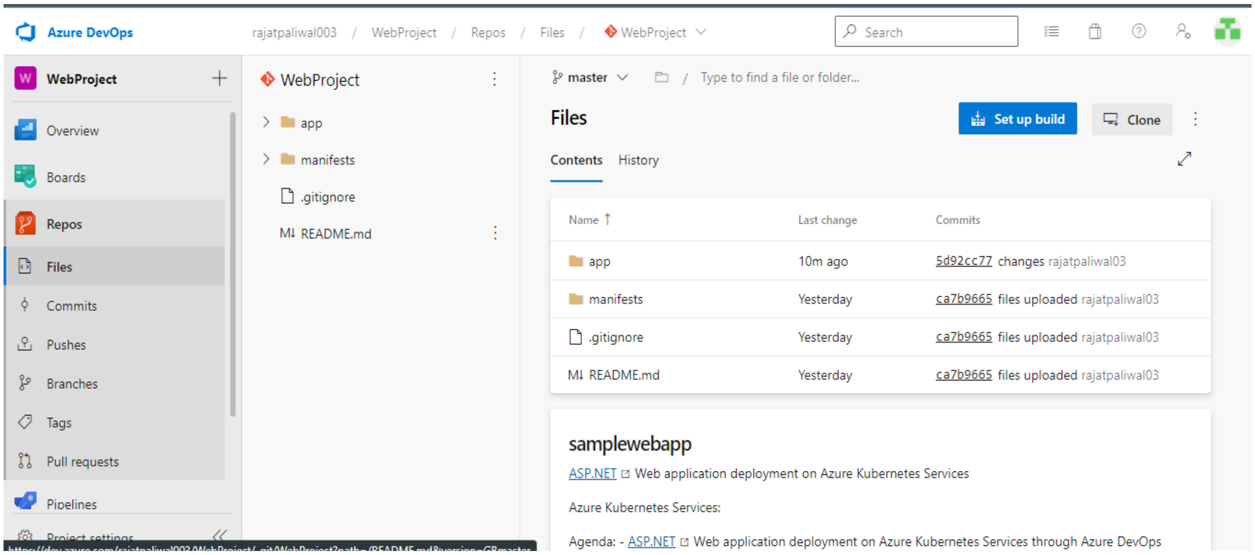
API server address	devopsaks-dns-120926f7.hcp.eastus.azmk8s.io
Network type (plugin)	Kubenet
Pod CIDR	10.244.0.0/16

Step 4 - Create a Pipeline For Deployment to Kubernetes

Login to your DevOps accounts and create a new project named WebProject.



Open this newly created project and add source code to this project repo.



Name ↑	Last change	Commits
app	10m ago	5d92cc77 changes rajatpaliwal03
manifests	Yesterday	ca7b9665 files uploaded rajatpaliwal03
.gitignore	Yesterday	ca7b9665 files uploaded rajatpaliwal03
README.md	Yesterday	ca7b9665 files uploaded rajatpaliwal03

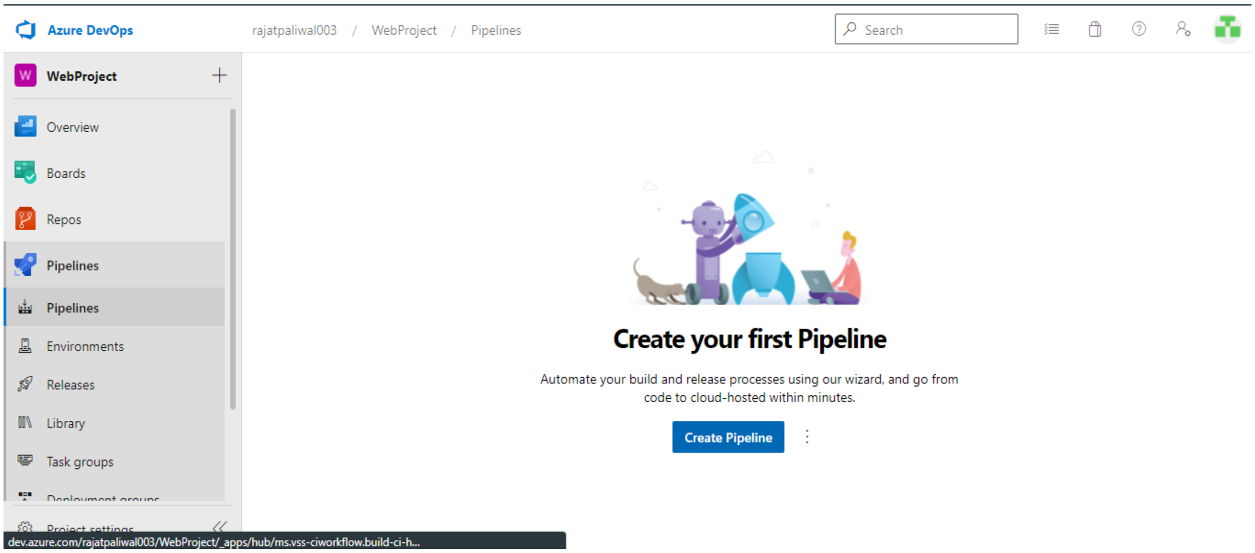
samplewebapp

ASP.NET Web application deployment on Azure Kubernetes Services

Azure Kubernetes Services:

Agenda: - ASP.NET Web application deployment on Azure Kubernetes Services through Azure DevOps

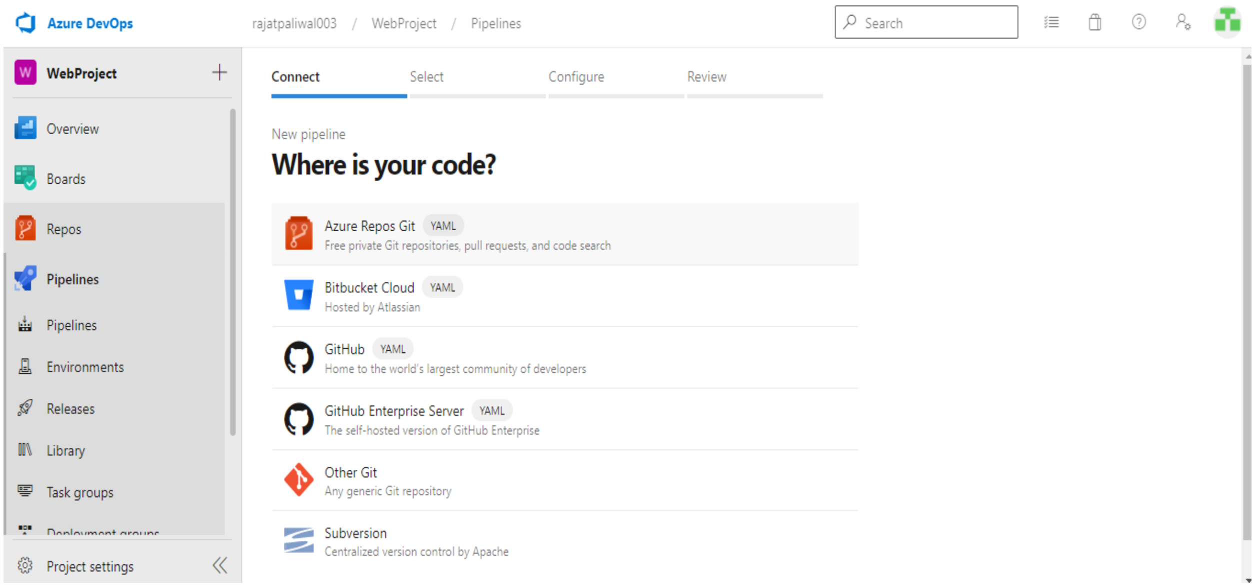
Next, click on the Pipeline tab and then choose the pipelines. This pipeline will build the source code and then create the docker image and will push to the Azure container registry that we created in the last step and create a new deployment to the Azure Kubernetes cluster. Click on create pipeline and choose the source code option. As I mentioned earlier that, I am using Azure Repos for my source code. I will select the Azure repo git option.



Create your first Pipeline

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

dev.azure.com/rajatpaliwal003/WebProject/_apps/hub/ms.vss-ciworkflow.build-ci-h...



