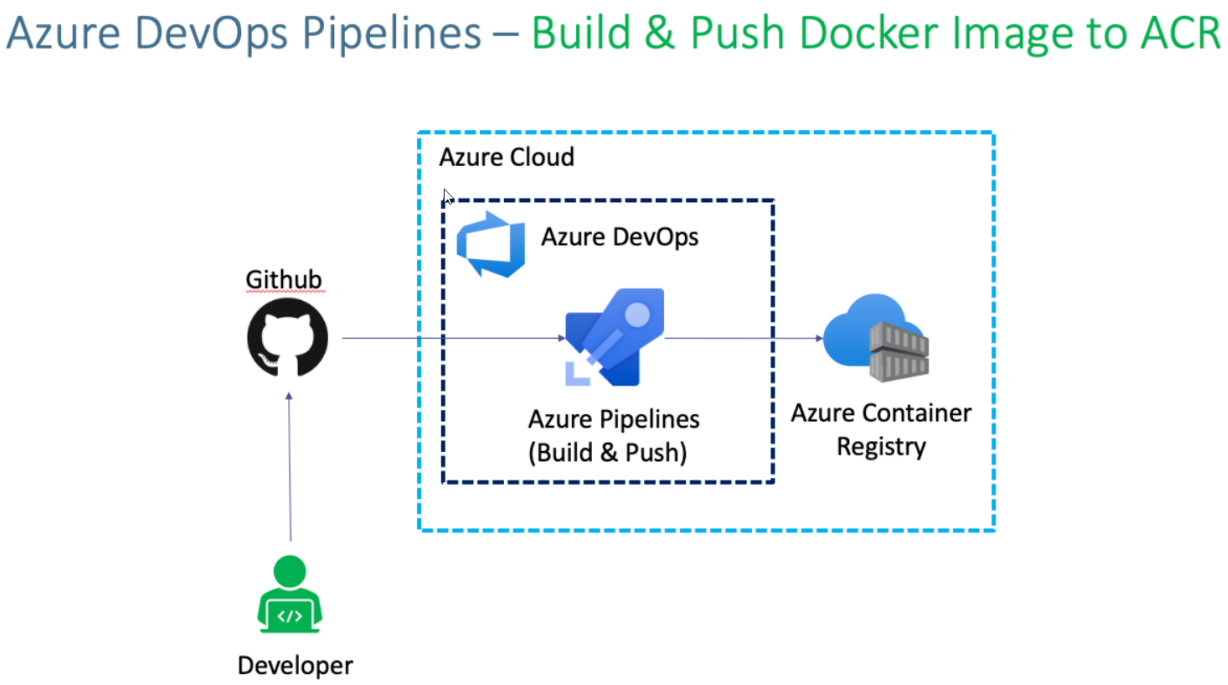
Build Docker Image and Push to Azure container Registry



Step-02: Create Github Project and Check-In Code

Create Github Repo in Github

- Name: azure-devops-github-acr-aks-app1
- Description: Azure DevOps App1 Demo with AKS, Github and Azure Containter Registry
- Repo Type: Public / Private (Your choice)
- Click on Create Repository

Create Local Git Repo and Check-In Code

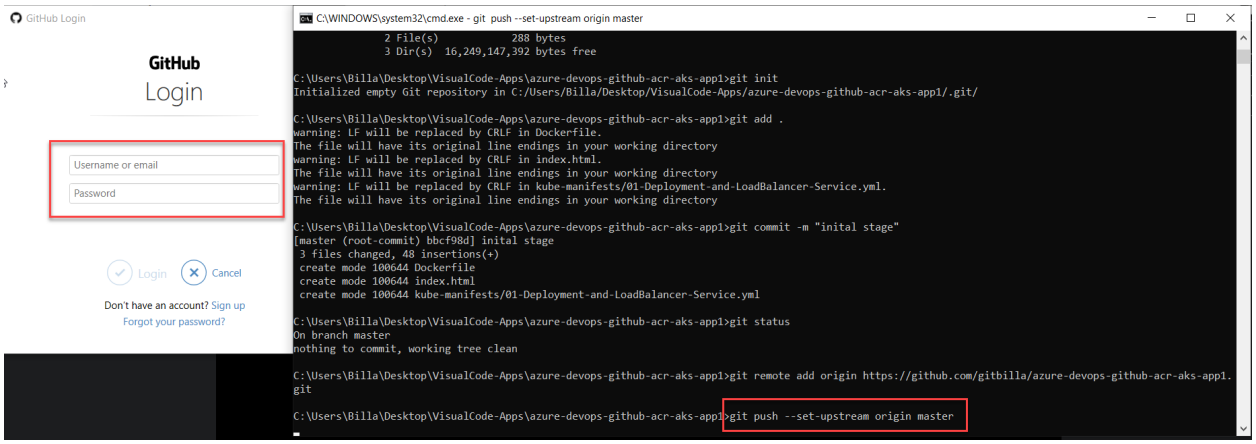
- Create Local folders

```
# Create a folder for all Repos we are going to create
mkdir azure-devops-aks-demo-repos
cd azure-devops-aks-demo-repos
```

```
# Create a Directory for Repo
mkdir azure-devops-github-acr-aks-app1
cd azure-devops-github-acr-aks-app1
```

```
C:\Users\Billa\Desktop\VisualCode-Apps\azure-devops-github-acr-aks-app1>dir
 Volume in drive C has no label.
 Volume Serial Number is 3EF1-9EAF

 Directory of C:\Users\Billa\Desktop\VisualCode-Apps\azure-devops-github-acr-aks-app1

09/03/2022  01:22    <DIR>          .
09/03/2022  01:22    <DIR>          ..
08/03/2022  18:38                48 Dockerfile
08/03/2022  18:38               240 index.html
09/03/2022  01:22    <DIR>          kube-manifests
               2 File(s)            288 bytes
               3 Dir(s)  16,249,147,392 bytes free

C:\Users\Billa\Desktop\VisualCode-Apps\azure-devops-github-acr-aks-app1>
```

```
C:\Users\Billa\Desktop\VisualCode-Apps\azure-devops-github-acr-aks-app1>git init
Initialized empty Git repository in C:/Users/Billa/Desktop/VisualCode-Apps/azure-devops-github-acr-aks-app1/.git/

C:\Users\Billa\Desktop\VisualCode-Apps\azure-devops-github-acr-aks-app1>git add .
warning: LF will be replaced by CRLF in Dockerfile.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in index.html.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in kube-manifests/01-Deployment-and-LoadBalancer-Service.yml.
The file will have its original line endings in your working directory

C:\Users\Billa\Desktop\VisualCode-Apps\azure-devops-github-acr-aks-app1>git commit -m "inital stage"
[master (root-commit) bbcf98d] inital stage
 3 files changed, 48 insertions(+)
 create mode 100644 Dockerfile
 create mode 100644 index.html
 create mode 100644 kube-manifests/01-Deployment-and-LoadBalancer-Service.yml

C:\Users\Billa\Desktop\VisualCode-Apps\azure-devops-github-acr-aks-app1>git status
On branch master
nothing to commit, working tree clean

C:\Users\Billa\Desktop\VisualCode-Apps\azure-devops-github-acr-aks-app1>
```



- Copy all files from `Giti-Repository-files` folder to our new repo folder `azure-devops-github-acr-aks-app1`

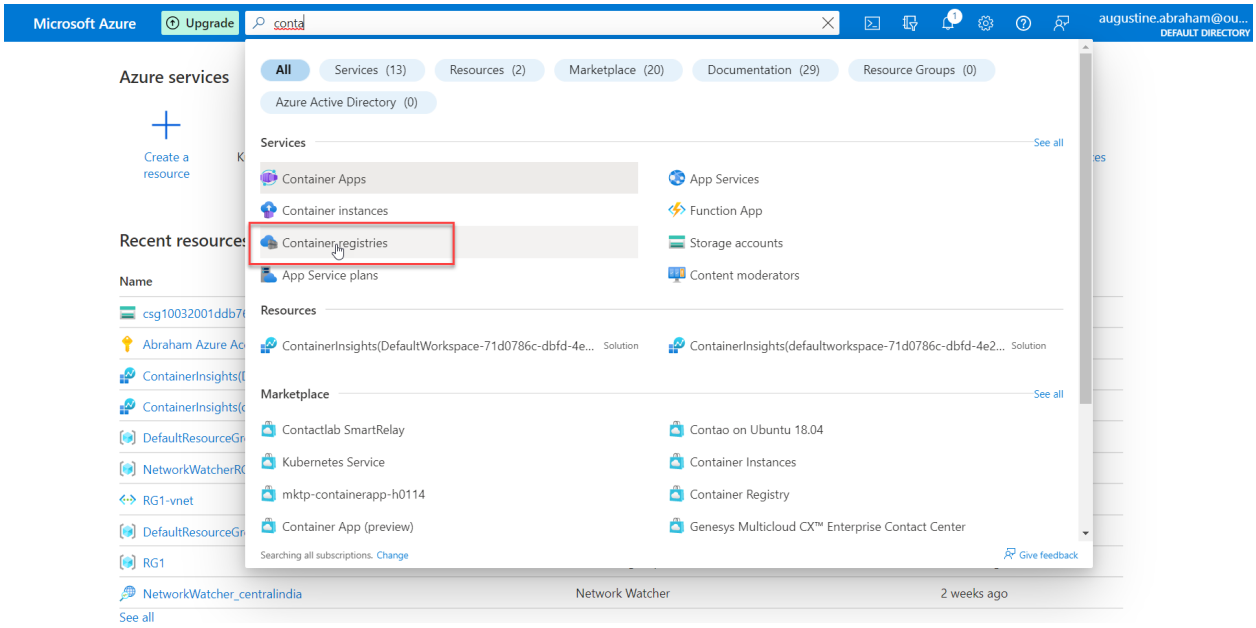
```
# Initialize Git Repo
cd azure-devops-github-acr-aks-app1
git init

# Do local Commit
echo "# Azure DevOps App1 Demo with AKS, Github and ACR" >> README.md
git add .
git commit -am "V1 Base Commit"

# Link Github Remote Repository
git remote add origin
https://github.com/gitbilla/azure-devops-github-acr-aks-app1.git

# Push to Remote Repository
git push --set-upstream origin master

# Go to Github Repo - Refresh and check files appeared in githbub repo
https://github.com/gitbilla/azure-devops-github-acr-aks-app1.git
```

