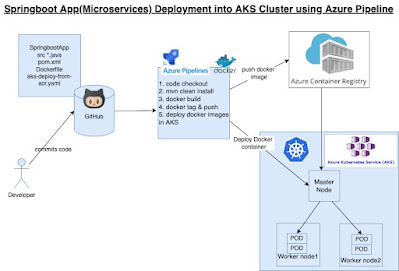
**How to Deploy Springboot App into AKS cluster using Azure Pipelines | Deploy Docker Containers into AKS cluster using Azure Rlease Pipelines | Deploy Microservices into AKS cluster using Azure Pipelines**

We are going to learn how to deploy Springboot Microservices Docker containers into Azure Kubernetes Cluster(AKS) using Azure pipelines.

**Sample springboot App Code:**

I have created a sample Springboot App setup in GitHub. to access code base in GitHub.



**Follow below the steps:**

**Pre-requisites:**

1. AKS cluster needs to be up running. You can create AKS cluster using any of one of the below options:

•Create AKS cluster in Azure portal directly

•Create AKS cluster using Azure CLI

You need to create a resource group first.

az group create --name myResourceGroup --location southcentralus

**Create AKS cluster with 2 worker nodes**

az aks create --resource-group myResourceGroup --name myAKSCluster **--node-count 2** --enable-addons monitoring --generate-ssh-keys

az aks show --name myAKSCluster --resource-group myResourceGroup

The above command should display Cluster exists in Azure portal

**Create Azure Container Registry**

Run the below command to create your own private container registry using Azure Container Registry (ACR).

az acr create --resource-group myResourceGroup --name myacrrepo31 --sku Standard --location southcentralus

**Connect to the cluster**

 az aks get-credentials --resource-group myResourceGroup --name myAKSCluster --overwrite-existing

To verify the connection to your cluster, use the [kubectl get](https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#get) command to return a list of the cluster nodes.

kubectl get nodes

•Create AKS cluster using Terraform

2. ACR is also setup in Azure cloud.

3.  Have Azure DevOps project dashboard in

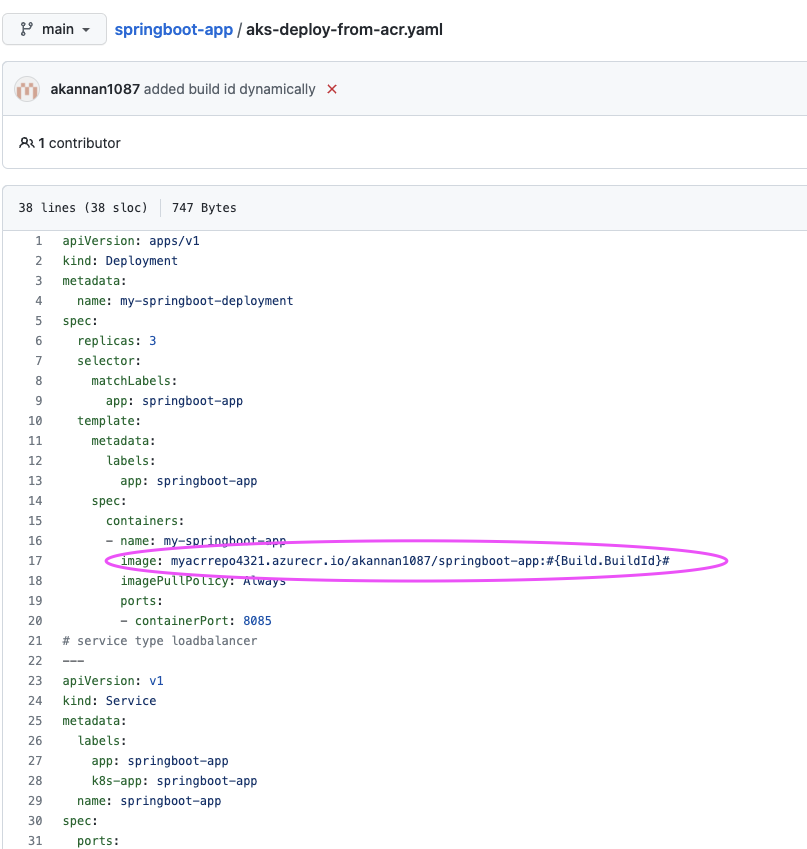
<https://dev.azure.com/>

4. Dockerfile created along with the application source code for springboot App.

5. Make sure AKS has pull access from ACR

 az aks get-credentials --resource-group myResourceGroup --name myAKSCluster --overwrite-existing