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

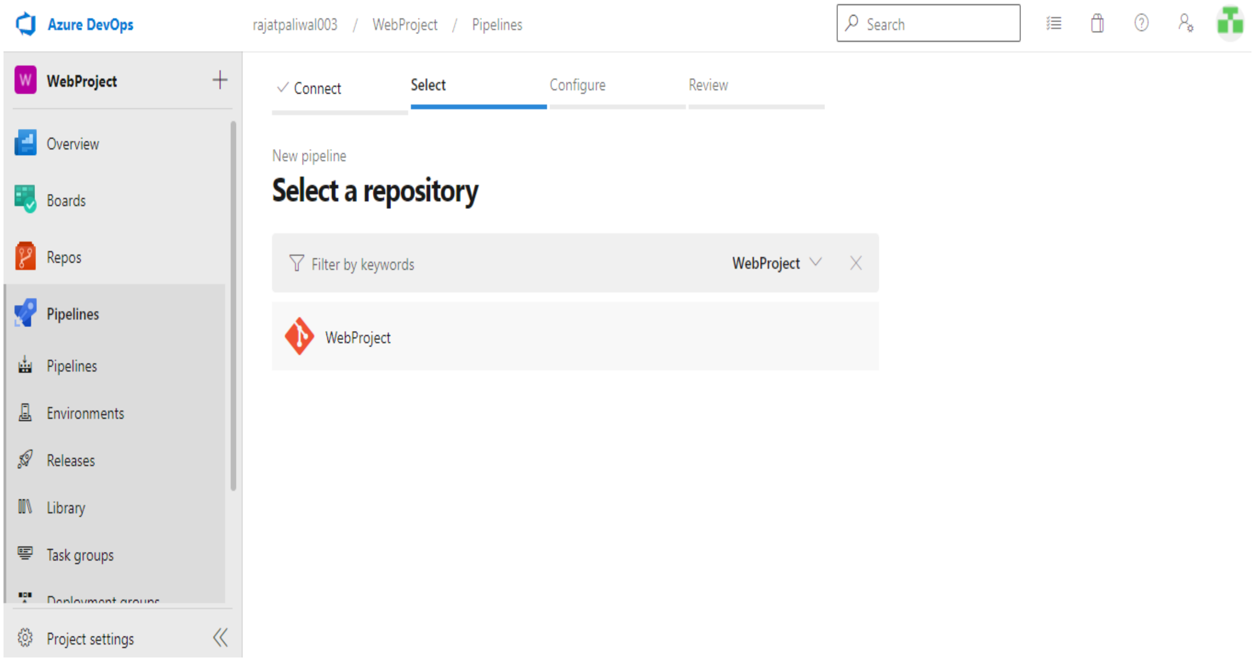
GitHub YAML
Home to the world's largest community of developers

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

Other Git
Any generic Git repository

Subversion
Centralized version control by Apache

Select your repository and then branch. In my case, I will choose *WebProject*.



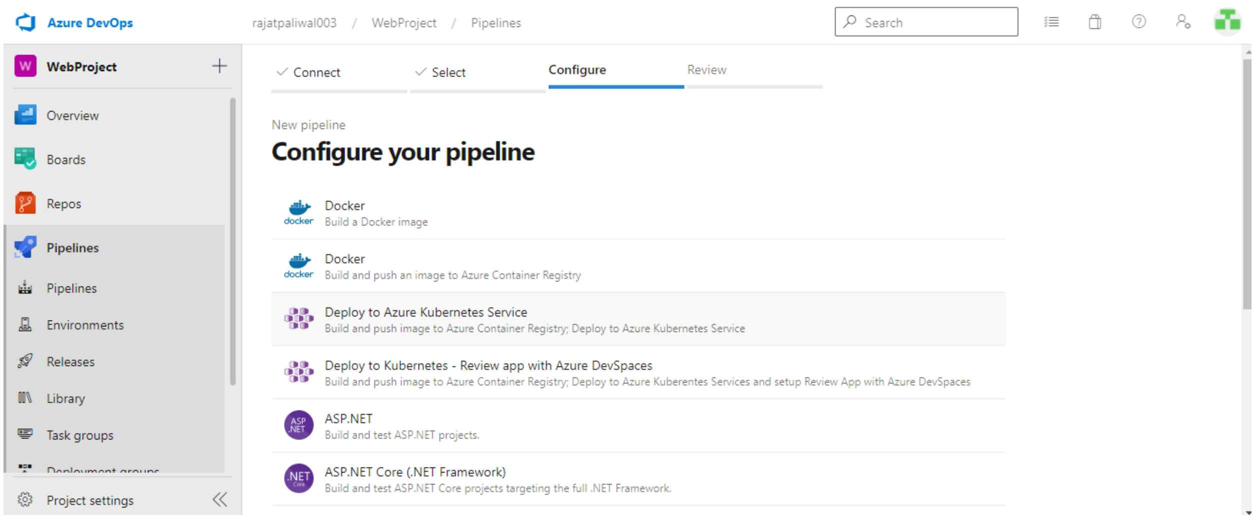
New pipeline

Select a repository

Filter by keywords WebProject

WebProject

Next, select the Deploy to Azure Kubernetes Service (build and push the image to Azure Container Registry; Deploy to Azure Kubernetes Service) option from the Configure your pipeline page.



New pipeline

Configure your pipeline

Docker
Build a Docker image

Docker
Build and push an image to Azure Container Registry

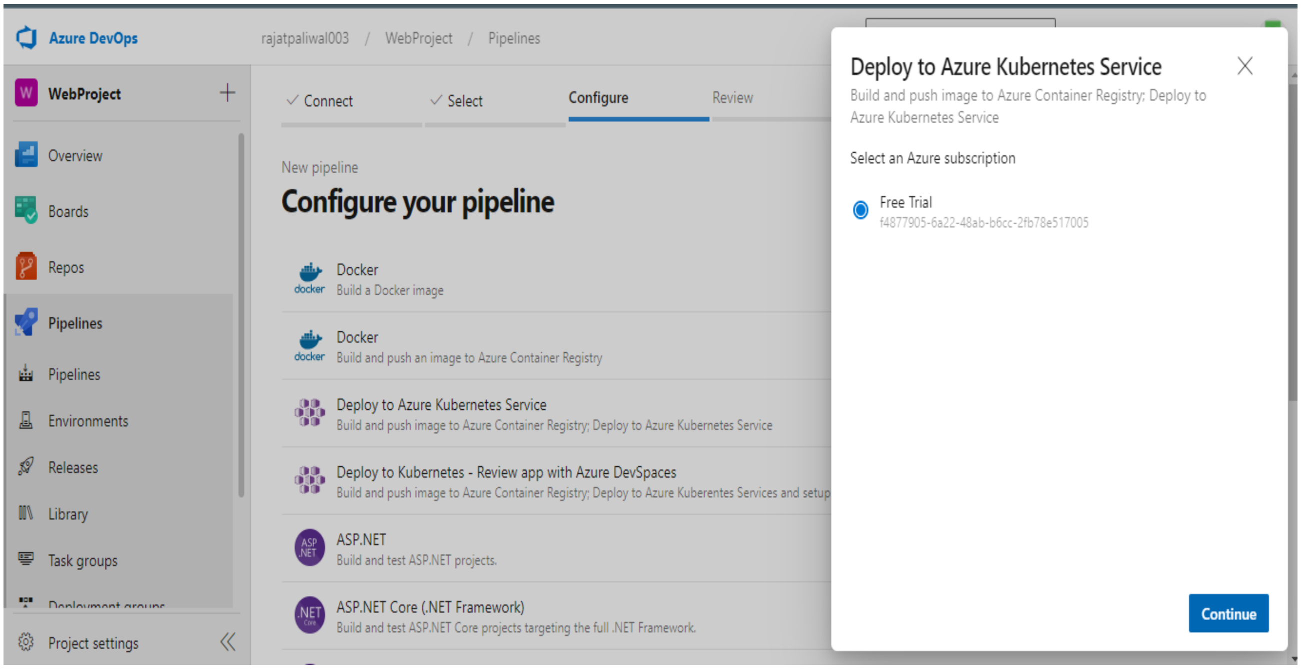
Deploy to Azure Kubernetes Service
Build and push image to Azure Container Registry; Deploy to Azure Kubernetes Service

Deploy to Kubernetes - Review app with Azure DevSpaces
Build and push image to Azure Container Registry; Deploy to Azure Kuberentes Services and setup Review App with Azure DevSpaces

ASP.NET
Build and test ASP.NET projects.

ASP.NET Core (.NET Framework)
Build and test ASP.NET Core projects targeting the full .NET Framework.

Next, select your active subscription from pop up and hit continue.


Azure DevOps
rajatpaliwal003 / WebProject / Pipelines
WebProject
Overview
Boards
Repos
Pipelines
Pipelines
Environments
Releases
Library
Task groups
Deployment groups
Project settings
✓ Connect
✓ Select
Configure
Review
New pipeline
Configure your pipeline
Docker
Build a Docker image
Docker
Build and push an image to Azure Container Registry
Deploy to Azure Kubernetes Service
Build and push image to Azure Container Registry; Deploy to Azure Kubernetes Service
Deploy to Kubernetes - Review app with Azure DevSpaces
Build and push image to Azure Container Registry; Deploy to Azure Kuberentes Services and setup
ASP.NET
Build and test ASP.NET projects.
ASP.NET Core (.NET Framework)
Build and test ASP.NET Core projects targeting the full .NET Framework.