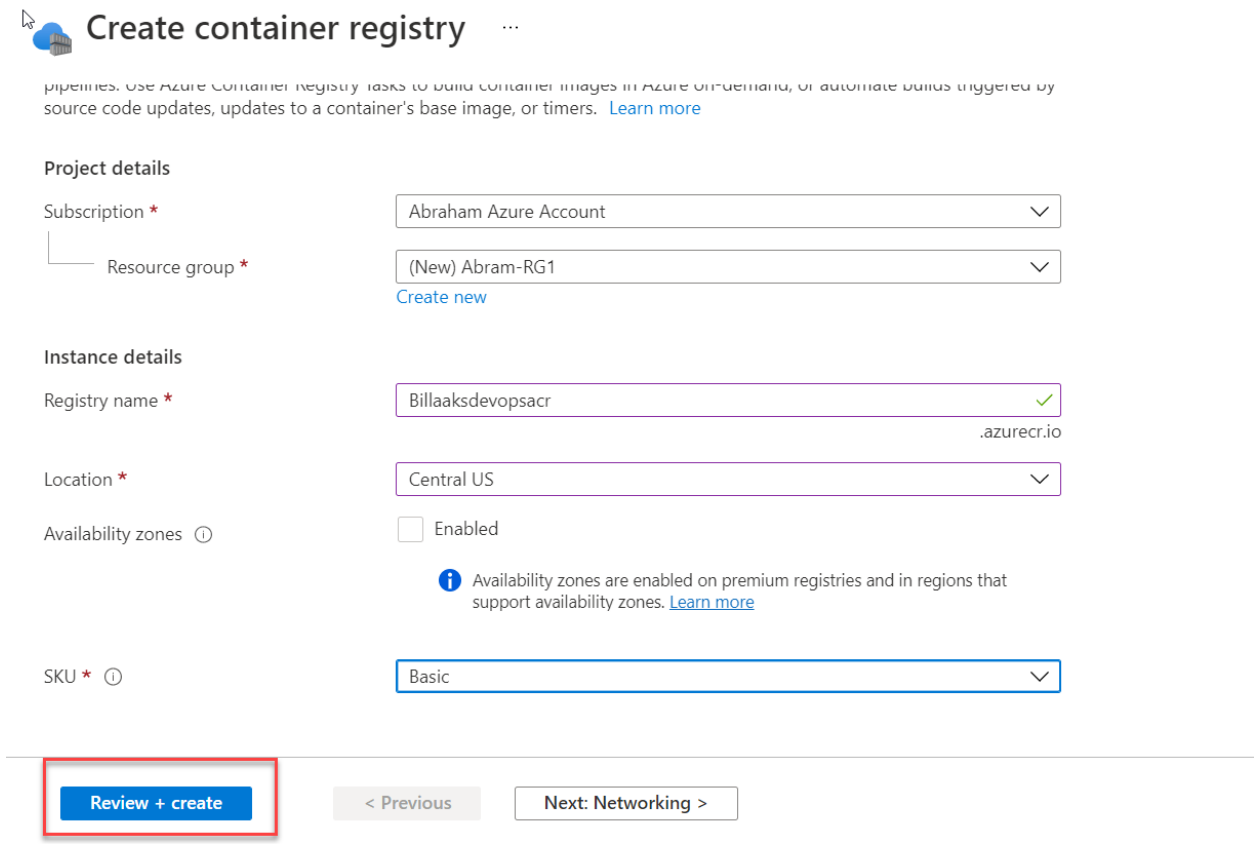
Step-03: Review github checked-in files

- kube-manifests
- Dockerfile
- index.html

Step-04: Create Azure Container Registry ACR

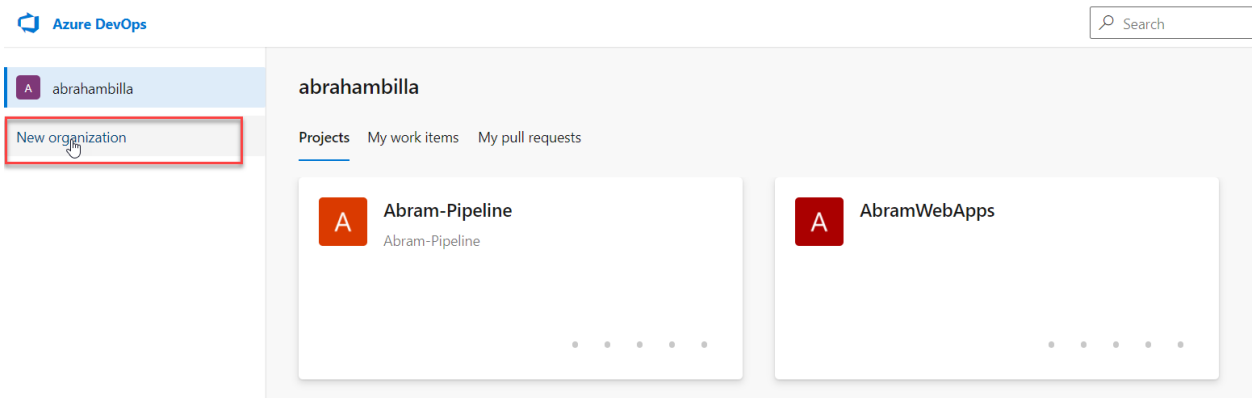
- Go to Services -> Container Registries
- Click on Add
- Subscription: Abraham Azure Account
- Resource Group: Abram-RG1
- Registry Name: `Billaaksdevopsacr`
- Location: Central US
- SKU: Basic Click on Review + Create
- Click on Create





Step-05: Creat DevOps Organization

- Go to
  - https://dev.azure.com/
  - Sign in to Azure DevOps
- Our Organization will be automatically created and if you want to manually create an organization you can create one.
- Organization Name: AbramDevops


Azure DevOps
Search
abrahambilla
New organization
abrahambilla
Projects
My work items
My pull requests
Abram-Pipeline
Abram-Pipeline
AbramWebApps

Azure DevOps

augustine.abraham@outlook.com Switch directory

Almost done...

Name your Azure DevOps organization

dev.azure.com/ AbramDevops

We'll host your projects in

Central US

I would like information, tips, and offers about Azure DevOps and other Microsoft products and services. Privacy Statement.