6. Modify K8S manifest file per acr, image name for AKS Deployment



**Implementation Steps:**

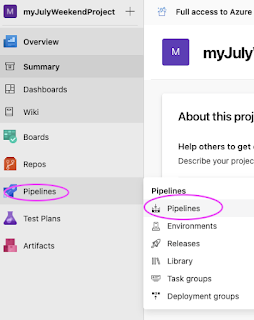
**Step 1**- Create Azure Build pipeline for building Docker images and uploading into ACR

**Step 2** - Create Azure Release pipeline for deploying Springboot Docker containers into AKS

**Step 1 - How to create a Azure Build Pipeline**

1. Login into your Azure DevOps dashboard

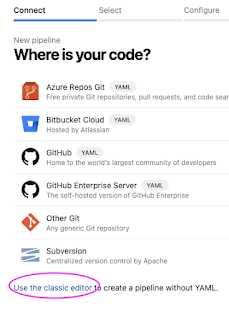
2. Click on Pipelines.

[](https://1.bp.blogspot.com/-JuBRkG2GXZw/YUN5x6idgHI/AAAAAAAADr8/d6fQG0gIXjsuCSlmAZvOVq2cnacaHBLAwCLcBGAsYHQ/s565/Screen+Shot+2021-09-16+at+12.05.02+PM.png)

3. Click on New Pipeline

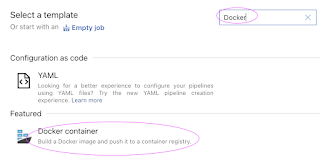
[https://1.bp.blogspot.com/-8lbBVkQk7LQ/YUN6N-vjJDI/AAAAAAAADsE/zqfb8m-JlCMJDYzgBQTN3L7Jsah9VcVmQCLcBGAsYHQ/s320/Screen%2BShot%2B2021-09-16%2Bat%2B12.07.47%2BPM.png](https://1.bp.blogspot.com/-8lbBVkQk7LQ/YUN6N-vjJDI/AAAAAAAADsE/zqfb8m-JlCMJDYzgBQTN3L7Jsah9VcVmQCLcBGAsYHQ/s1609/Screen+Shot+2021-09-16+at+12.07.47+PM.png)

4. Click on use the classic editor

[](https://1.bp.blogspot.com/-fE1HvBb7I8Q/YUN6N7DHihI/AAAAAAAADsI/lvZpDSMLvpkEoNS5dwyAMxd3YJwQoOrNQCLcBGAsYHQ/s566/Screen+Shot+2021-09-16+at+12.07.56+PM.png)

Enter your repo name and branch name where you have stored your source code along with Dockerfile:

Click on Continue. Now choose the template by typing Docker, Select Docker container and Apply.

[](https://1.bp.blogspot.com/-f8OzEO232lc/YVWdxQAu14I/AAAAAAAADso/IE2K4hplN7U1pJ-6ud0MdowXN29KBDekwCLcBGAsYHQ/s1318/Screen+Shot+2021-09-30+at+4.50.38+PM.png)

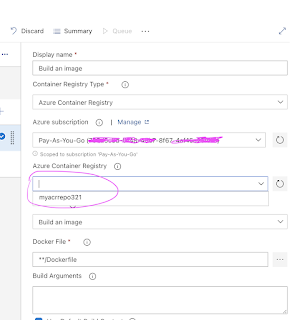
Now pipeline is created with two tasks already. We need to more tasks:

Let's add Maven build task for building the JAR file.

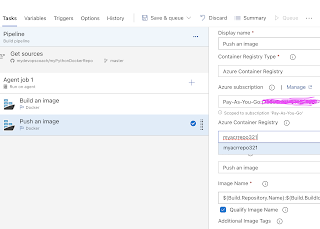
Click on + icon and type Maven

And then enter maven goal as package

Let's modify Build an image task.