Deploy to Azure Kubernetes Service

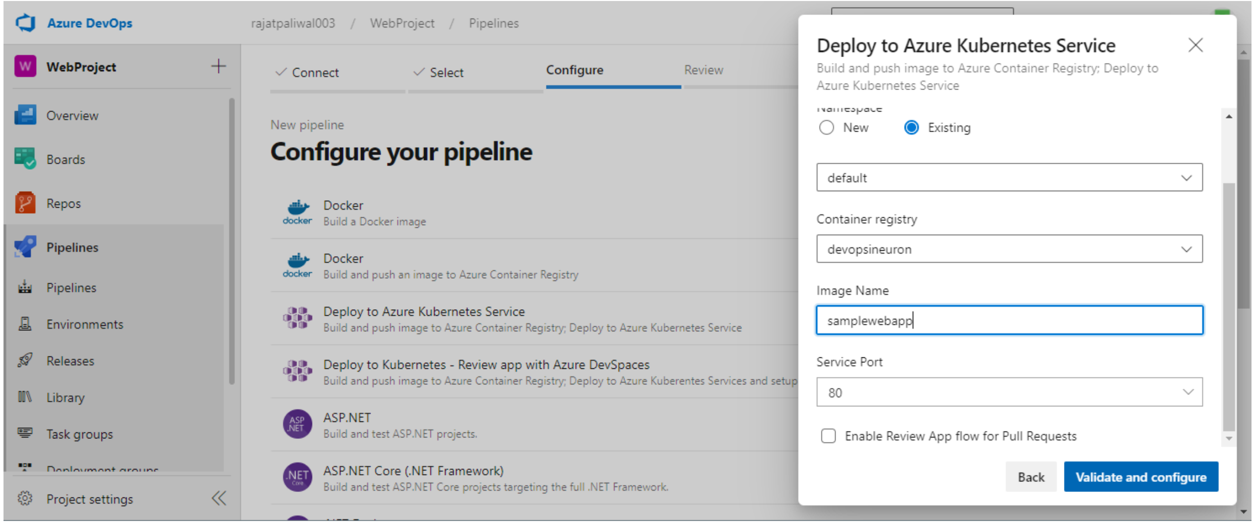
Build and push image to Azure Container Registry; Deploy to Azure Kubernetes Service

Select an Azure subscription

● Free Trial
f4877905-6a22-48ab-b6cc-2fb78e517005

Continue

Next, select your Cluster name from the dropdown and choose the Namespace (existing or new), and select the container registry and enter the image name that you want to use. The service Port option leaves it with a default value. Finally, click on validate and configure.


Azure DevOps
rajatpaliwal003 / WebProject / Pipelines
WebProject
Overview
Boards
Repos
Pipelines
Pipelines
Environments
Releases
Library
Task groups
Project settings
✓ Connect
✓ Select
Configure
Review
New pipeline
Configure your pipeline
Docker
Build a Docker image
Docker
Build and push an image to Azure Container Registry
Deploy to Azure Kubernetes Service
Build and push image to Azure Container Registry; Deploy to Azure Kubernetes Service
Deploy to Kubernetes - Review app with Azure DevSpaces
Build and push image to Azure Container Registry; Deploy to Azure Kuberentes Services and setup
ASP.NET
Build and test ASP.NET projects.
ASP.NET Core (.NET Framework)
Build and test ASP.NET Core projects targeting the full .NET Framework.

Deploy to Azure Kubernetes Service

Build and push image to Azure Container Registry: Deploy to Azure Kubernetes Service

Namespace

○ New ● Existing

default

Container registry: devopsineuron

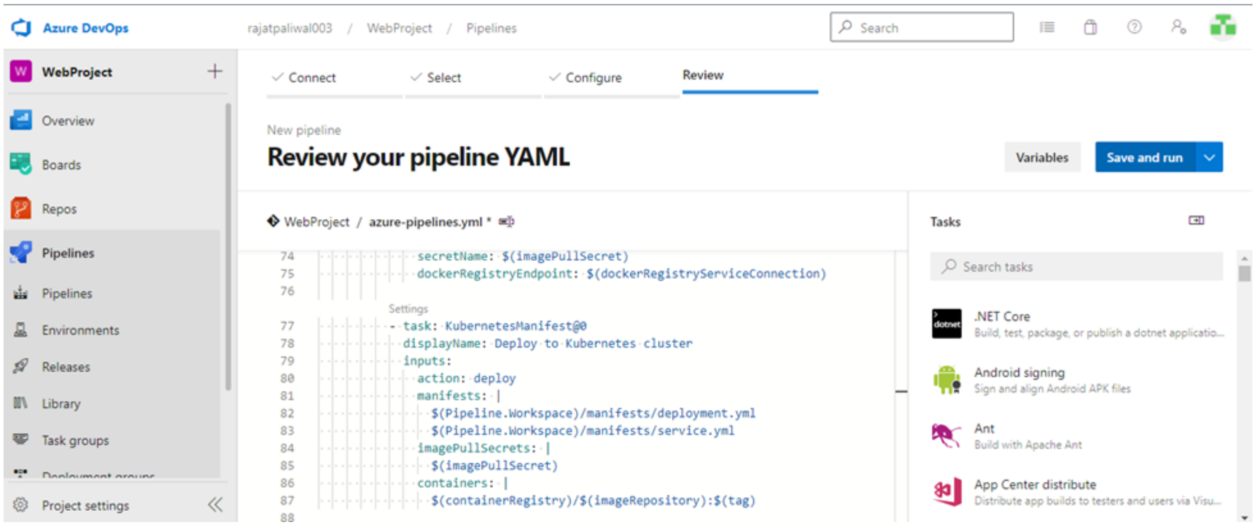
Image Name: samplewebapp

Service Port: 80

□ Enable Review App flow for Pull Requests

Back
Validate and configure

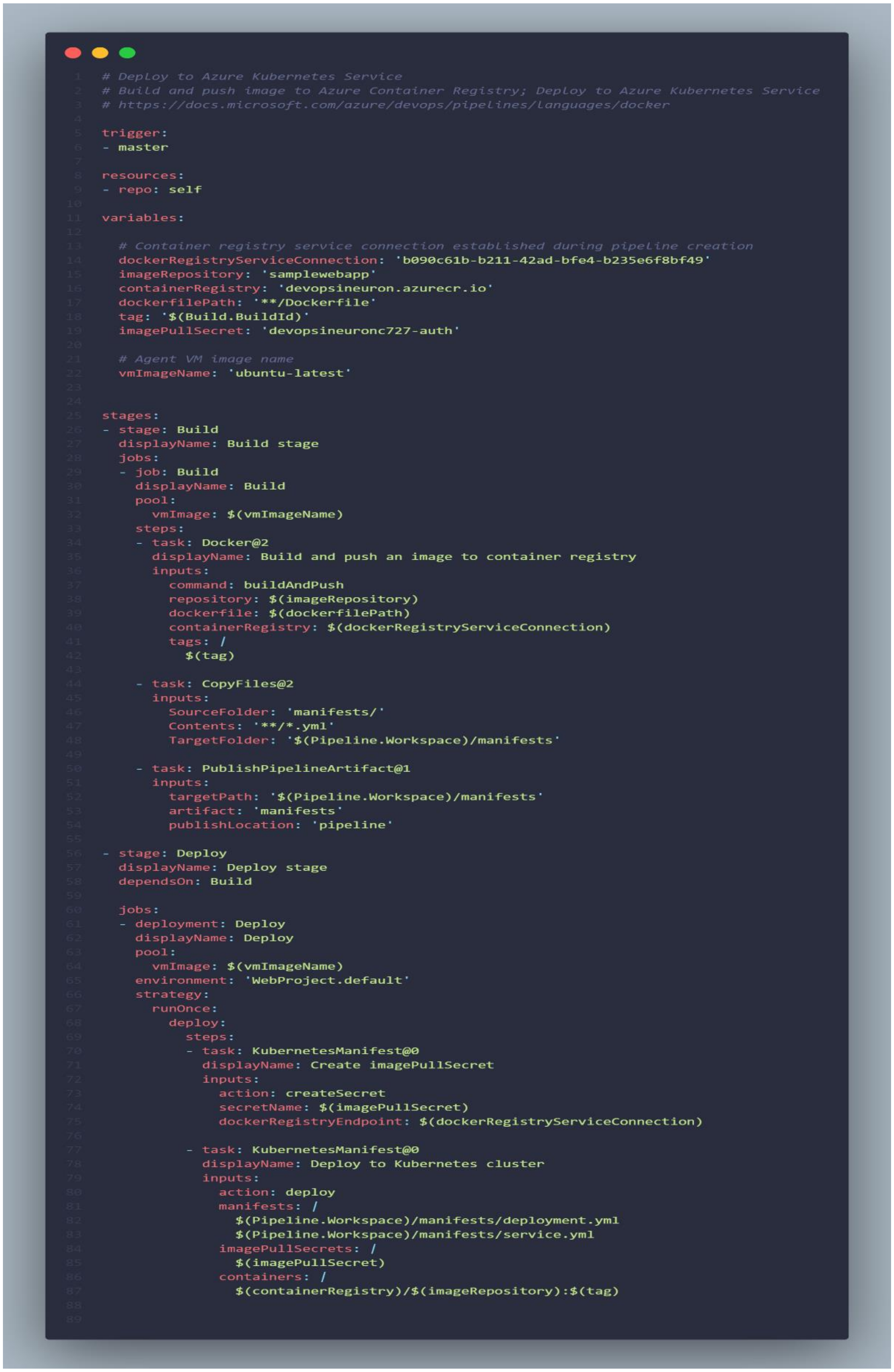
Next, you will see an azure-pipelines.yml file that is a predefined template for building and pushing images to the Azure container registry and deploying them to the Kubernetes cluster. Just update the branch trigger name main to master or any branch name that you want to use for this build pipeline. Now we are good to go and we can save this pipeline as-is. Click on save and run.