Enter the characters you see

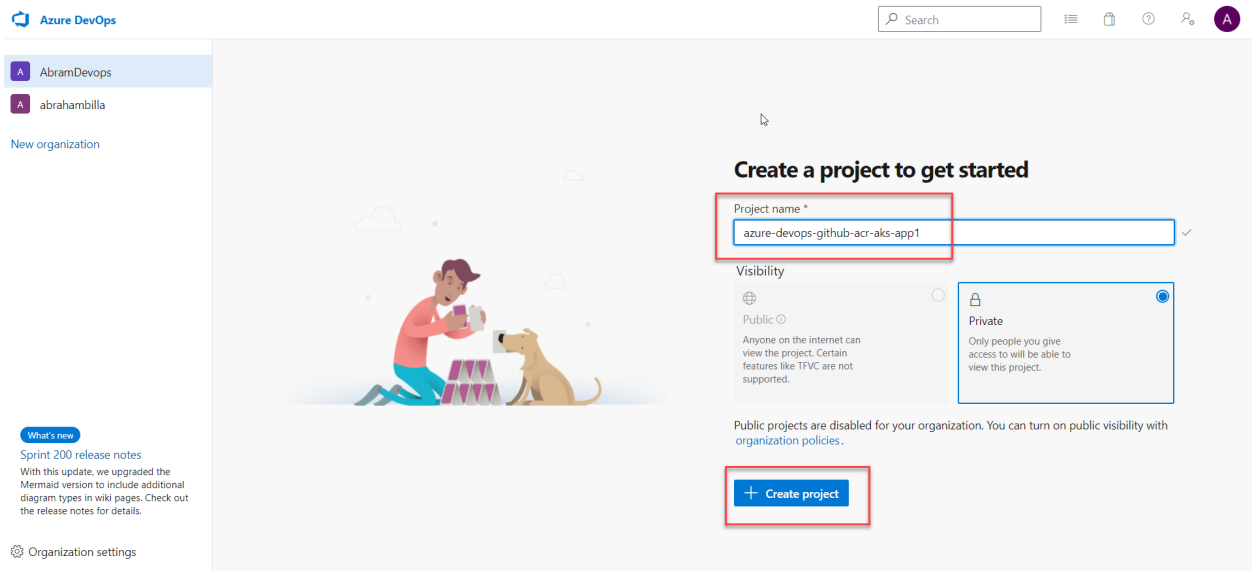
New | Audio

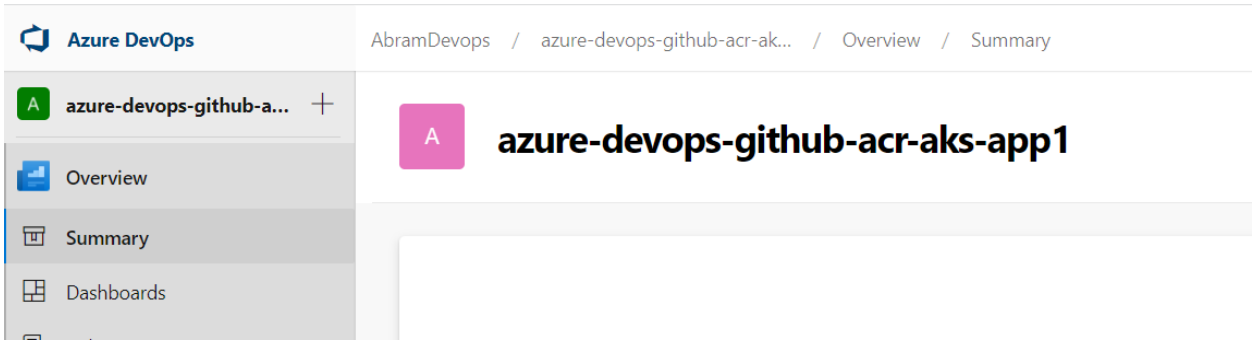
X464YXLGa

Continue

Step-06 : Create DevOps Project

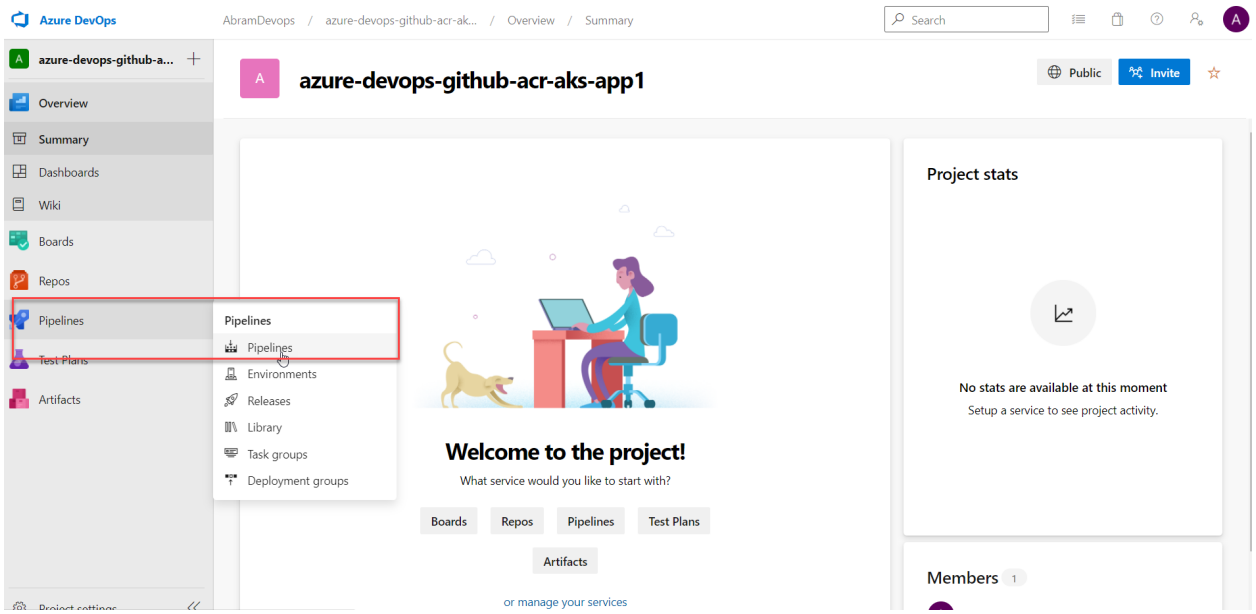
- Project Name: azure-devops-github-acr-aks-app1
- Project Description: AKS CICD Pipelines with Github and Azure Container Registry ACR
- Visibility: Private
- Advanced: Leave to defaults
    - Version Control: Git
    - Work Item Process: Basic





Step-07: Create Basic Build Pipeline

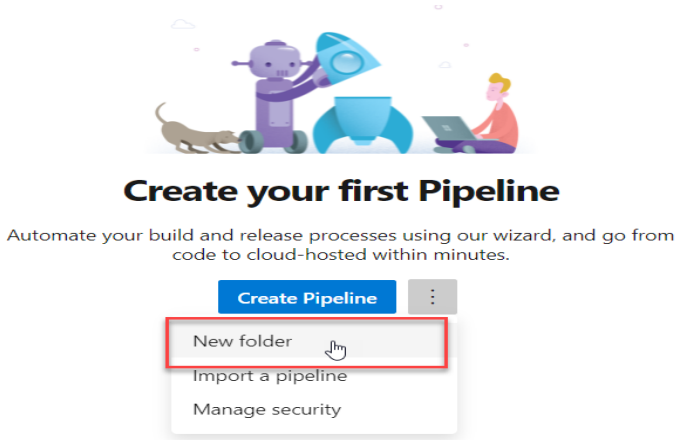
Configure the repository for the artifacts