[](https://1.bp.blogspot.com/-NjobUV1kKEw/YVWnOFsuT-I/AAAAAAAADsw/9EnOD-D9JCAQrdcSkEOShCu0b6KdBwWJgCLcBGAsYHQ/s1386/Screen+Shot+2021-09-30+at+5.11.27+PM.png)

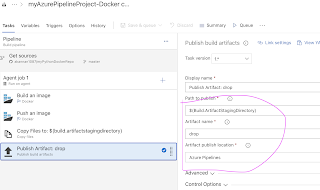
Select Push an image task

[](https://1.bp.blogspot.com/-4AfR6Y3cElM/YVWnoB8KM0I/AAAAAAAADs8/DVUmY67YIfM2C9iTV1PNiBqMSZAjYjwTACLcBGAsYHQ/s1722/Screen+Shot+2021-09-30+at+5.14.27+PM.png)

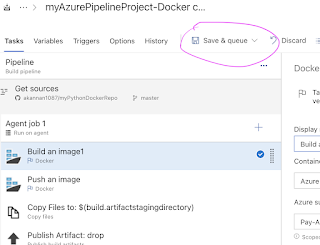
Add a task for Copying YAML file, enter the Kubernetes deployment YAML file -

**aks-deploy-from-acr.yaml**

Add Publish artifact task

[](https://1.bp.blogspot.com/-pyl4UgTG8r0/YXrVx7KBDBI/AAAAAAAADuI/7eyhuLLvzrgYiE2cXJ-0X8d_NEBcaCKhACLcBGAsYHQ/s2032/Screen+Shot+2021-10-28+at+11.53.38+AM.png)

Now click Save + Queue and run to start Building the pipeline

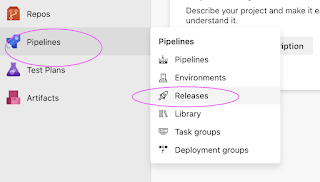
[](https://1.bp.blogspot.com/-E9GuDI5qGHQ/YXrWdLkE_UI/AAAAAAAADuQ/pU8rM5Zxc2E-I5EqaGrUATWAdklbPTqHgCLcBGAsYHQ/s1282/Screen+Shot+2021-10-28+at+11.56.25+AM.png)

Once the build is completed, you should be able to see the Docker images under

**Services --> Repositories**

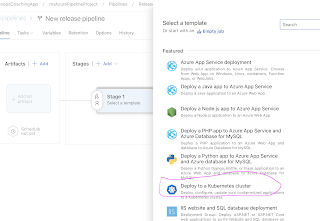
**Step 2 - How to Create Release pipeline for deploying Docker containers into AKS Cluster**

**Go to Pipelines --> Click on Releases --> New Release pipeline**

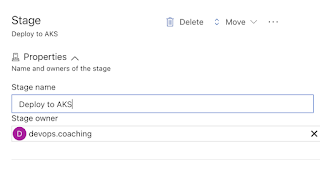
**[](https://1.bp.blogspot.com/-7QEy-4E2gC4/YXrqk6_g3yI/AAAAAAAADuY/9F9lb3DZsDU6VnhhhSRWoZ-fuM-fiPKSgCLcBGAsYHQ/s1098/Screen+Shot+2021-10-28+at+1.19.06+PM.png)**

**Click on Stage 1 and choose a template by selecting**

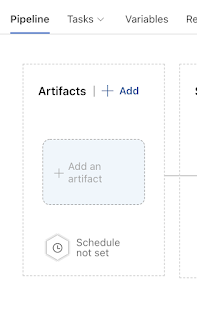
**Deploy to a Kubernetes cluster and click on Apply**

**[](https://1.bp.blogspot.com/-BYzxqp-OM_c/YXrrP9fKDrI/AAAAAAAADus/ureRO0979MYKAOW_C6_1Jt6BSp1UDwATACLcBGAsYHQ/s2026/Screen+Shot+2021-10-28+at+1.21.06+PM.png)**

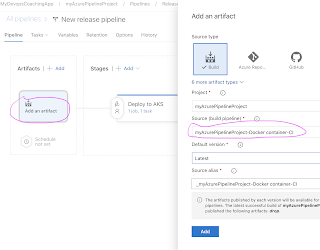
**Change the stage name to Deploy to AKS**