WebProject / azure-pipelines.yml *

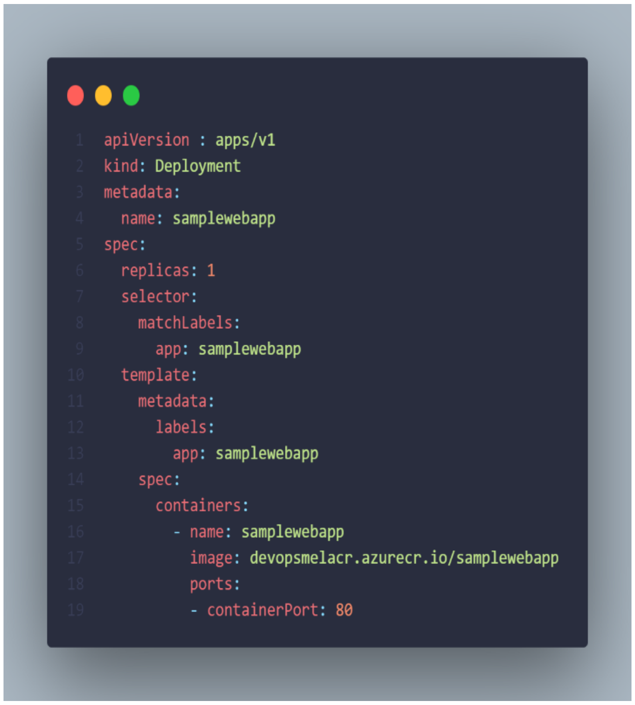
```
            secretName: $(imagePullSecret)
            dockerRegistryEndpoint: $(dockerRegistryServiceConnection)

          Settings
        - task: KubernetesManifest@0
          displayName: Deploy to Kubernetes cluster
          inputs:
            action: deploy
            manifests: |
              $(Pipeline.Workspace)/manifests/deployment.yml
              $(Pipeline.Workspace)/manifests/service.yml
            imagePullSecrets: |
              $(imagePullSecret)
            containers: |
              $(containerRegistry)/$(imageRepository):$(tag)
```

azure-pipelines.yml :

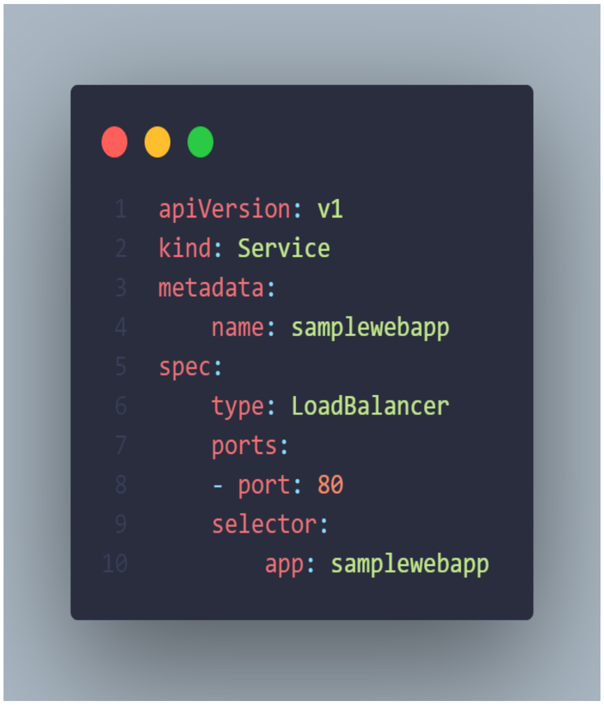


```yaml
# Deploy to Azure Kubernetes Service
# Build and push image to Azure Container Registry; Deploy to Azure Kubernetes Service
# https://docs.microsoft.com/azure/devops/pipelines/languages/docker

trigger:
- master

resources:
- repo: self

variables:

  # Container registry service connection established during pipeline creation
  dockerRegistryServiceConnection: 'b090c61b-b211-42ad-bfe4-b235e6f8bf49'
  imageRepository: 'samplewebapp'
  containerRegistry: 'devopsineuron.azurecr.io'
  dockerfilePath: '**/Dockerfile'
  tag: '$(Build.BuildId)'
  imagePullSecret: 'devopsineuronc727-auth'

  # Agent VM image name
  vmImageName: 'ubuntu-latest'


stages:
- stage: Build
  displayName: Build stage
  jobs:
  - job: Build
    displayName: Build
    pool:
      vmImage: $(vmImageName)
    steps:
    - task: Docker@2
      displayName: Build and push an image to container registry
      inputs:
        command: buildAndPush
        repository: $(imageRepository)
        dockerfile: $(dockerfilePath)
        containerRegistry: $(dockerRegistryServiceConnection)
        tags: /
          $(tag)

    - task: CopyFiles@2
      inputs:
        SourceFolder: 'manifests/'
        Contents: '**/*.yml'
        TargetFolder: '$(Pipeline.Workspace)/manifests'

    - task: PublishPipelineArtifact@1
      inputs:
        targetPath: '$(Pipeline.Workspace)/manifests'
        artifact: 'manifests'
        publishLocation: 'pipeline'

- stage: Deploy
  displayName: Deploy stage
  dependsOn: Build

  jobs:
  - deployment: Deploy
    displayName: Deploy
    pool:
      vmImage: $(vmImageName)
    environment: 'WebProject.default'
    strategy:
      runOnce:
        deploy:
          steps:
          - task: KubernetesManifest@0
            displayName: Create imagePullSecret
            inputs:
              action: createSecret
              secretName: $(imagePullSecret)
              dockerRegistryEndpoint: $(dockerRegistryServiceConnection)

          - task: KubernetesManifest@0
            displayName: Deploy to Kubernetes cluster
            inputs:
              action: deploy
              manifests: /
                $(Pipeline.Workspace)/manifests/deployment.yml
                $(Pipeline.Workspace)/manifests/service.yml
              imagePullSecrets: /
                $(imagePullSecret)
              containers: /
                $(containerRegistry)/$(imageRepository):$(tag)
```

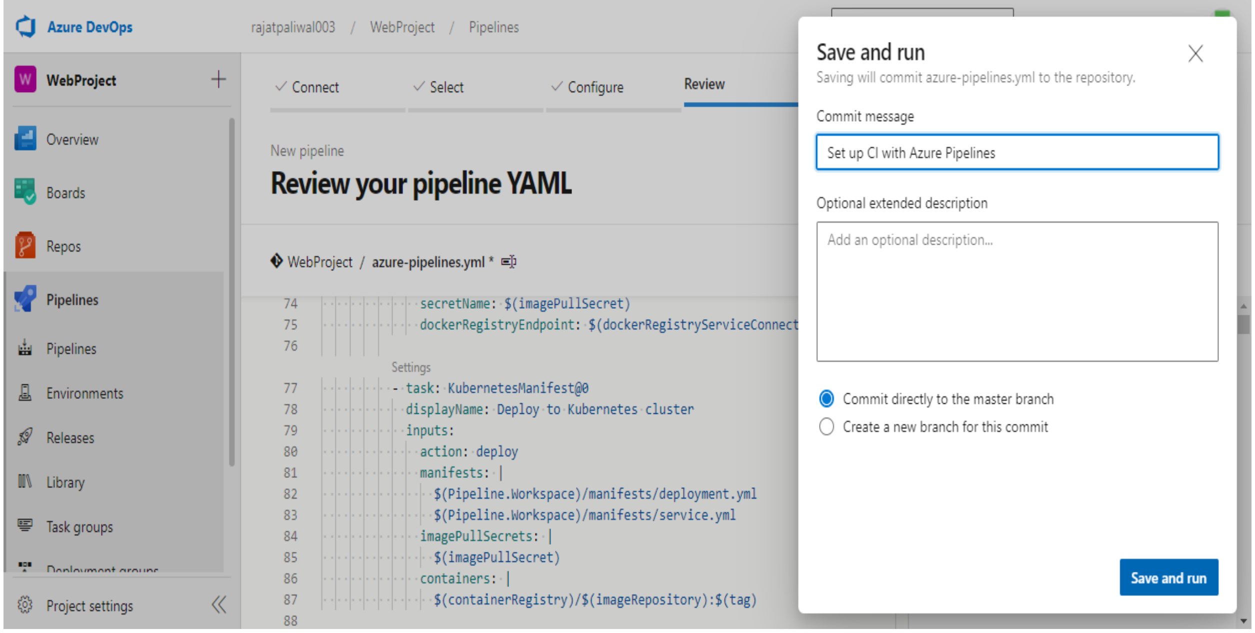
Here you will see in the output popup the three files will be added to our repository (*azure-pipeline, deployment.yml, service.yml*) choose the commit directly to the master branch option and hit Save and Run.



```yaml
apiVersion : apps/v1
kind: Deployment
metadata:
  name: samplewebapp
spec:
  replicas: 1
  selector:
    matchLabels:
      app: samplewebapp
  template:
    metadata:
      labels:
        app: samplewebapp
    spec:
      containers:
        - name: samplewebapp
          image: devopsmelacr.azurecr.io/samplewebapp
          ports:
          - containerPort: 80
```



```yaml
apiVersion: v1
kind: Service
metadata:
    name: samplewebapp
spec:
    type: LoadBalancer
    ports:
    - port: 80
    selector:
        app: samplewebapp
```



New pipeline

Review your pipeline YAML

WebProject / azure-pipelines.yml *

```yaml
            secretName: $(imagePullSecret)
            dockerRegistryEndpoint: $(dockerRegistryServiceConnect

        - task: KubernetesManifest@0
          displayName: Deploy to Kubernetes cluster
          inputs:
            action: deploy
            manifests: |
              $(Pipeline.Workspace)/manifests/deployment.yml
              $(Pipeline.Workspace)/manifests/service.yml
            imagePullSecrets: |
              $(imagePullSecret)
            containers: |
              $(containerRegistry)/$(imageRepository):$(tag)