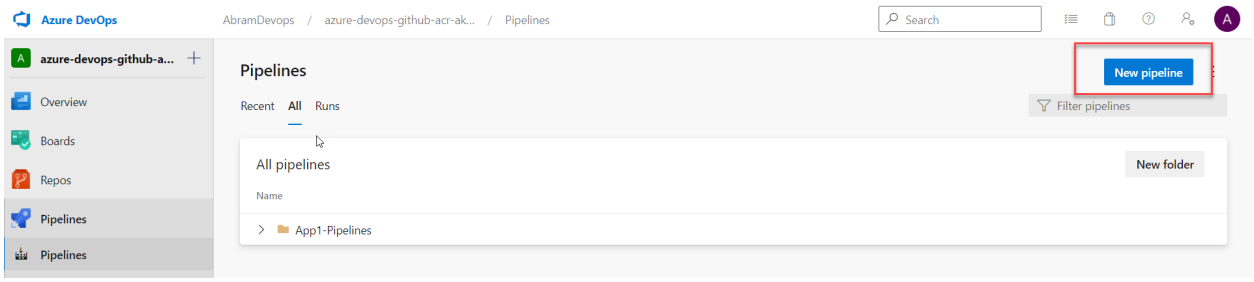
Verify the files

Create the Pipeline

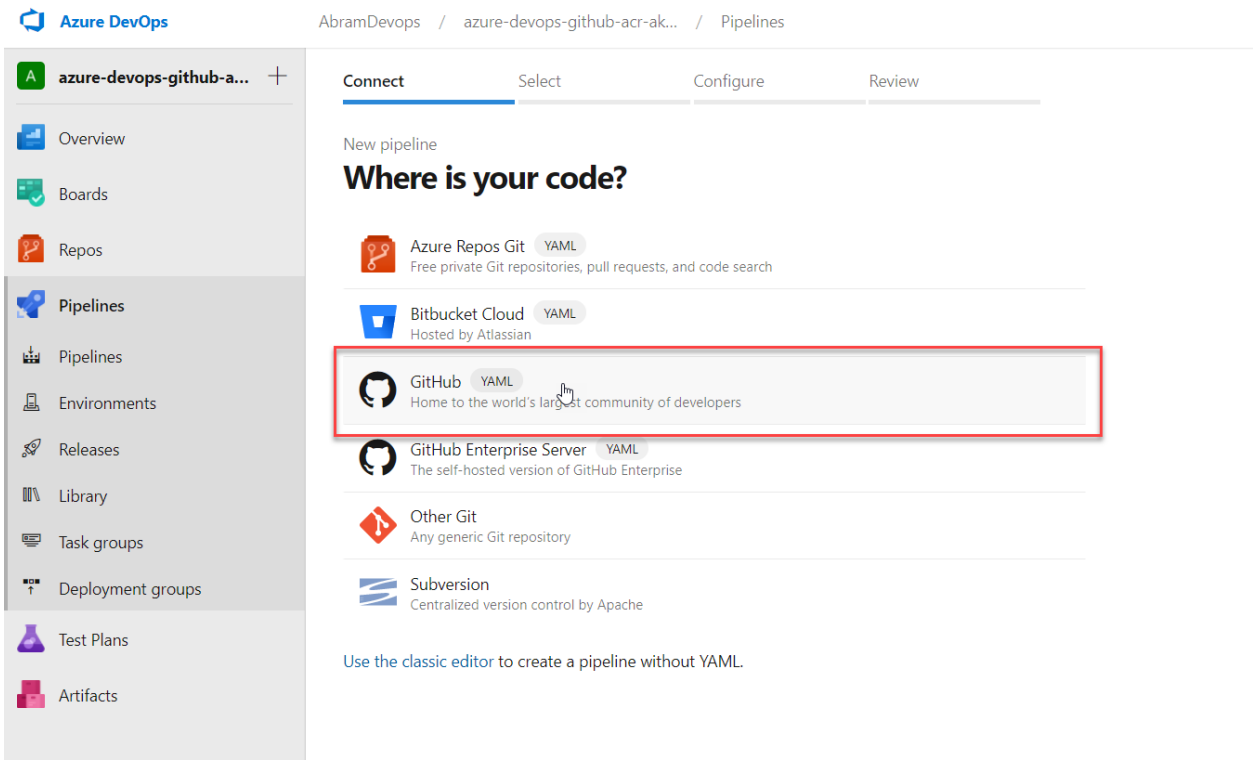
- Create Folder -> App1-Pipelines

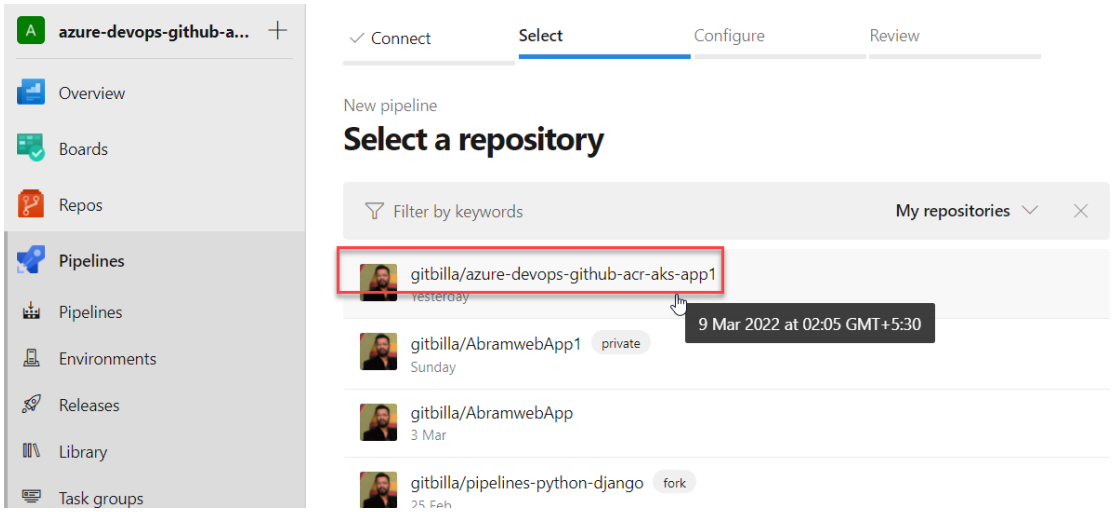


- Go to Pipelines -> Create New Pipeline

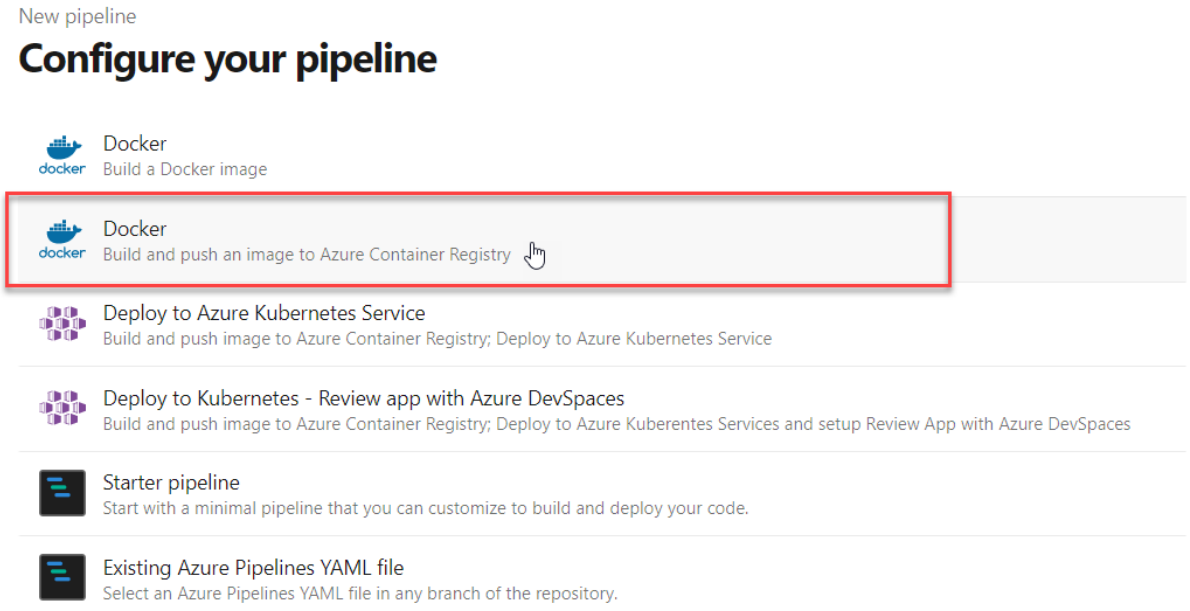


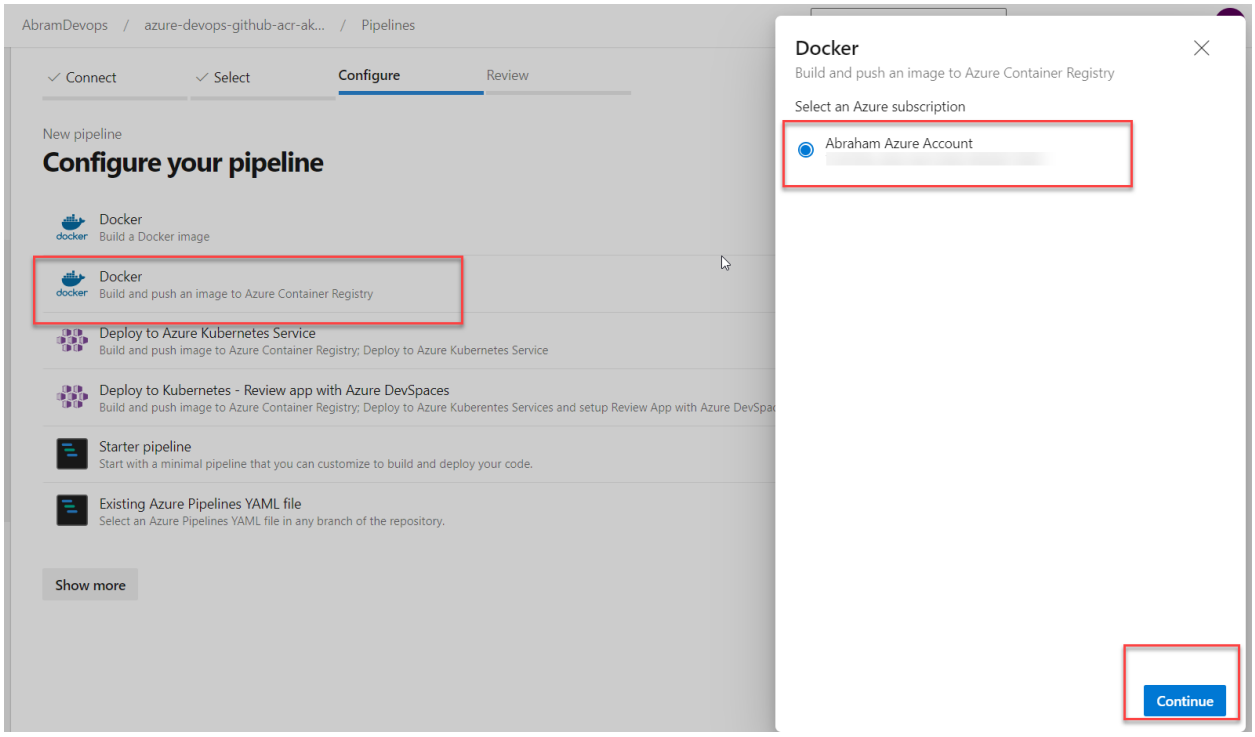
- Where is your Code?: Github

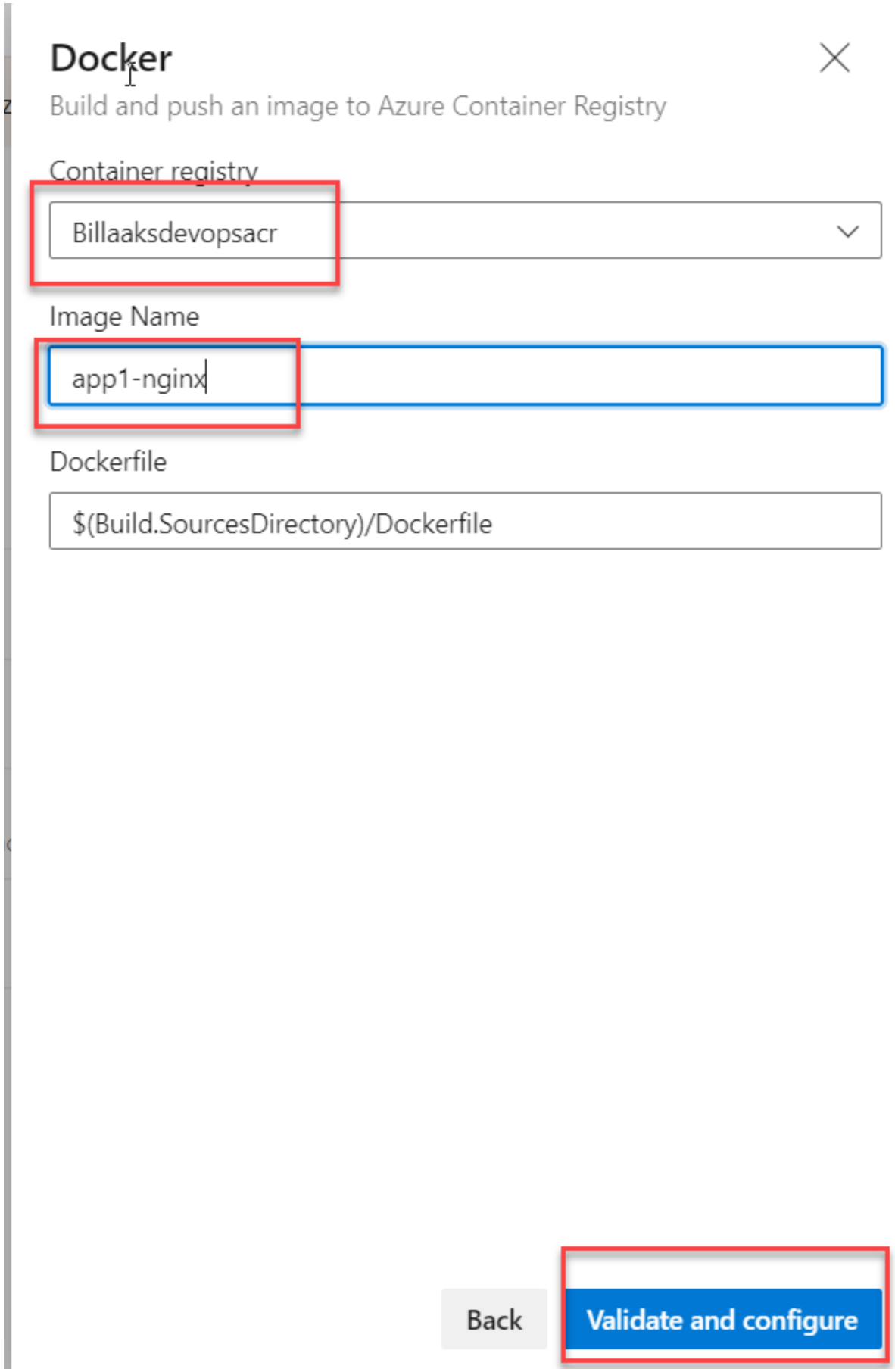




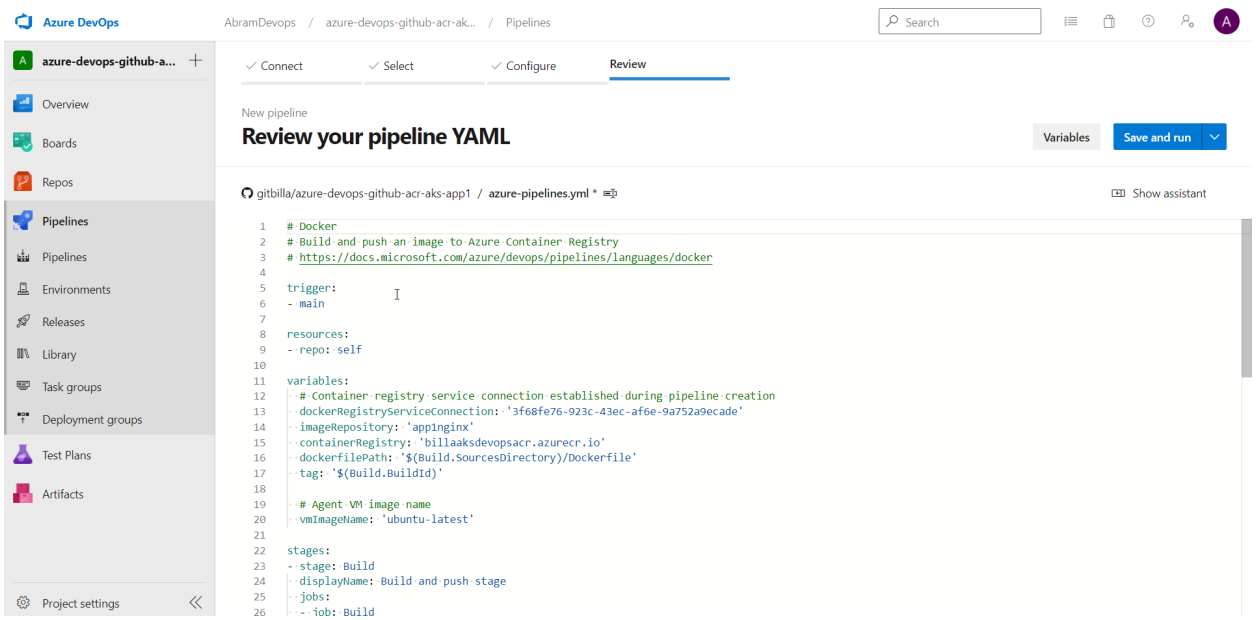
- Select Repository: azure-devops-github-acr-aks-app1
    - Provide github username and Password
    - Click on Approve and Install for Repositories selected
- Configure Your Pipeline: Docker (Build and Push Image to Azure Container Registry)



