**Deploy Node.Js app web using docker/CI-CD**

1. Create a sample node.js application with add DockerFile to the repository
2. Create Azure Container Register for Docker images
   1. Go to azure portal
   2. Create new > Container > Container Register
   3. Provide :-
      1. Registry Name
      2. Subscription
      3. Resource Group
      4. Location
      5. Admin user as “Enable”
      6. SKU
3. Web App Deployment using offline Docker image
   1. Install azure cli if not already using below command    
      brew update && brew install azure-cli
   2. Run docker on machine
   3. Clone the code repository(azure devops/git) to local machine and go to folder using terminal
   4. Login to azure container registry using below command    
      az acr login --name <<acrreponame>>
   5. Build image using below command    
      docker build . -f <<pathToDockerFileInCode>> -t <<ACRName>>.[azurecr.io](http://azurecr.io)/<<ImageName>>:<<TagName>>
   6. Verify Docker image creation using command docker images
   7. Push image to Azure Container Repository using below command:-   
      docker push <<ACRName>>.[azurecr.io](http://azurecr.io)/<<ImageName>>:<<TagName>>
   8. Now go to Azure Portal then
      1. Create a resource > Web > Web App for containers
      2. In Basic tab Provide :-
         1. Subscription
         2. Resource group
         3. Name for web App
         4. Publish as “Docker container”
         5. OS as “Linux”
         6. Region
      3. In Docker Tab
         1. Options as Single Container
         2. Image Source as “Azure Container Registry”
         3. Select newly created Registry <<ACRName>>
         4. Select newly created Image in previous step <<ImageName>>:
         5. Select newly created tag in previous step <<TagName>>
      4. Tap on Review/Create in button ribbon to create web app
      5. Web app will deploy using selected docker image <<TagName>>
      6. Browse the website
4. Web App Deployment using Azure DevOps CI/CD
   1. Pre Requisite
      1. Code Repository on Azure DevOps containing DockerFile
      2. Create service connection in Azure Devops Portal. Following are the details to create service connection
         1. In <https://dev.azure.com/> go to your project and left panel in bottom go to “Project Settings”
         2. In Project setting under pipeline go to “Service Connections”
         3. On Service connection page tap on new “Service connection Button” select connection type as “Docker Registery” press next
         4. On Next page Select Registry Type as “Azure Container Registry”
         5. Select Your subscription then select appropriate container registry
         6. Provide service connection name as ”dockerServiceConnection” and description and Save the connection
   2. Create a new Build pipeline as follows
      1. Go to Azure DevOps service and select the project in Azure DevOps
      2. Go to pipeline >> Build >> New Build pipeline
      3. select the appropriate source then select the repository
      4. In configure select “Docker(Build a Docker Image)” press next
      5. In next page select the docker file location press validate and confgure
      6. A YAML file will be created Attached in references
   3. Create a new Release pipeline as follows:-
      1. Go to Azure DevOps service and select the project in Azure DevOps
      2. Go to pipeline >> Release >> New Release pipeline
      3. Select template as Empty Job
      4. In Artifacts Section tap on Add an Artifacts
      5. In Add a Artifacts page select source type “Build”
         1. Select Project
         2. Select source(From build pipeline) and provide default version and source alias
         3. Also, make “Continuous deployment trigger” as required
      6. Now on Stage Section
         1. Tap on 1 job,0 task link and go to Add task section
         2. Under Agent Job tap on + sign to add task
         3. In Add task template Select “Azure App Service Deploy” and add the template
         4. In the task select following:-
            1. Connection type as “Azure resource manager”
            2. Select appropriate azure subscription
            3. App Service Type as “Web App for Container (Linux)”
            4. App Service name as <<App Name>>
            5. Registry or Name space provide registry name like below:-  
               << ACRName >>.azurecr.io
            6. Image as <<ImageName>>
            7. Tags as “$(Build.BuildId)”
         5. On Top section Save then Create Release