```

Save and run

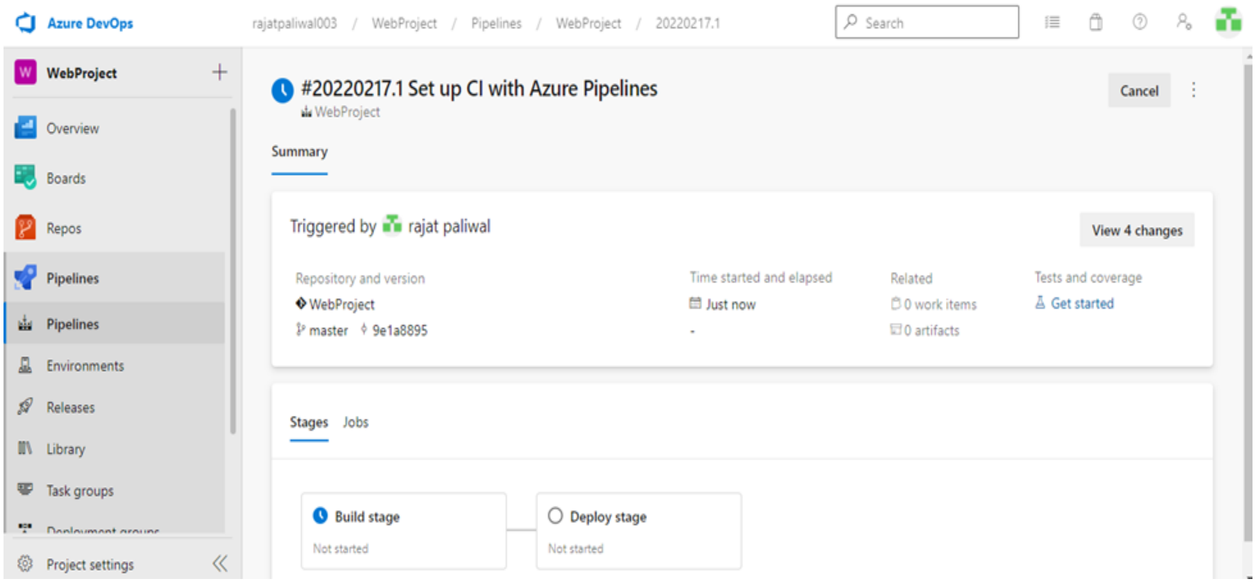
Saving will commit azure-pipelines.yml to the repository.

Commit message: Set up CI with Azure Pipelines

Optional extended description: Add an optional description...

● Commit directly to the master branch
○ Create a new branch for this commit

After Save and Run, you will see two stages. In the first stage, our code will be built and the docker image will be created and pushed to the Azure container registry. In the second stage, it will be deployed to the Kubernetes cluster.



#20220217.1 Set up CI with Azure Pipelines
WebProject

Triggered by rajat paliwal

Repository and version: WebProject, master, 9e1a8895

Time started and elapsed: Just now, -

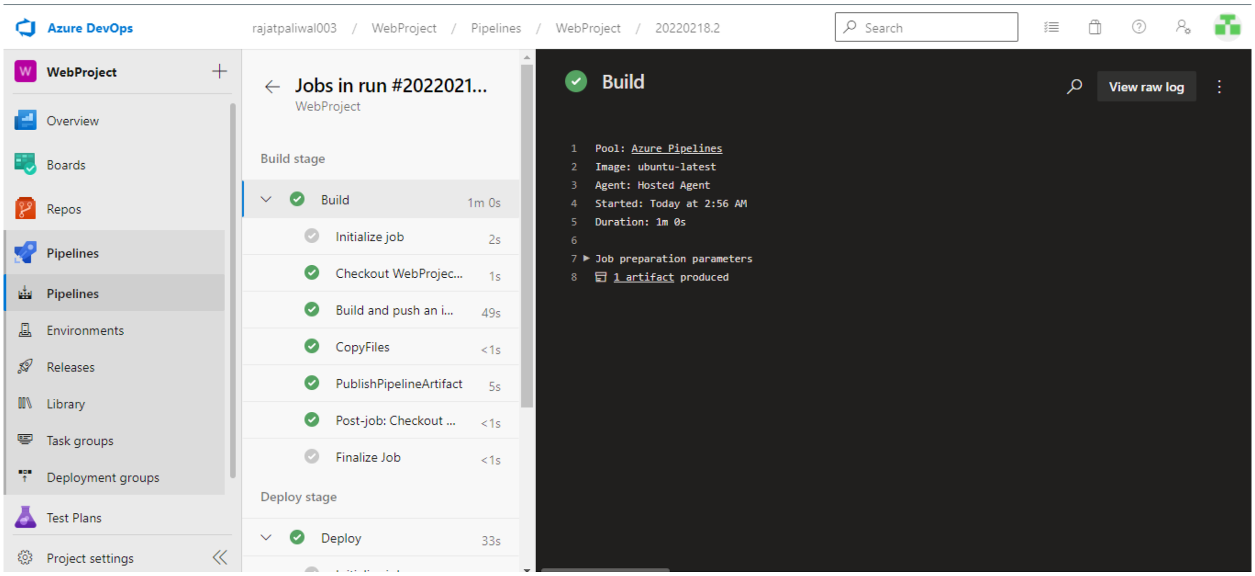
Related: 0 work items, 0 artifacts

Tests and coverage: Get started

Build stage: Not started

Deploy stage: Not started

The stage of building an image and pushing it to the Azure container registry is done.



Jobs in run #2022021...
WebProject

Build stage

Build	1m 0s
Initialize job	2s
Checkout WebProjec...	1s
Build and push an i...	49s
CopyFiles	<1s
PublishPipelineArtifact	5s
Post-job: Checkout ...	<1s
Finalize Job	<1s

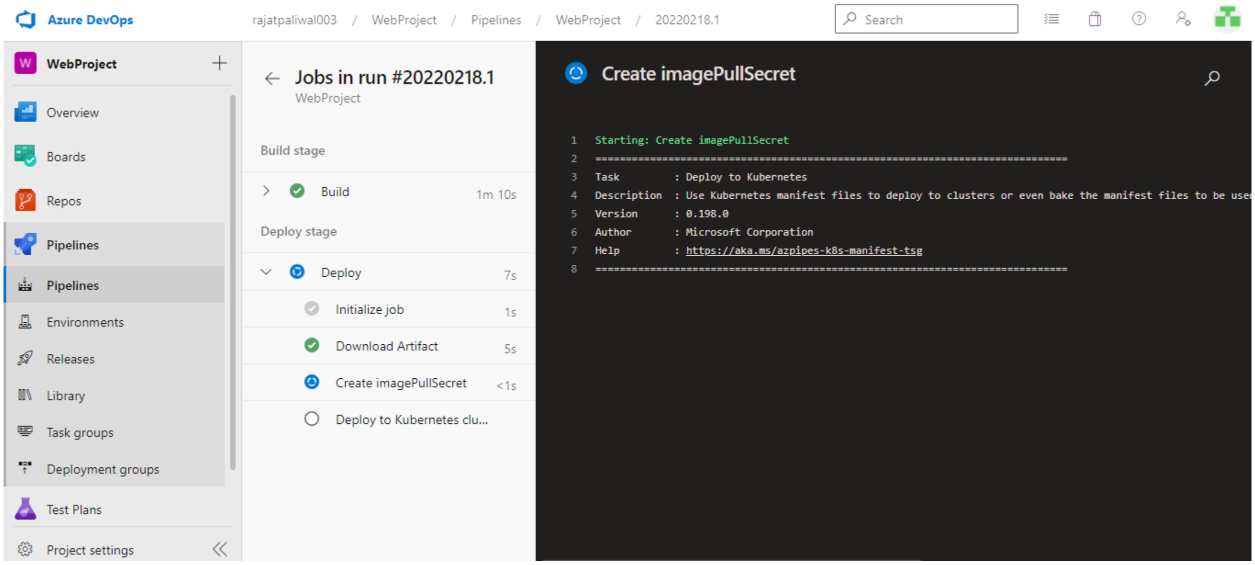
Deploy stage

Deploy	33s

Build

```
Pool: Azure Pipelines
Image: ubuntu-latest
Agent: Hosted Agent
Started: Today at 2:56 AM
Duration: 1m 0s

▶ Job preparation parameters
1 artifact produced
```

The process of deploying to the Kubernetes cluster is in progress.