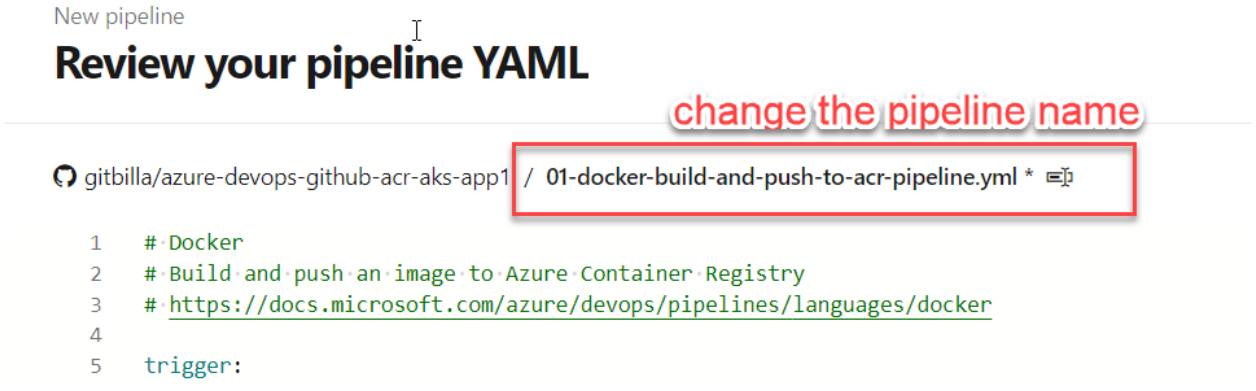

AbramDevops / azure-devops-github-acr-ak... / Pipelines
Connect
Select
Configure
Review
New pipeline
Configure your pipeline
Docker
Build a Docker image
Docker
Build and push an image to Azure Container Registry
Deploy to Azure Kubernetes Service
Build and push image to Azure Container Registry; Deploy to Azure Kubernetes Service
Deploy to Kubernetes - Review app with Azure DevSpaces
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.
Show more
Docker
Build and push an image to Azure Container Registry
Select an Azure subscription
Abraham Azure Account
Continue


Docker
Build and push an image to Azure Container Registry
Container registry
Billaaksdevopsacr
Image Name
app1-nginx
Dockerfile
$(Build.SourcesDirectory)/Dockerfile
Back
Validate and configure

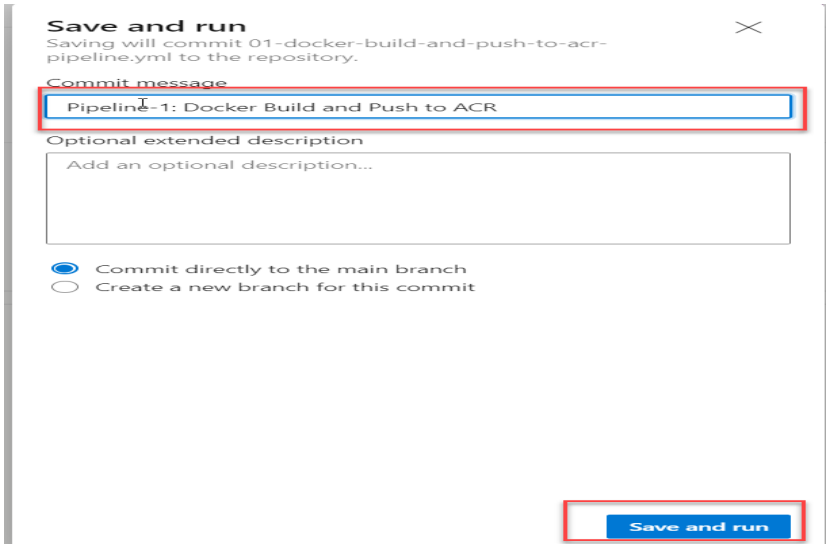
- Select an Azure Subscription: Abraham Azure Account
- Continue (Login as admin user)
- Container Registry: Billaaksdevopsacr
- Image Name: app1-nginx
- Dockerfile: $(Build.SourcesDirectory)/Dockerfile
- Click on Validate and Configure



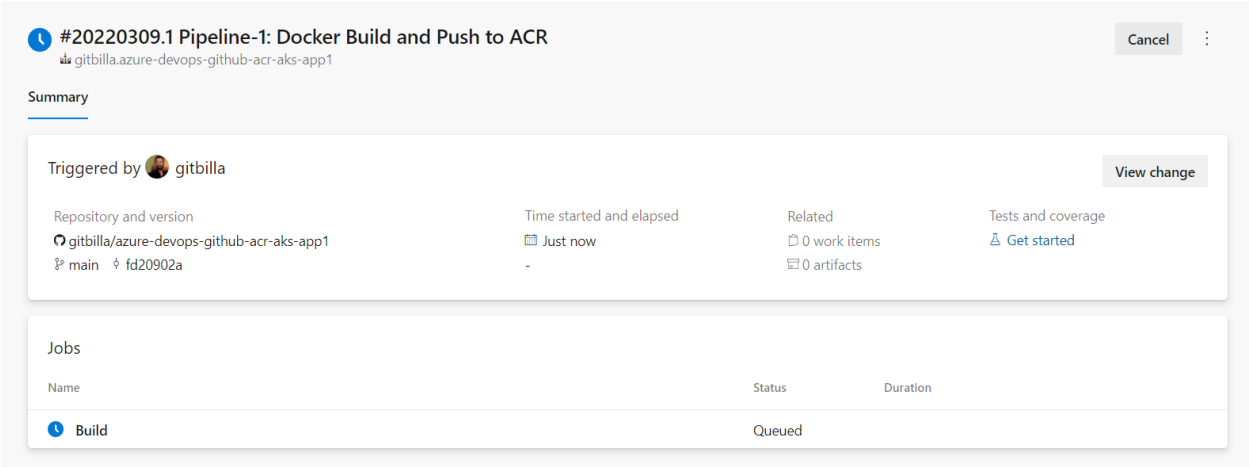
- Change Pipeline Name: 01-docker-build-and-push-to-acr-pipeline.yml
- Click on Save and Run



