INSTITUTE OF COMPUTER TECHNOLOGY

B-TECH COMPUTER SCIENCE ENGINEERING 2025-26

SUBJECT: IDENTITY ACCESS MANAGEMENT

NAME: Rahul Prajapati

ENRLL. NO: 23162171020

BRANCH: CYBER SECURITY

BATCH: 52

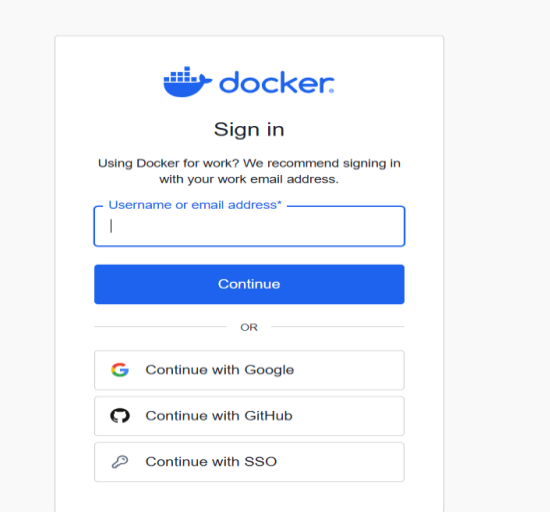
Lab 1: Installing Docker and Keycloak

STEP_1:

➔ INSTALL DOCKER DESKTOP

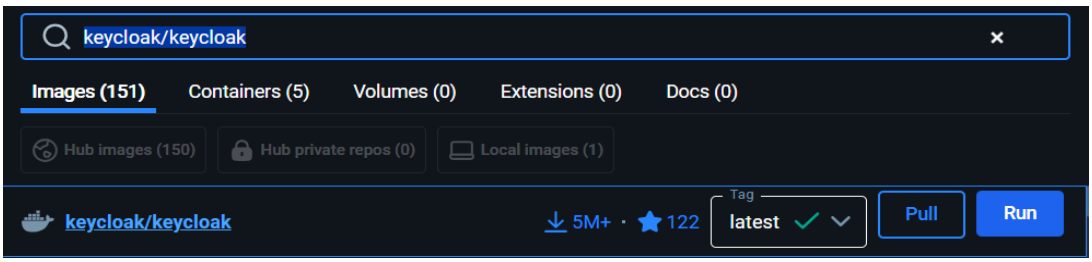
STEP_2

➔ SIGNING UP DOCKER ACCOUNT



STEP_3:

➔ PULLING KEYCLOAK IMAGE FROM DOCKER



STEP_4:

➔ RUN KEYCLOAK

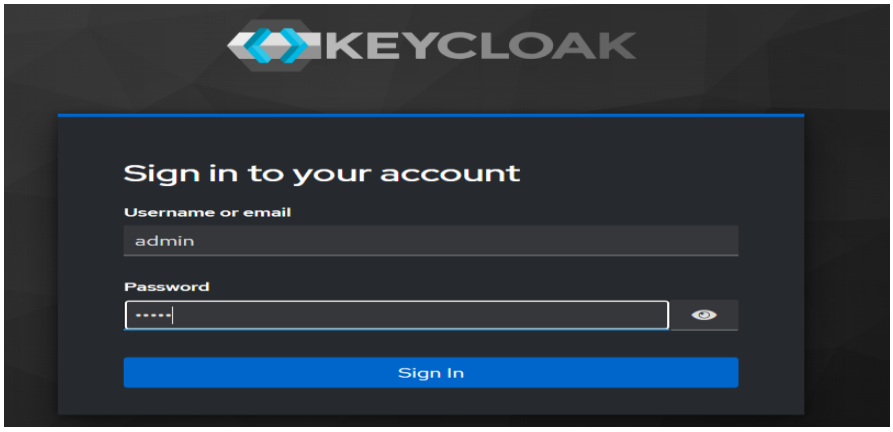
- >>>docker pull keycloak/keycloak
- >>>docker run -p 8080:8080 -e KEYCLOAK_ADMIN=admin -e KEYCLOAK_ADMIN_PASSWORD=admin keycloak/keycloak start-dev
- NOTE:RUN ON CMD