rajatpaliwal003 / WebProject / Pipelines / WebProject / 20220218.1

Jobs in run #20220218.1
WebProject

Build stage

- Build 1m 10s

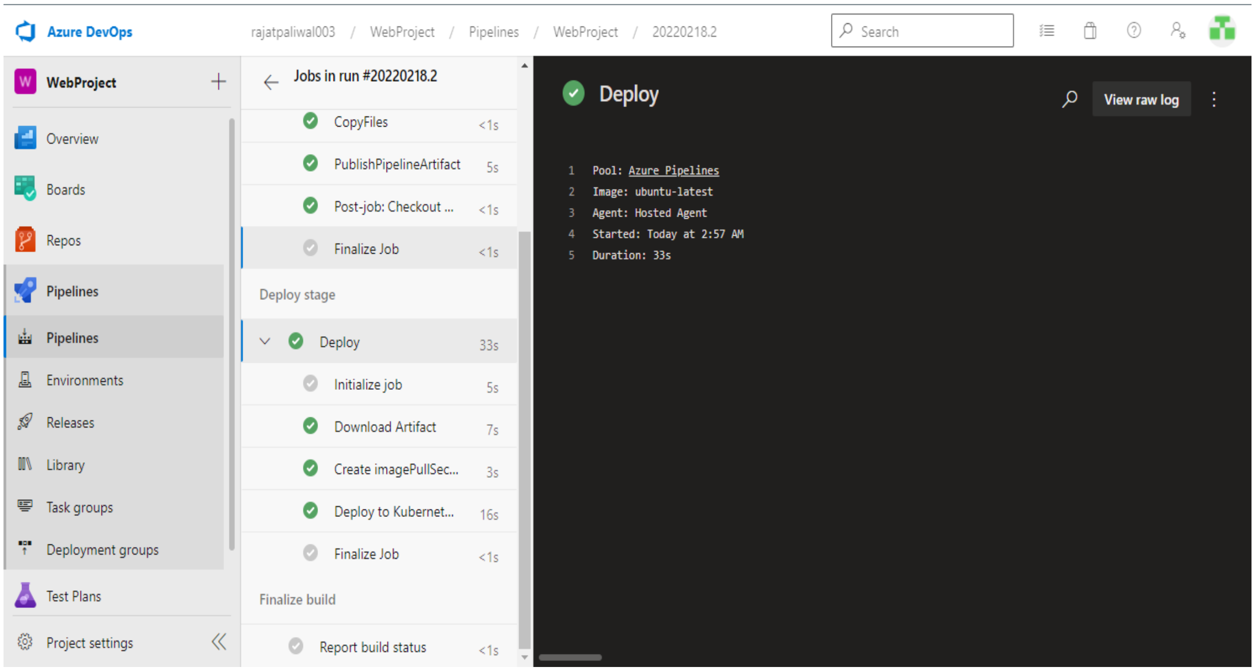
Deploy stage

- Deploy 7s
  - Initialize job 1s
  - Download Artifact 5s
  - Create imagePullSecret <1s
  - Deploy to Kubernetes clu...

Create imagePullSecret

```
Starting: Create imagePullSecret
==============================================================================
Task         : Deploy to Kubernetes
Description  : Use Kubernetes manifest files to deploy to clusters or even bake the manifest files to be used
Version      : 0.198.0
Author       : Microsoft Corporation
Help         : https://aka.ms/azpipes-k8s-manifest-tsg
==============================================================================
```

The stage of deploying to the Kubernetes cluster is done.



rajatpaliwal003 / WebProject / Pipelines / WebProject / 20220218.2

Jobs in run #20220218.2

- CopyFiles <1s
- PublishPipelineArtifact 5s
- Post-job: Checkout ... <1s
- Finalize Job <1s

Deploy stage

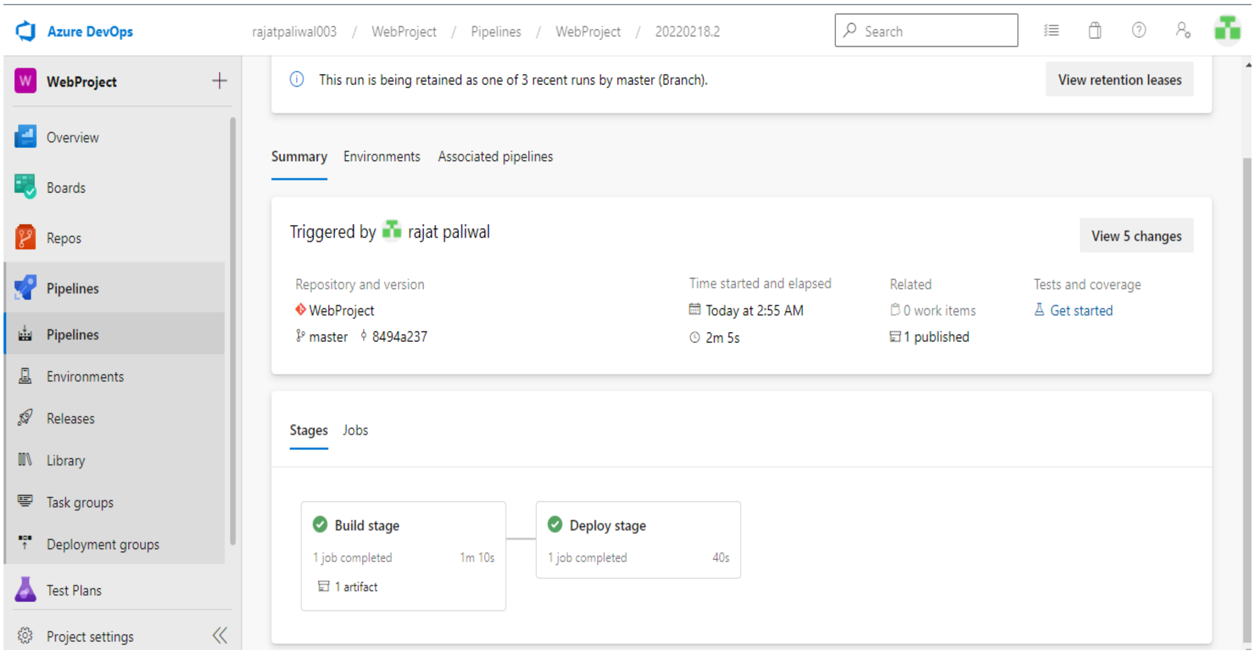
- Deploy 33s
  - Initialize job 5s
  - Download Artifact 7s
  - Create imagePullSec... 3s
  - Deploy to Kubernet... 16s
  - Finalize Job <1s

Finalize build

- Report build status <1s

Deploy

```
Pool: Azure Pipelines
Image: ubuntu-latest
Agent: Hosted Agent
Started: Today at 2:57 AM
Duration: 33s
```



This run is being retained as one of 3 recent runs by master (Branch).

Triggered by rajat paliwal

Repository and version: WebProject, master, 8494a237

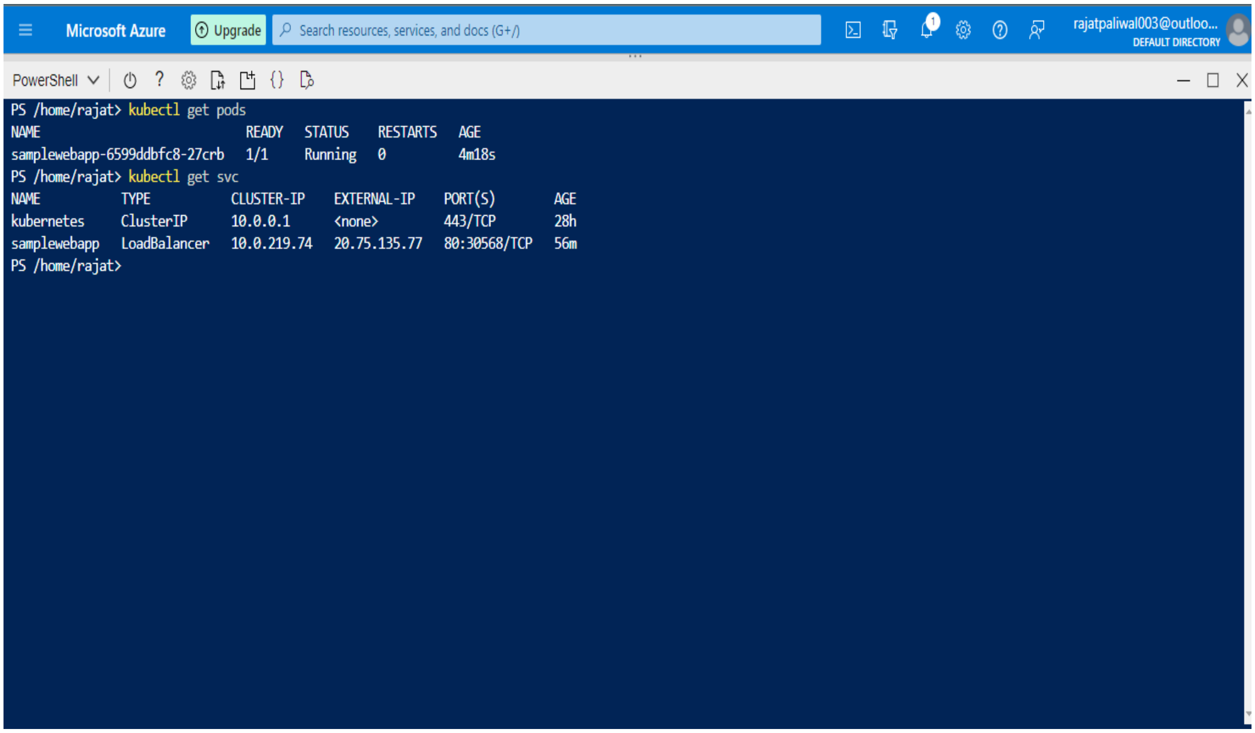
Time started and elapsed: Today at 2:55 AM, 2m 5s