**[](https://1.bp.blogspot.com/-vct_yuZM2sc/YXrqkxd_4GI/AAAAAAAADuc/UvpiMsWKcI4N8UPL-wVFrMDR9zUTBy4uwCLcBGAsYHQ/s1112/Screen+Shot+2021-10-28+at+1.21.51+PM.png)**

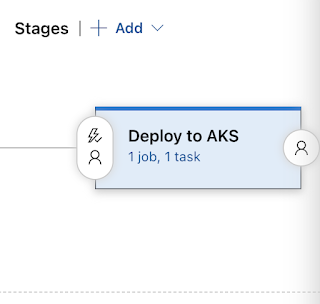
**Now click on Add an artifact**

**[](https://1.bp.blogspot.com/-dxc2Ol4lcpY/YXrsO6C9IaI/AAAAAAAADu8/UVGLRsTCSXAFBMt8MWaZX6ts7o95aSXxgCLcBGAsYHQ/s874/Screen+Shot+2021-10-28+at+1.22.03+PM.png)**

**Select the Build pipeline and click on the latest version**

[](https://1.bp.blogspot.com/-iQMVFgXHYoM/YXrsFN1DaDI/AAAAAAAADu0/Q0AH0rqVTXoCyzBitTjI92LEj41PDbi0gCLcBGAsYHQ/s1874/Screen+Shot+2021-10-28+at+1.22.33+PM.png)

Now click on Deploy to AKS stage

[](https://1.bp.blogspot.com/-YjRkCD8PvOg/YXrsw88fw5I/AAAAAAAADvM/eoff82JjVsY9Cj0KJ3amOiD0Q8IqedSeQCLcBGAsYHQ/s620/Screen+Shot+2021-10-28+at+1.32.16+PM.png)

**Add Replace token task**

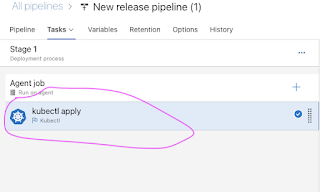
Click on + to add task, type token and choose replace token task.

Now click on replace token task and Clik on root directory, click on ... dots

 select the drop directory from below:

and enter Target file as  **aks-deploy-from-acr.yaml**

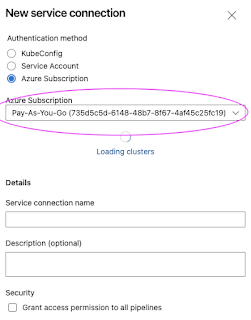
Click on kubectl apply

[](https://1.bp.blogspot.com/-CZzEh1EX194/YXr3lvBRkNI/AAAAAAAADv0/f68uIrG7aDw-EElKmuUTUDD6SvBYW0VLQCLcBGAsYHQ/s638/Screen+Shot+2021-10-28+at+2.18.11+PM.png)

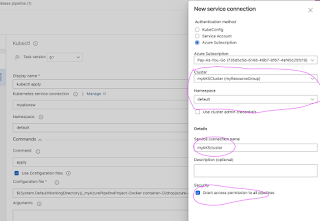
Now Click on New to enter AKS cluster connection info

[](https://1.bp.blogspot.com/-Cl1VF5D9y3g/YXr33HkkxjI/AAAAAAAADwA/7q4J0Ayn7eUVgSOpw1ADhd1OP62qMHdewCLcBGAsYHQ/s1048/Screen+Shot+2021-10-28+at+2.04.03+PM.png)

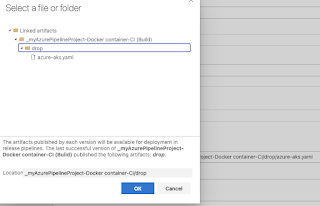
Choose the Azure subscription and enter Microsoft user credentials.