- Commit Message: Pipeline-1: Docker Build and Push to ACR

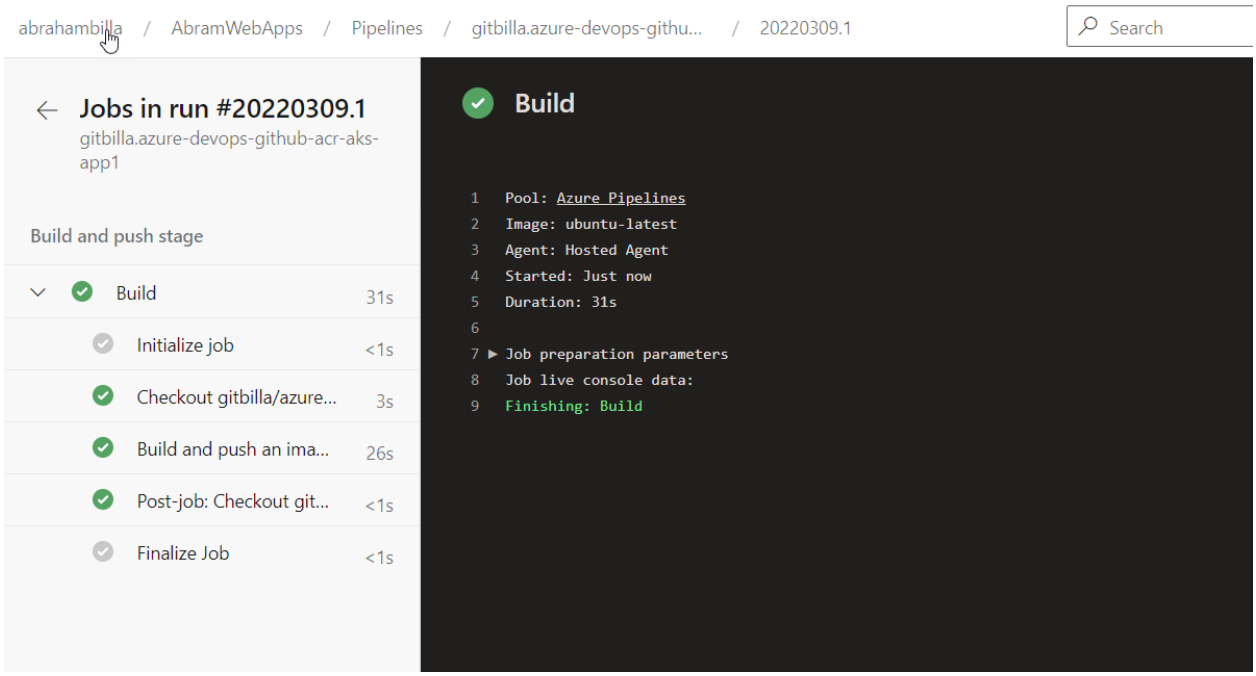


- Commit directly to master branch: check
- Click on Save and Run

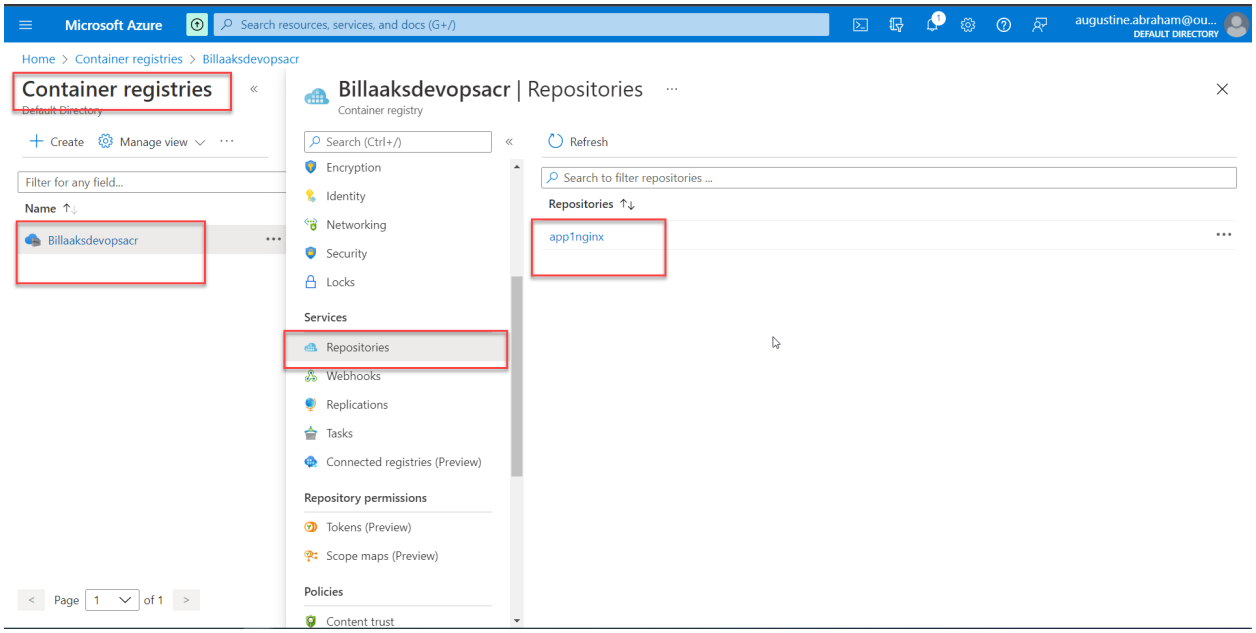


Step-08: Review Build Logs & Docker Image in ACR

- Review Build logs
- Review Image in ACR

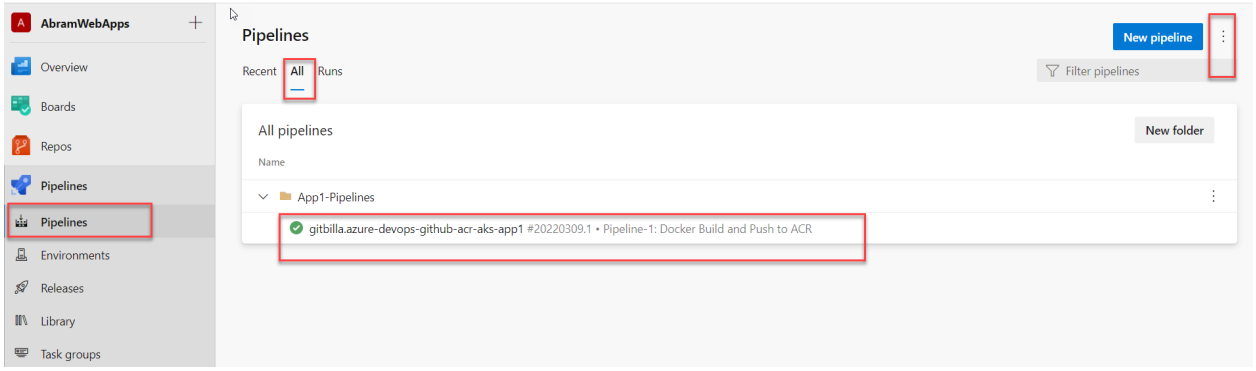


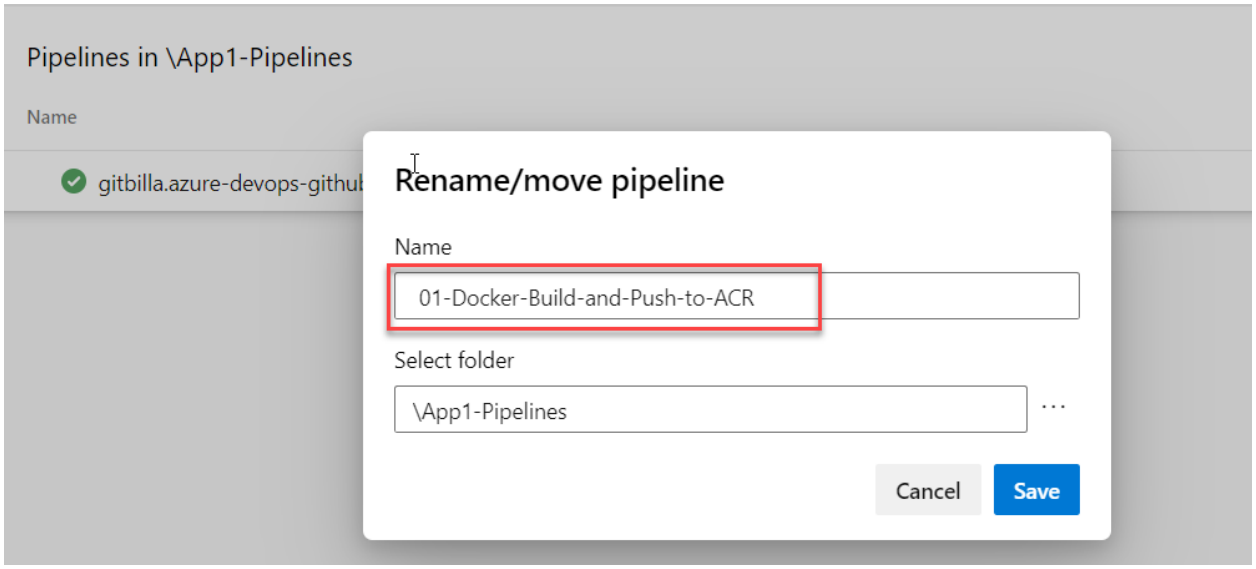
Verify from azure portal

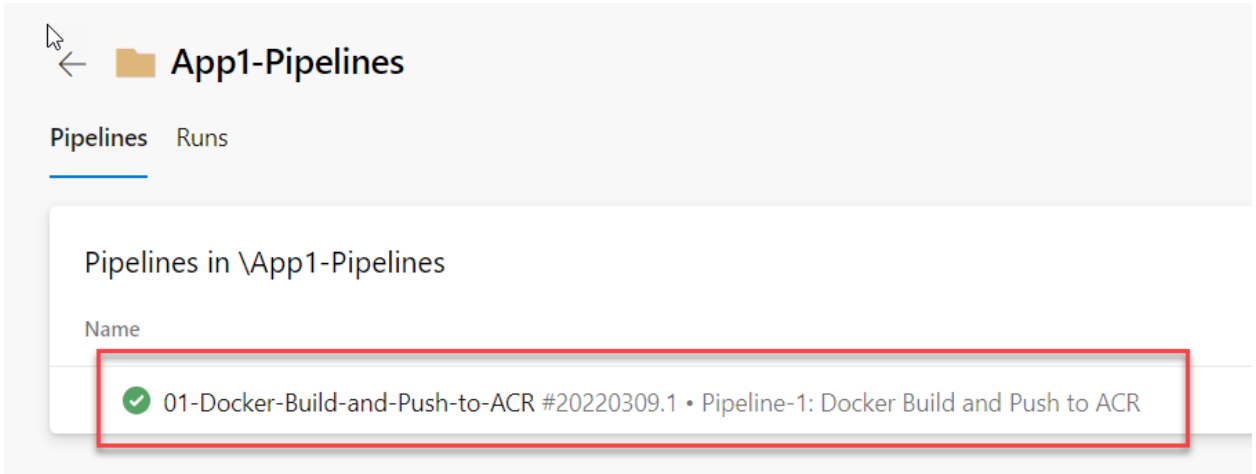


Step-09: Rename Pipeline Name

- Click on Pipeline -> Rename/Move
- Name: 01-Docker-Build-and-Push-to-ACR







Step-10: Make changes to index.html and push changes to git repo - V2 Commit

```
# Pull changes related to pipeline to local repo
git pull
ls -lrt
```

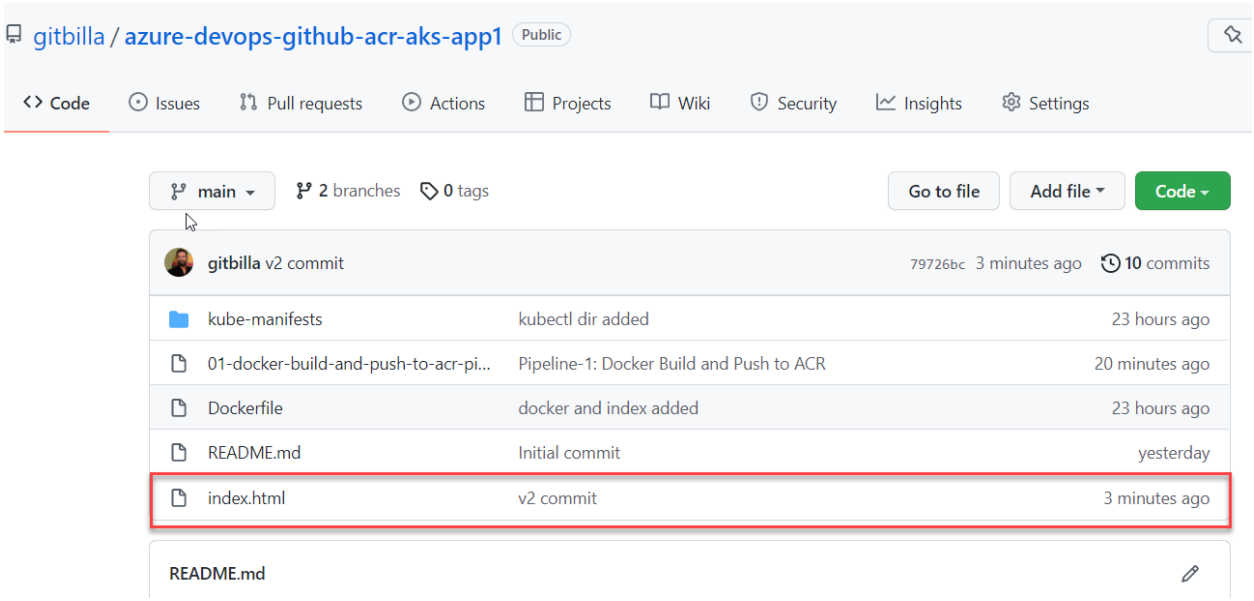
```
Billa@DESKTOP-2SFP2EH MINGW64 ~/Desktop/VisualCode-Apps/azure-devops-github-acr-aks-app1 (main)
$ git pull
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 7 (delta 4), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (7/7), 1.61 KiB | 4.00 KiB/s, done.
From https://github.com/gitbilla/azure-devops-github-acr-aks-app1
   fafeec4..82f835f  main       -> origin/main
Updating fafeec4..82f835f
Fast-forward
 01-docker-build-and-push-to-acr-pipeline.yml | 39 ++++++++++++++++++++++++++++++++++++++
 1 file changed, 39 insertions(+)
 create mode 100644 01-docker-build-and-push-to-acr-pipeline.yml

Billa@DESKTOP-2SFP2EH MINGW64 ~/Desktop/VisualCode-Apps/azure-devops-github-acr-aks-app1 (main)
$ ls -la
total 19
drwxr-xr-x 1 Billa 197121    0 Mar 10 00:54 ./
drwxr-xr-x 1 Billa 197121    0 Mar  9 19:09 ../
drwxr-xr-x 1 Billa 197121    0 Mar 10 00:54 .git/
-rw-r--r-- 1 Billa 197121 1059 Mar 10 00:54 01-docker-build-and-push-to-acr-pipeline.yml
-rw-r--r-- 1 Billa 197121   49 Mar  9 19:09 Dockerfile
-rw-r--r-- 1 Billa 197121  248 Mar  9 19:09 index.html
drwxr-xr-x 1 Billa 197121    0 Mar  9 19:09 kube-manifests/
-rw-r--r-- 1 Billa 197121  108 Mar  9 19:09 README.md

Billa@DESKTOP-2SFP2EH MINGW64 ~/Desktop/VisualCode-Apps/azure-devops-github-acr-aks-app1 (main)
$
```

```
# Make changes to index.html
index.html file - change version v1
```