Related: 0 work items, 1 published

Tests and coverage: Get started

Build stage: 1 job completed, 1m 10s, 1 artifact

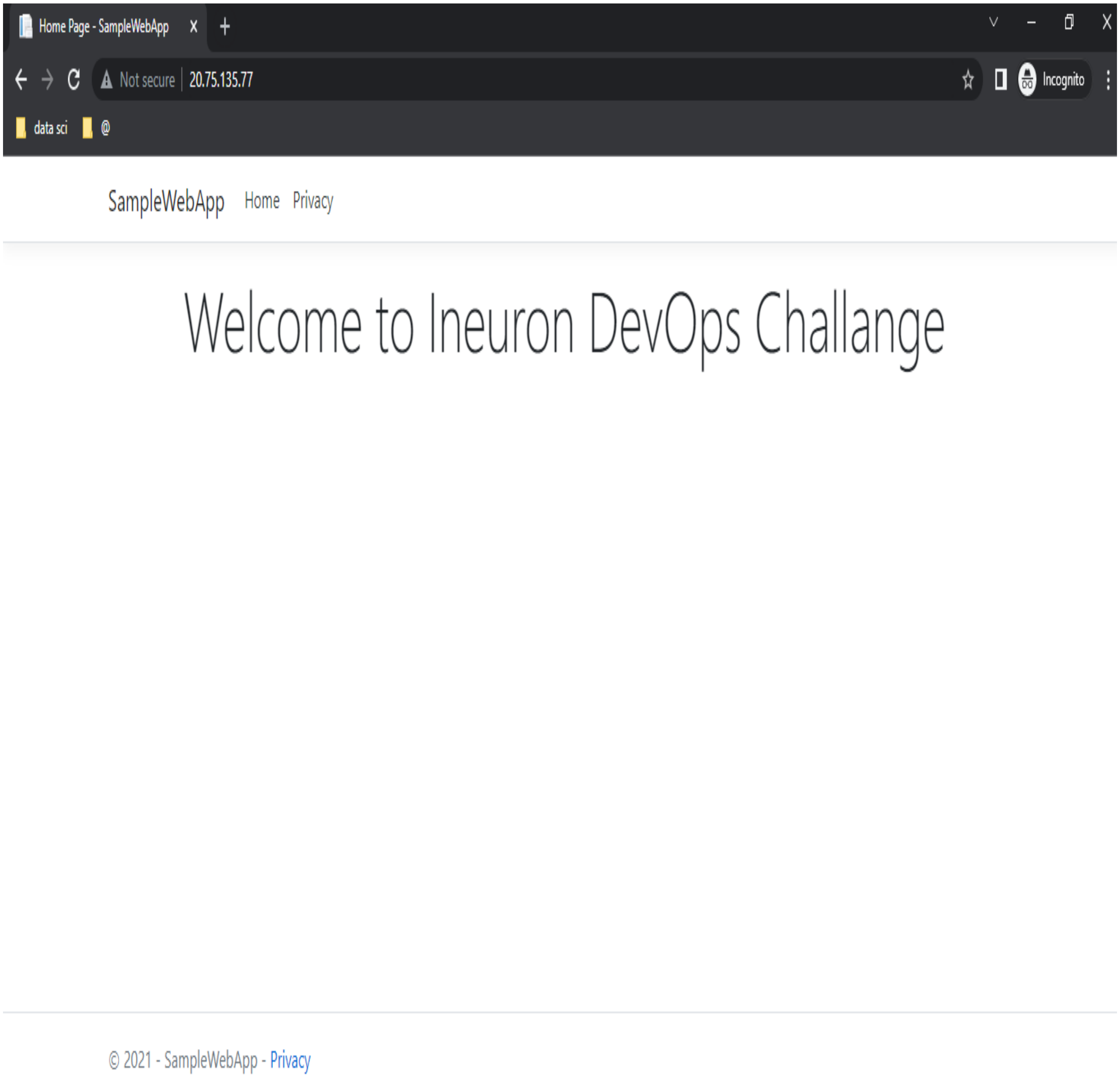
Deploy stage: 1 job completed, 40s


Microsoft Azure
Upgrade
Search resources, services, and docs (G+/)
rajatpaliwal003@outloo...
DEFAULT DIRECTORY
PowerShell

```
PS /home/rajat> kubectl get pods
NAME                          READY   STATUS    RESTARTS   AGE
samplewebapp-6599ddbfc8-27crb   1/1     Running   0          4m18s
PS /home/rajat> kubectl get svc
NAME           TYPE           CLUSTER-IP    EXTERNAL-IP    PORT(S)        AGE
kubernetes     ClusterIP      10.0.0.1      <none>         443/TCP        28h
samplewebapp   LoadBalancer   10.0.219.74   20.75.135.77   80:30568/TCP   56m
PS /home/rajat>
```

Grab the External IP address of your service and open it in any browser. Amazing, the Azure Kubernetes cluster is up and running in the Azure portal. So, this was the process of building a CI/CD pipeline for Azure Kubernetes Cluster.

Web App:

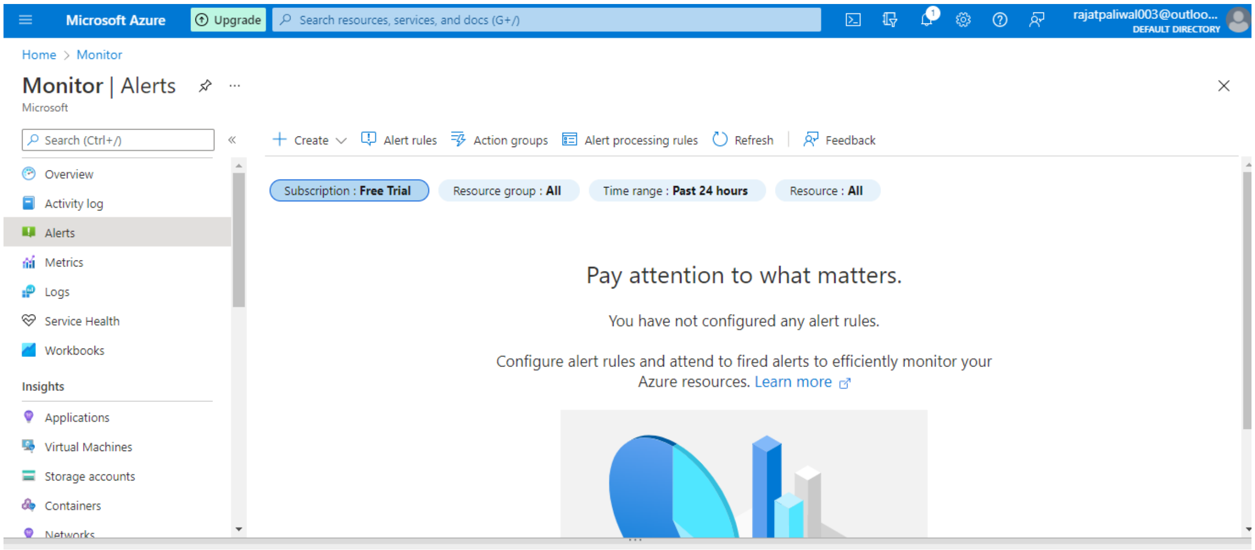


Welcome to Ineuron DevOps Challange

© 2021 - SampleWebApp - Privacy

Monitoring logs and email alerts:

1. In the portal, select the relevant resource.
2. Under **Monitoring**, select **Alerts**.
3. From the top command bar, select **Alert rules**.
4. Select the alert rule that you want to edit.
5. Edit any fields necessary, then select **Save** on the top command bar.

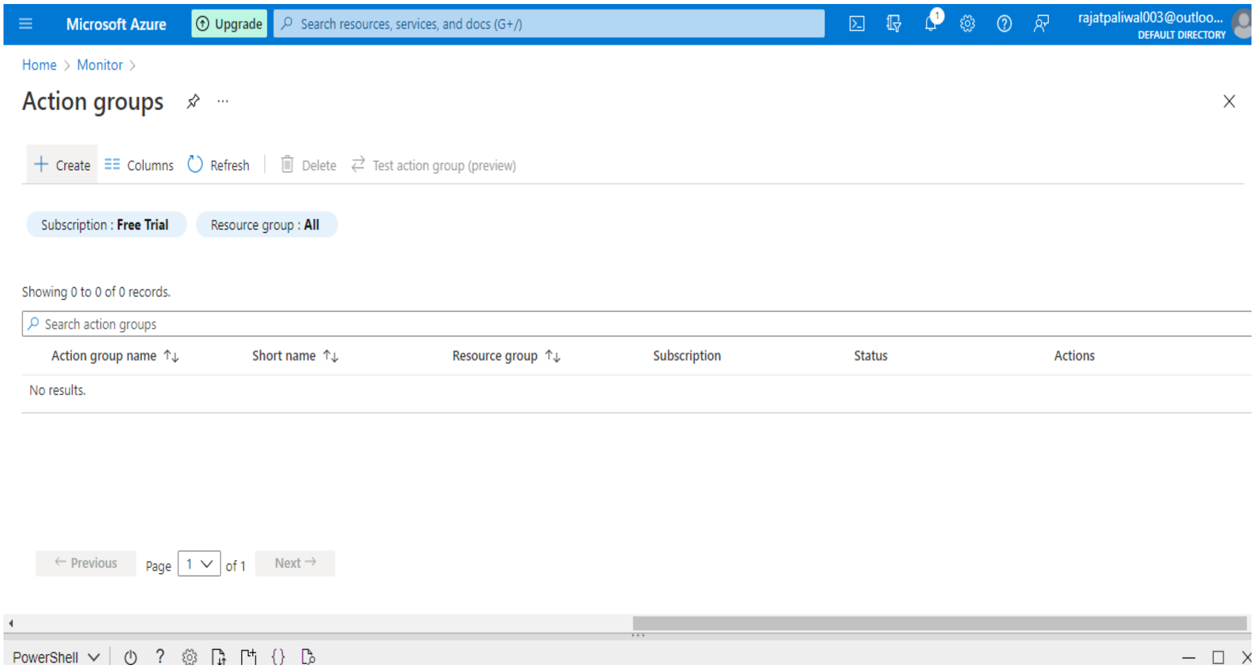


Subscription : **Free Trial** | Resource group : **All** | Time range : **Past 24 hours** | Resource : **All**

Pay attention to what matters.

