**Jenkins Pipeline for Building and Pushing Docker Images to ECR**

This document explains how to create a Jenkins pipeline that will build a Docker image and push it to an ECR repository.

**Prerequisites**

* Jenkins installed and configured
* AWS CLI installed on the Jenkins server
* IAM permissions to access ECR
* Docker installed on Jenkins server

**Steps**

1. Create a new pipeline job in Jenkins
2. Add the following environment variables:

Copy code

AWS\_ACCOUNT\_ID = "123456789012" AWS\_DEFAULT\_REGION = "us-east-1" IMAGE\_REPO\_NAME = "myapp" IMAGE\_TAG = "latest"

1. Add a Git SCM step to clone your code repository
2. Add a Login to ECR stage:

Copy code

sh "aws ecr get-login-password --region ${AWS\_DEFAULT\_REGION} | docker login --username AWS --password-stdin ${AWS\_ACCOUNT\_ID}.dkr.ecr.${AWS\_DEFAULT\_REGION}.amazonaws.com"

1. Add a Build Image stage:

Copy code

dockerImage = docker.build "${IMAGE\_REPO\_NAME}:${IMAGE\_TAG}"

1. Add a Pushing to ECR stage:

Copy code

sh "docker tag ${IMAGE\_REPO\_NAME}:${IMAGE\_TAG} ${REPOSITORY\_URI}:${IMAGE\_TAG}" sh "docker push ${REPOSITORY\_URI}:${IMAGE\_TAG}"

1. Save and run the pipeline

This will build the Docker image, tag it properly for ECR, login to ECR using IAM credentials, and push the image to your repository.

The full pipeline code can be found in the script provided. Update the values accordingly.