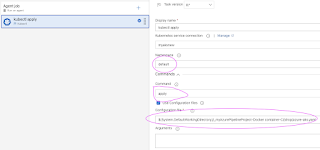
[](https://1.bp.blogspot.com/-dp0B25fZfTA/YXr3KOE-jhI/AAAAAAAADvk/WweKa_gTjA00V9e3MzDg4eN3M7G6BlcRgCLcBGAsYHQ/s587/Screen+Shot+2021-10-28+at+2.04.39+PM.png)

Select AKS cluster from the drop down, choose default namespace

[](https://1.bp.blogspot.com/-MvCryu566A0/YXr3KJanQ8I/AAAAAAAADvc/SghHNEd8YLQOXXJ3iEKDDvvhqZqlMdPTACLcBGAsYHQ/s1141/Screen+Shot+2021-10-28+at+2.16.05+PM.png)

Choose command as apply and select the yaml file from the dropdown from Configuration file

[](https://1.bp.blogspot.com/-AQbGub4x37o/YXr5KBe0DMI/AAAAAAAADwU/-mBKgqCQ6-8bMJgR7FuIJNTgeWianTa6ACLcBGAsYHQ/s868/Screen+Shot+2021-10-28+at+2.25.12+PM.png)

[](https://1.bp.blogspot.com/-lbLtwPqdlu8/YXr46PsKItI/AAAAAAAADwM/6j4yojGBVtgo7WiPmkwtTRVcsceVw4v-QCLcBGAsYHQ/s1352/Screen+Shot+2021-10-28+at+2.23.28+PM.png)

Now click on Save,

Click on Create a release

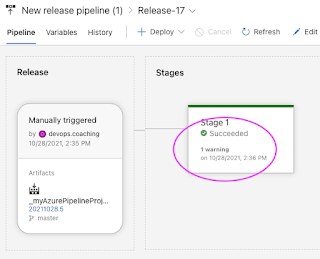
[https://1.bp.blogspot.com/--Y-NT_aZ2eY/YXr7tpaxQqI/AAAAAAAADwg/-9AMUvTgqo8mGQBlvmxoCn_m9hEzaYmBgCLcBGAsYHQ/s320/Screen%2BShot%2B2021-10-28%2Bat%2B2.35.42%2BPM.png](https://1.bp.blogspot.com/--Y-NT_aZ2eY/YXr7tpaxQqI/AAAAAAAADwg/-9AMUvTgqo8mGQBlvmxoCn_m9hEzaYmBgCLcBGAsYHQ/s1380/Screen+Shot+2021-10-28+at+2.35.42+PM.png)

and then click Create to run the deployment

[](https://1.bp.blogspot.com/-Fl3BZ93EcMY/YXr7tiQQyrI/AAAAAAAADwc/FKOrv2ilUEYNySKLAfvbmeEcgTh2XVUQACLcBGAsYHQ/s1027/Screen+Shot+2021-10-28+at+2.35.54+PM.png)

[https://1.bp.blogspot.com/-xjaSOm7WlvE/YXr8PKSQzQI/AAAAAAAADw0/832ADLrRH1QHSeuUdMVFSDWG-OiTZ3pkgCLcBGAsYHQ/s320/Screen%2BShot%2B2021-10-28%2Bat%2B2.35.42%2BPM.png](https://1.bp.blogspot.com/-xjaSOm7WlvE/YXr8PKSQzQI/AAAAAAAADw0/832ADLrRH1QHSeuUdMVFSDWG-OiTZ3pkgCLcBGAsYHQ/s1380/Screen+Shot+2021-10-28+at+2.35.42+PM.png)

Click on Stage to see the logs

[](https://1.bp.blogspot.com/-DqyhIWclXFQ/YXr8DqumJzI/AAAAAAAADww/NrQJVi1Jywo6Qc-EeTKOADcoUCwTu9OugCLcBGAsYHQ/s603/Screen+Shot+2021-10-28+at+2.37.17+PM.png)

Now you will see the following tasks are in green to confirm Deployment was successful.

Let's check if deployment created any pods

kubectl get deployments

Screen Shot 2022-06-29 at 4.08.26 PM.png

kubectl get pods

Screen Shot 2022-06-29 at 4.07.41 PM.png

kubectl get svc

Screen Shot 2022-06-29 at 4.07.19 PM.png

Now try to access spring boot application running inside AKS cluster by using external IP and port number

If you see any errors after deploying the pods, you can check the pod logs.

kubectl logs <pod\_name>

Go to the browser enter http://external IP