You have not configured any alert rules.

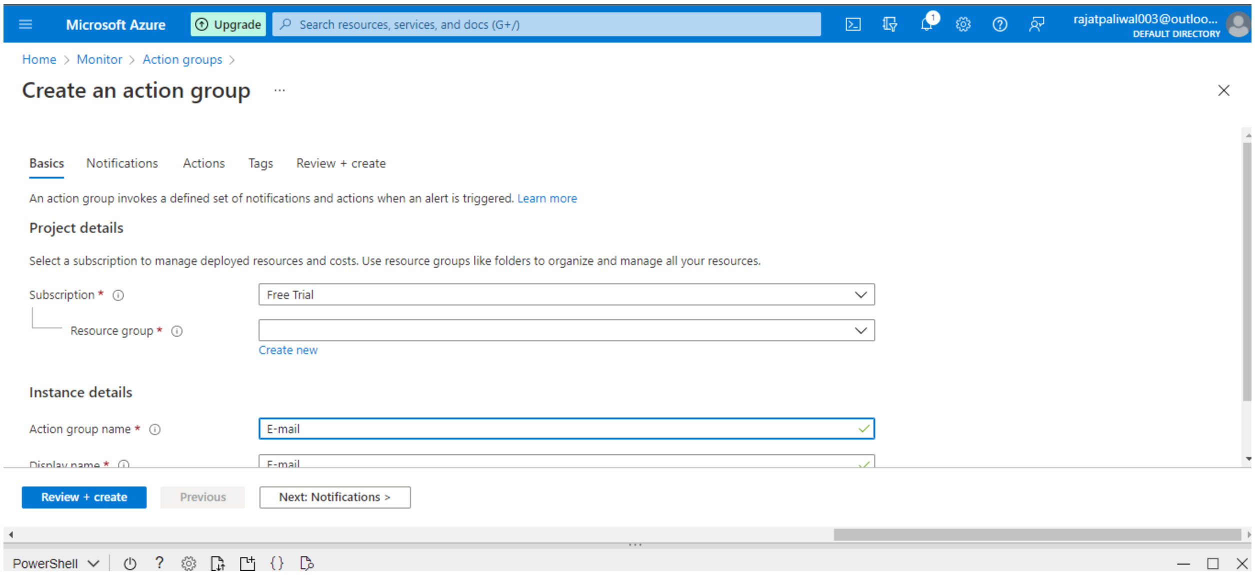
Configure alert rules and attend to fired alerts to efficiently monitor your Azure resources. Learn more



Subscription : **Free Trial** | Resource group : **All**

Showing 0 to 0 of 0 records.

Action group name ↑↓	Short name ↑↓	Resource group ↑↓	Subscription	Status	Actions
No results.					



Create an action group

An action group invokes a defined set of notifications and actions when an alert is triggered. Learn more

Project details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * : Free Trial

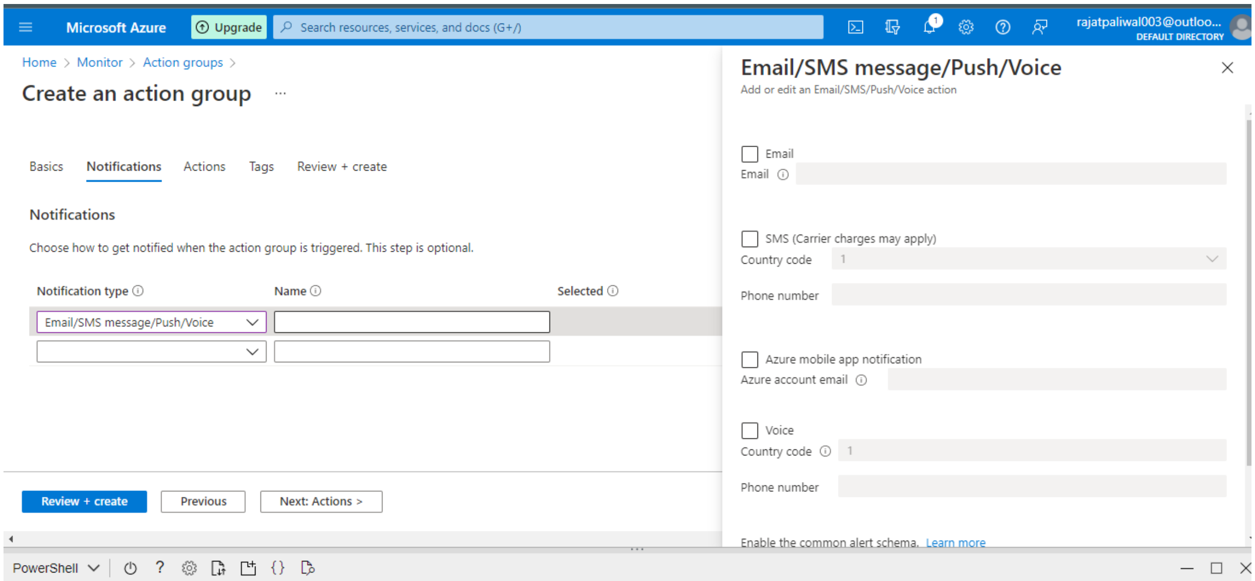
Resource group * :

Create new

Instance details

Action group name * : E-mail

Display name * : E-mail



Create an action group

Notifications

Choose how to get notified when the action group is triggered. This step is optional.

Notification type	Name	Selected
Email/SMS message/Push/Voice		

Email/SMS message/Push/Voice

Add or edit an Email/SMS/Push/Voice action

□ Email

Email

□ SMS (Carrier charges may apply)

Country code 1

Phone number

□ Azure mobile app notification

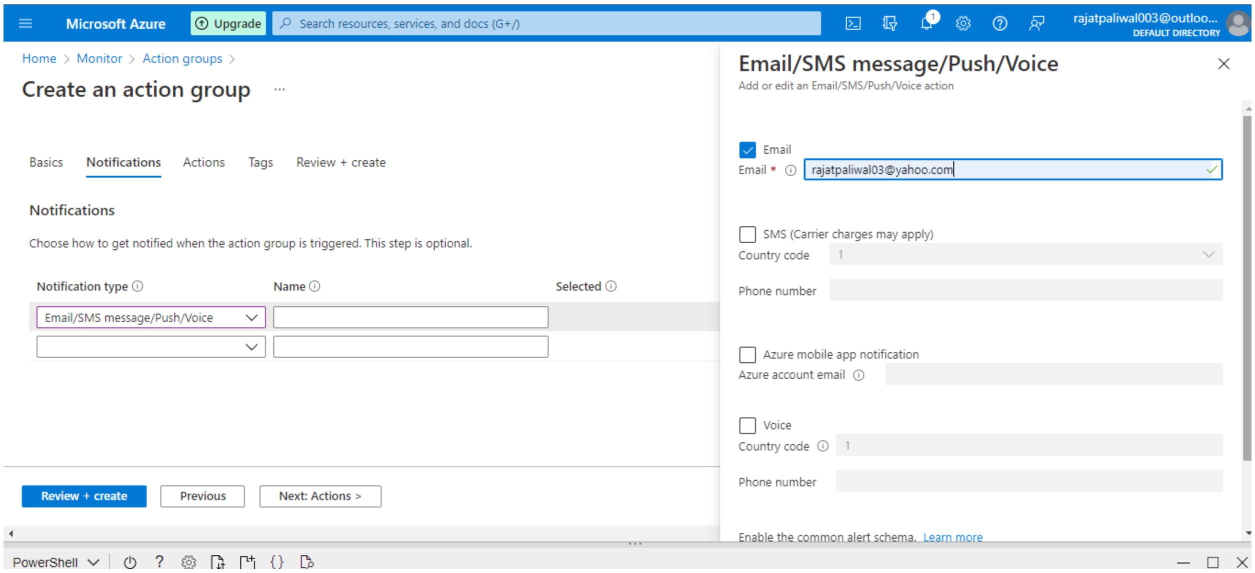
Azure account email

□ Voice

Country code 1

Phone number

Enable the common alert schema. Learn more



Create an action group ...

Notifications

Choose how to get notified when the action group is triggered. This step is optional.

Notification type ⓘ	Name ⓘ	Selected ⓘ
Email/SMS message/Push/Voice		

Email/SMS message/Push/Voice

Add or edit an Email/SMS/Push/Voice action

☑ Email

Email * ⓘ rajatpaliwal03@yahoo.com

□ SMS (Carrier charges may apply)

Country code 1

Phone number

□ Azure mobile app notification

Azure account email ⓘ

□ Voice

Country code ⓘ 1

Phone number

Enable the common alert schema. Learn more

After entering e-mail, click review + create .