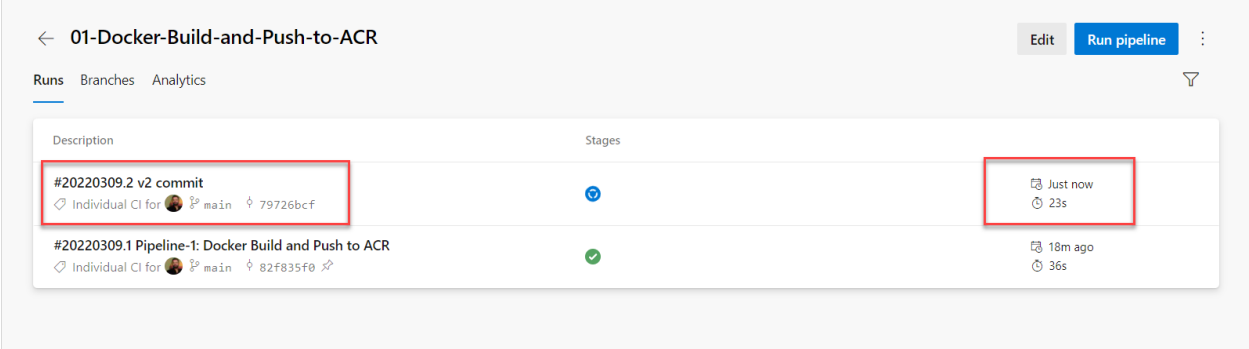
```
$ cat index.html
<!DOCTYPE html>
<html>
    <body style="background-color:rgb(212, 240, 234);">
        <h1>Abraham Welcome to - Azure DevOps App1 - V1</h1>
        <h2>Azure DevOps Demo App1</h2>
        <h2>Application Version: V1</h2>
    </body>
</html>
```

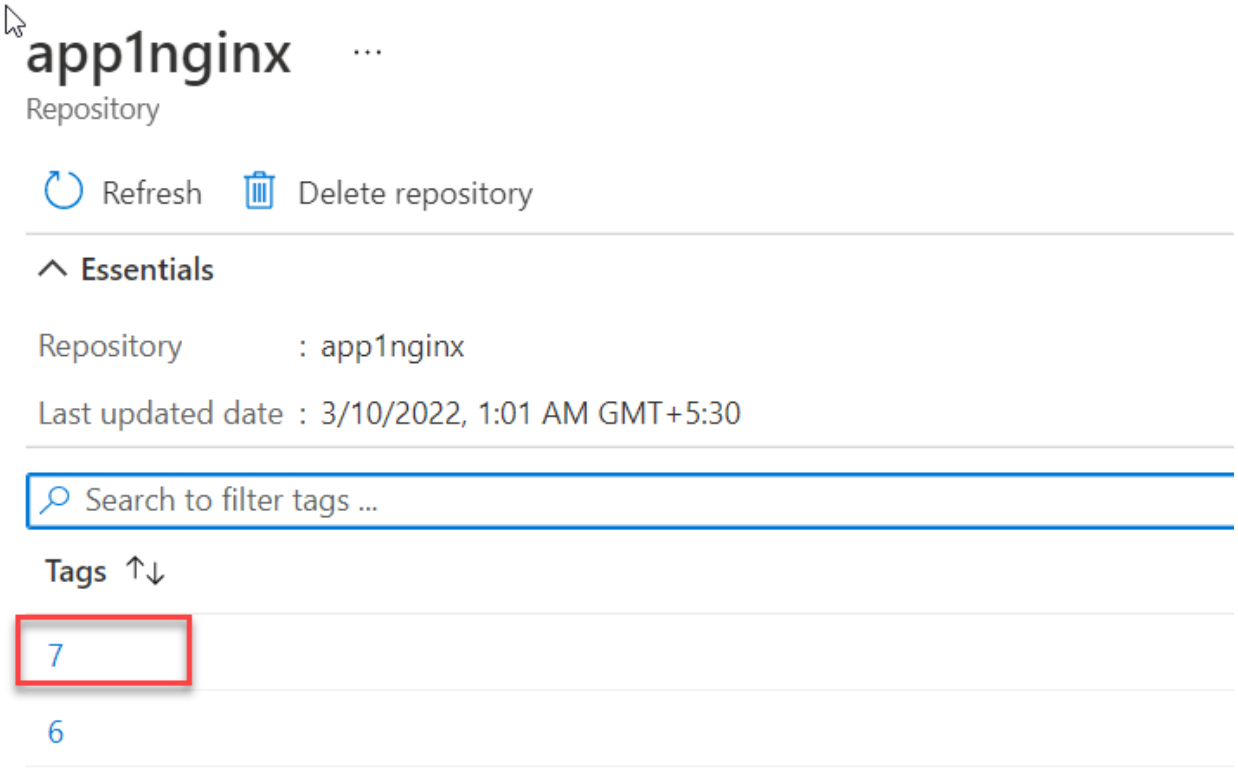
```
# Push changes
git add .
git commit -am "V2 Commit for index.html"
git push
```

```
Billa@DESKTOP-2SFP2EH MINGW64 ~/Desktop/VisualCode-Apps/azure-devops-github-acr-aks-app1 (main)
$ git add .

Billa@DESKTOP-2SFP2EH MINGW64 ~/Desktop/VisualCode-Apps/azure-devops-github-acr-aks-app1 (main)
$ git commit -m "v2 commit"
[main 79726bc] v2 commit
 1 file changed, 1 insertion(+), 1 deletion(-)

Billa@DESKTOP-2SFP2EH MINGW64 ~/Desktop/VisualCode-Apps/azure-devops-github-acr-aks-app1 (main)
$ pwd
/c/Users/Billa/Desktop/VisualCode-Apps/azure-devops-github-acr-aks-app1

Billa@DESKTOP-2SFP2EH MINGW64 ~/Desktop/VisualCode-Apps/azure-devops-github-acr-aks-app1 (main)
$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 307 bytes | 307.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/gitbilla/azure-devops-github-acr-aks-app1.git
   82f835f..79726bc  main -> main
```



The pipeline will automatically trigger



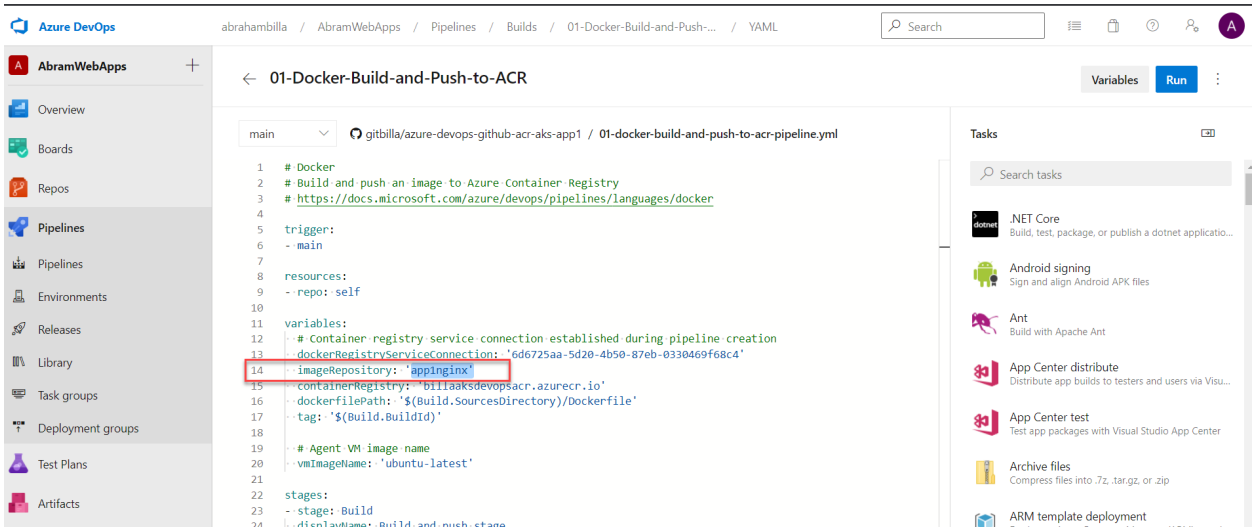


- Verify Build logs
- Verify ACR Image

Step-12: Add Namespace for Docker Images stored in ACR

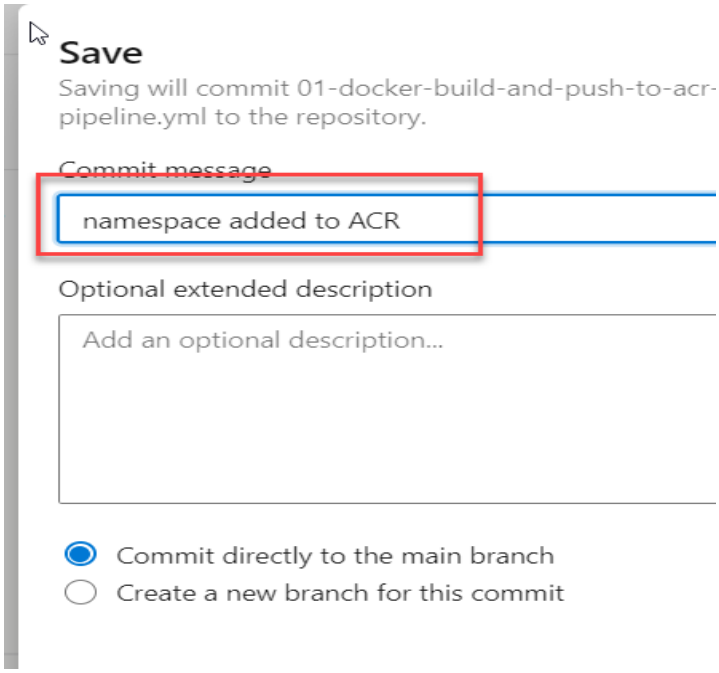
- Go to Pipeline -> 01-Docker-Build-and-Push-to-ACR -> Edit
- Update the below and Save

```
# Before
  imageRepository: 'app1nginx'
```