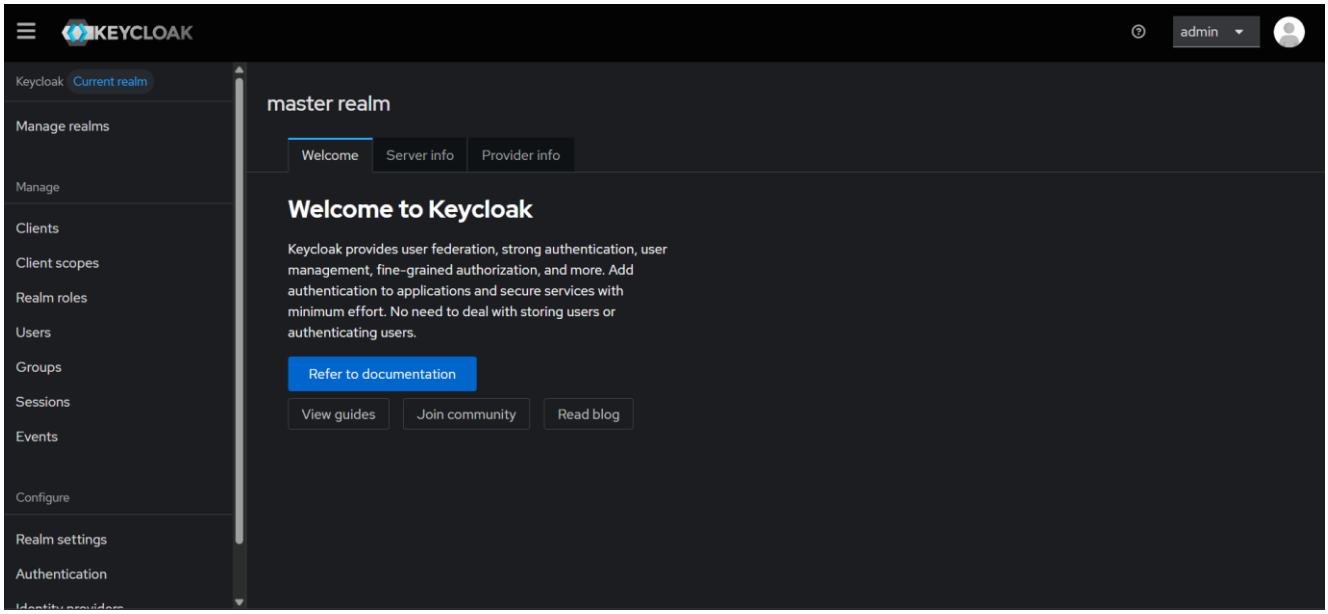
```
PS C:\Users\Hp> docker run -p 8080:8080 -e KEYCLOAK_ADMIN=admin -e KEYCLOAK_ADMIN_PASSWORD=admin keycloak/keycloak start-dev
Updating the configuration and installing your custom providers, if any. Please wait.
2025-07-25 03:25:47,822 INFO  [io.quarkus.deployment.QuarkusAugmentor] (main) Quarkus augmentation completed in 11317ms
Running the server in development mode. DO NOT use this configuration in production.
2025-07-25 03:25:56,922 INFO  [org.keycloak.quarkus.runtime.storage.database.liquibase.QuarkusJpaUpdaterProvider] (main) Initializing database schema. Using changelog META-INF/jpa-changelog-master.xml
2025-07-25 03:26:03,920 INFO  [org.keycloak.spi.infinispan.impl.embedded.JGroupsConfigurator] (main) JGroup
```

STEP_5:

➔ Run http://localhost:8080 on browser

➔ Login by username=admin & password=admin





What is the difference between docker and Virtualization? Why do we use docker?

No.	Feature	Docker (Containerization)	Virtualization (VMs)
1	**OS requirement**	Shares **host OS kernel**	Each VM has **its own full OS**
2	**Boot-up speed**	Starts in **seconds**	Takes **minutes** to boot
3	**Resource usage**	**Lightweight**, uses less RAM and CPU	**Heavy**, consumes more RAM and CPU
4	**Performance**	**Near-native** performance	Slightly **slower** due to full OS layer
5	**Size**	Small container image size (MBs)	Large VM images (GBs)
6	**Isolation level**	**Process-level** isolation (less secure than VMs)	**Full OS-level** isolation (more secure)
7	**Portability**	Highly portable across systems	Less portable (OS dependencies)
8	**Use case**	Ideal for **microservices**, CI/CD, cloud apps	Ideal for **running different OSes**, full environments
9	**Dependency handling**	Bundles all app dependencies inside the container	Needs separate setup for each VM
10	**Management**	Easier to manage with tools like Docker Compose, Kubernetes	More complex to manage multiple VMs
11	**Storage**	Containers share layers, saving space	Each VM has its own full disk image
12	**Security**	Good, but **shares kernel** – some risks	Better isolation due to **separate OS**

- Docker is used to package applications with all their dependencies into lightweight, portable containers that run consistently across different environments. It makes development, testing, and deployment faster, more reliable, and easier to scale.