**4. Creating and Running Containers**

b. Write a document that describes how to create and run containers using the Docker command-line interface (CLI) on AWS, Azure, and GCP.

c. Include commands for pulling images from a registry, running a container, and checking the status of running containers.

**Running Docker Containers on AWS, Azure, and GCP**

**Prerequisites**

Before using Docker CLI to create and run containers on AWS, Azure, and GCP, ensure you have the following:

* Docker installed on your local machine ([Install Docker](https://docs.docker.com/get-docker/))
* Cloud provider CLI tools installed and configured:
  + AWS CLI ([Install AWS CLI](https://aws.amazon.com/cli/))
  + Azure CLI ([Install Azure CLI](https://learn.microsoft.com/en-us/cli/azure/install-azure-cli))
  + GCP CLI ([Install Google Cloud SDK](https://cloud.google.com/sdk/docs/install))

**Pulling Docker Images**

To pull an image from a container registry (Docker Hub, AWS ECR, Azure ACR, or GCP Artifact Registry):

**From Docker Hub:**

docker pull nginx:latest

**From AWS Elastic Container Registry (ECR):**

1. Authenticate with ECR:

aws ecr get-login-password --region <region> | docker login --username AWS --password-stdin <aws\_account\_id>.dkr.ecr.<region>.amazonaws.com

1. Pull the image:

docker pull <aws\_account\_id>.dkr.ecr.<region>.amazonaws.com/my-image:latest

**From Azure Container Registry (ACR):**

1. Authenticate with ACR:

az acr login --name <acr\_name>

1. Pull the image:

docker pull <acr\_name>.azurecr.io/my-image:latest

**From GCP Artifact Registry:**

1. Authenticate with GCP:

gcloud auth configure-docker

1. Pull the image:

docker pull gcr.io/<project-id>/my-image:latest

**Running a Container**

To start a container from an image:

docker run -d --name my-container -p 80:80 nginx:latest

* -d runs the container in detached mode.
* --name assigns a name to the container.
* -p 80:80 maps port 80 of the container to port 80 on the host.

**Checking the Status of Running Containers**

List running containers:

docker ps

List all containers (including stopped ones):

docker ps -a

**Stopping and Removing Containers**

To stop a running container:

docker stop my-container

To remove a stopped container:

docker rm my-container

**Conclusion**

This document provides a guide on using Docker CLI to pull, run, and manage containers on AWS, Azure, and GCP. Be sure to configure the respective cloud CLI tools properly before working with cloud container registries.