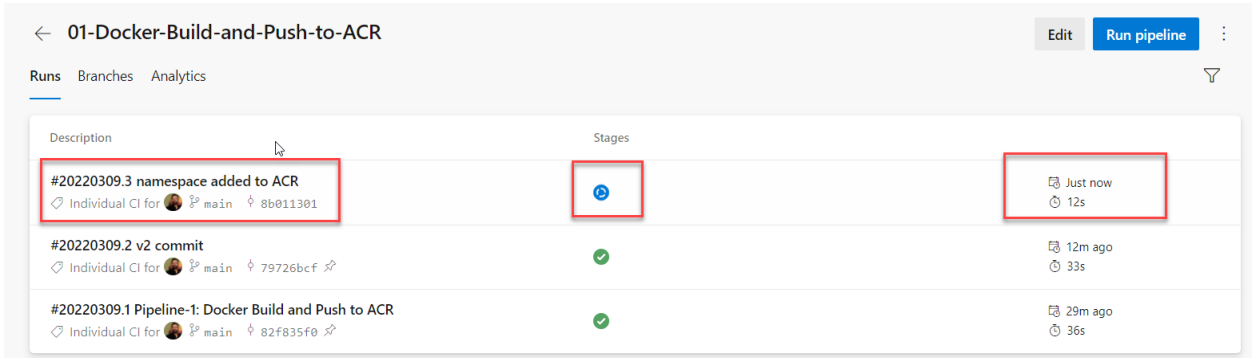



```
# After
  imageRepository: 'app1/app1nginx'
```

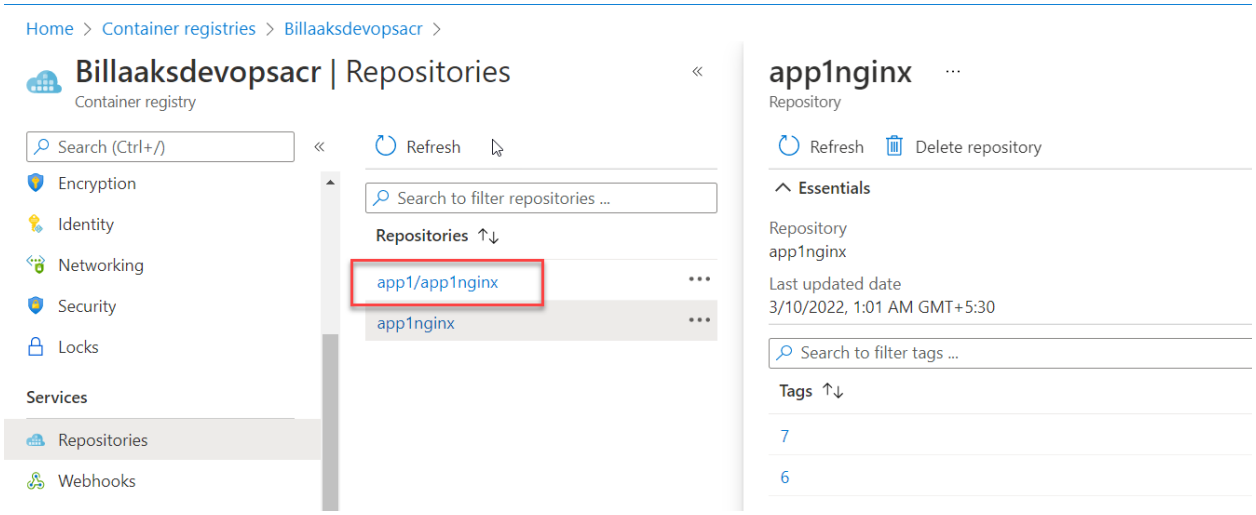
```
11  variables:
12    # Container registry service connection established during pipeline creation
13    dockerRegistryServiceConnection: '6d6725aa-5d20-4b50-87eb-0330469f68c4'
14    imageRepository: 'app1/app1nginx'
15    containerRegistry: 'billaaksdevopsacr.azurecr.io'
16    dockerfilePath: '$(Build.SourcesDirectory)/Dockerfile'
17    tag: '$(Build.BuildId)'
18
```



Once the name is changed the pipeline gets triggered



- Verify Build logs
- Verify ACR Image



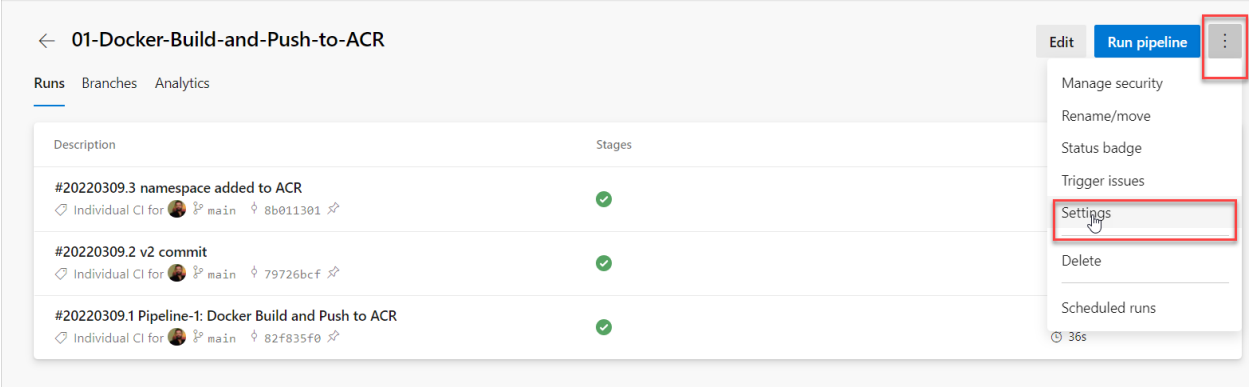
Step-13: Make changes to index.html and push changes to git repo - V3 Commit

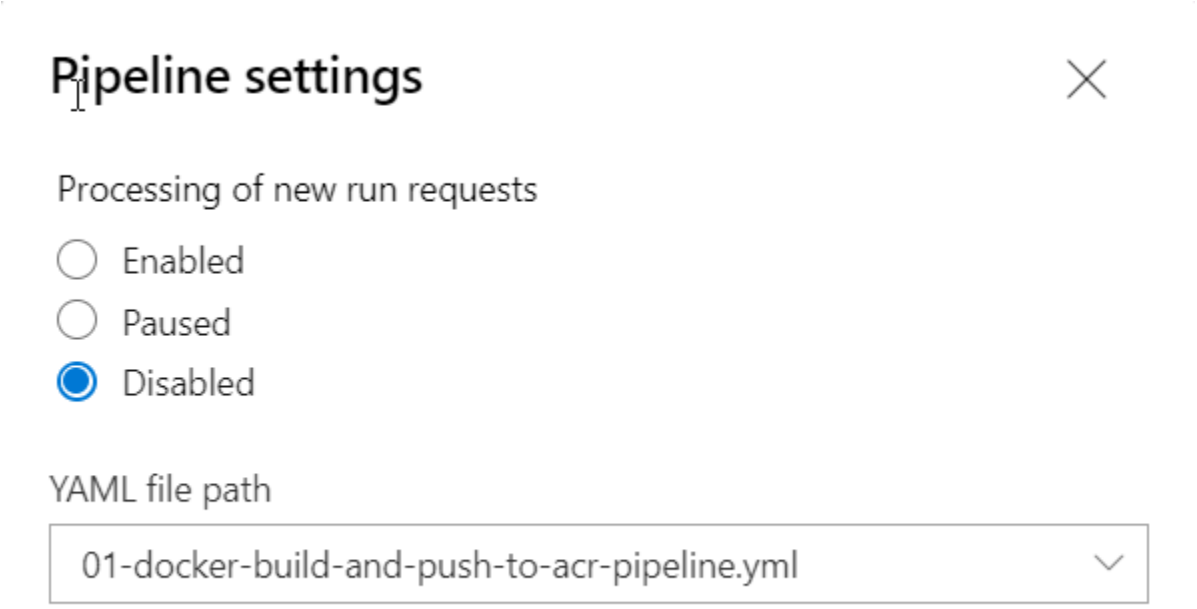
```
# Pull changes related to pipeline to local repo
git pull
ls -lrt

# Make changes to index.html
index.html file - change version v3

# Push changes
git add .
git commit -am "V3 Commit for index.html"
git push
```

- Verify Build logs
- Verify ACR Image

Step-14: Disable Pipeline

- Go to Pipeline -> 01-Docker-Build-and-Push-to-ACR -> Settings -> Disable





Step-15: Review Pipeline code

- Click on Pipeline -> Edit Pipeline
- Review pipeline code
- Review Service Connections

```
# Docker
# Build and push an image to Azure Container Registry
# https://docs.microsoft.com/azure/devops/pipelines/languages/docker

trigger:
- master
```

```
resources:
- repo: self

variables:
  # Container registry service connection established during pipeline creation
  dockerRegistryServiceConnection: '6a8843fd-7313-48e2-9381-3f9ef59ce82d' ##
Review Service Connections
  imageRepository: 'app1/app1nginx'
  containerRegistry: 'aksdevopsacr.azurecr.io'
  dockerfilePath: '$(Build.SourcesDirectory)/Dockerfile'
  tag: '$(Build.BuildId)'

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

stages:
- stage: Build
  displayName: Build and push 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)
```

AZURE Storage

1. Create a AKS

az group create --name abram-rg --location southindia

abraham@Azure:~$ az aks create --resource-group abram-rg --name abramAKS --location southindia --kubernetes-version 1.20.15 --node-count 1 --network-plugin azure --disable-rbac --generate-ssh-keys

```
abraham@Azure:~$ kubectl cluster-info

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
error: You must be logged in to the server (the server has asked for the client to provide credentials)
abraham@Azure:~$ az group create --name abram-rg --location southindia
{
  "id": "/subscriptions/71d0786c-dbfd-4e2f-9d48-49838b718991/resourceGroups/abram-rg",
  "location": "southindia",
  "managedBy": null,
  "name": "abram-rg",
  "properties": {
    "provisioningState": "Succeeded"
  },
  "tags": null,
  "type": "Microsoft.Resources/resourceGroups"
}
abraham@Azure:~$ az provider show -n Microsoft.OperationsManagement -o table
Namespace                      RegistrationPolicy    RegistrationState
-----------------------------  --------------------  -------------------
Microsoft.OperationsManagement  RegistrationRequired  Registered
abraham@Azure:~$  az provider show -n Microsoft.OperationalInsights -o table
Namespace                     RegistrationPolicy    RegistrationState
----------------------------  --------------------  -------------------
Microsoft.OperationalInsights  RegistrationRequired  Registered
abraham@Azure:~$ az aks create --resource-group abram-rg --name abramAKS --location southindia --kubernetes-version 1.20.15 --node-count 1 --network-plugin a
zure --disable-rbac --generate-ssh-keys
SSH key files '/home/abraham/.ssh/id_rsa' and '/home/abraham/.ssh/id_rsa.pub' have been generated under ~/.ssh to allow SSH access to the VM. If using machin
es without permanent storage like Azure Cloud Shell without an attached file share, back up your keys to a safe location
 \ Running
```

2. #az aks get-credentials --resource-group abram-rg --name abramAKS

```
abraham@Azure:~$ az aks get-credentials --resource-group abram-rg --name abramAKS
Merged "abramAKS" as current context in /home/abraham/.kube/config
abraham@Azure:~$
```