

Institute of Computer Technology
B. Tech Computer Science and Engineering

Sub: Identity and Access Management (2CSE507)

Name - Nisarg Prajapati

Branch - Cyber Security

Enrollment no. – 23162171019

Semester - 5

Class - A

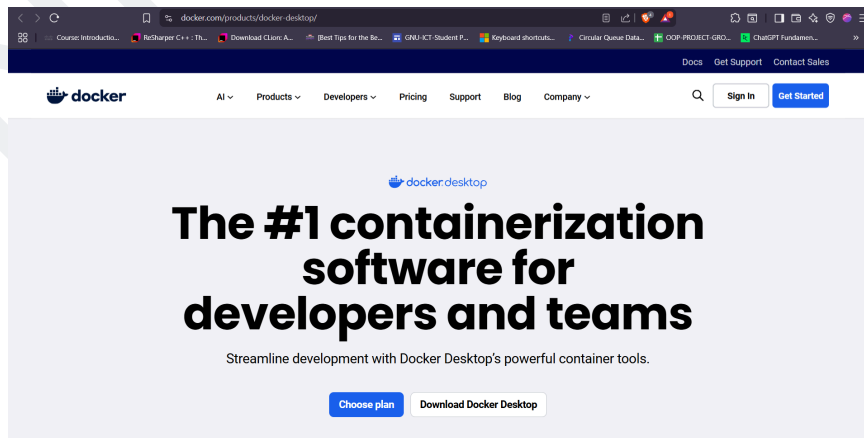
Batch – 52

PRACTICAL NO:- 1

Installing Docker:

1. Download Docker from below given URL:

<https://www.docker.com/products/docker-desktop/>



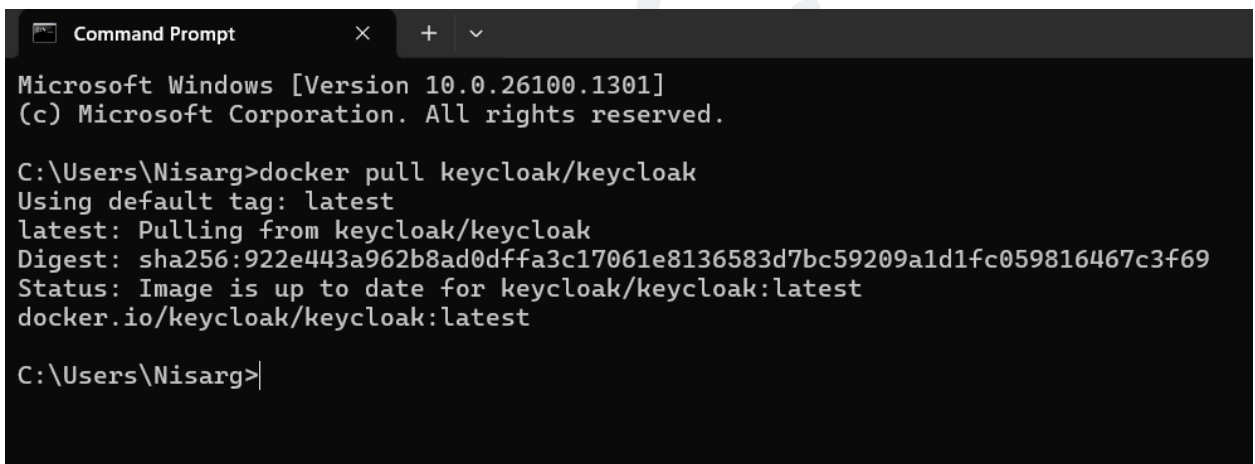
2. After the download completes, Run the installer to install Docker.
3. After installation completes, 'sign-up' for an account.

Pulling an image and Creating a Container:

1. Open command prompt (press 'win+r', then type 'cmd' and press 'enter').
2. Then type following command:

```
docker pull keycloak/keycloak
```

Explanation: The command docker pull 'keycloak/keycloak' downloads the Keycloak container image from Docker Hub to your local system. This image contains everything needed to run the Keycloak identity and access management server inside a Docker container.



```
Command Prompt
Microsoft Windows [Version 10.0.26100.1301]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Nisarg>docker pull keycloak/keycloak
Using default tag: latest
latest: Pulling from keycloak/keycloak
Digest: sha256:922e443a962b8ad0dffa3c17061e8136583d7bc59209a1d1fc059816467c3f69
Status: Image is up to date for keycloak/keycloak:latest
docker.io/keycloak/keycloak:latest

C:\Users\Nisarg>
```

3. Then in order to create an new container, use following command:

```
docker run -p 8080:8080 -e KEYCLOAK_ADMIN=admin -e
KEYCLOAK_ADMIN_PASSWORD=admin keycloak/keycloak
```

Explanation: It runs the Keycloak container and maps host port 8080 to container port 8080. It also sets the environment variables KEYCLOAK_ADMIN and KEYCLOAK_ADMIN_PASSWORD to create the initial admin user (admin/admin). This lets you access the Keycloak admin console via 'http://localhost:8080'.

```

Command Prompt
C:\Users\Nisarg>docker run -p 8080:8080 -e KEYCLOAK_ADMIN=admin -e KEYCLOAK_ADMIN_PASSWORD=admin keycloak/keycloak
Keycloak - Open Source Identity and Access Management

Find more information at: https://www.keycloak.org/docs/latest

Usage:
kc.sh [OPTIONS] [COMMAND]

Use this command-line tool to manage your Keycloak cluster.

Options:
-cf, --config-file <file>      Set the path to a configuration file. By default, configuration properties are
                                read from the "keycloak.conf" file in the "conf" directory.
-h, --help                    This help message.
-v, --verbose                 Print out error details when running this command.
-V, --version                 Show version information

Commands:
build                          Creates a new and optimized server image.
start                          Start the server.
start-dev                      Start the server in development mode.
export                         Export data from realms to a file or directory.
import                        Import data from a directory or a file.
show-config                   Print out the current configuration.
tools                         Utilities for use and interaction with the server.
    completion               Generate bash/zsh completion script for kc.sh.
bootstrap-admin              Commands for bootstrapping admin access
    user                     Add an admin user with a password
    service                 Add an admin service account
update-compatibility         [Preview] Tool for configuration compatibility.
check                       Checks if the metadata is compatible with the current
                                configuration. A zero exit code means a rolling
                                update is possible between old and the current
                                metadata.
metadata                     Stores the metadata necessary to determine if a
                                configuration is compatible.

```

```

Command Prompt
export                         Export data from realms to a file or directory.
import                        Import data from a directory or a file.
show-config                   Print out the current configuration.
tools                         Utilities for use and interaction with the server.
    completion               Generate bash/zsh completion script for kc.sh.
bootstrap-admin              Commands for bootstrapping admin access
    user                     Add an admin user with a password
    service                 Add an admin service account
update-compatibility         [Preview] Tool for configuration compatibility.
check                       Checks if the metadata is compatible with the current
                                configuration. A zero exit code means a rolling
                                update is possible between old and the current
                                metadata.
metadata                     Stores the metadata necessary to determine if a
                                configuration is compatible.

Examples:
Start the server in development mode for local development or testing:

    $ kc.sh start-dev

Building an optimized server runtime:

    $ kc.sh build <OPTIONS>

Start the server in production mode:

    $ kc.sh start <OPTIONS>

Enable auto-completion to bash/zsh:

    $ source <(kc.sh tools completion)

Please, take a look at the documentation for more details before deploying in
production.

Use "kc.sh start --help" for the available options when starting the server.
Use "kc.sh <command> --help" for more information about other commands.

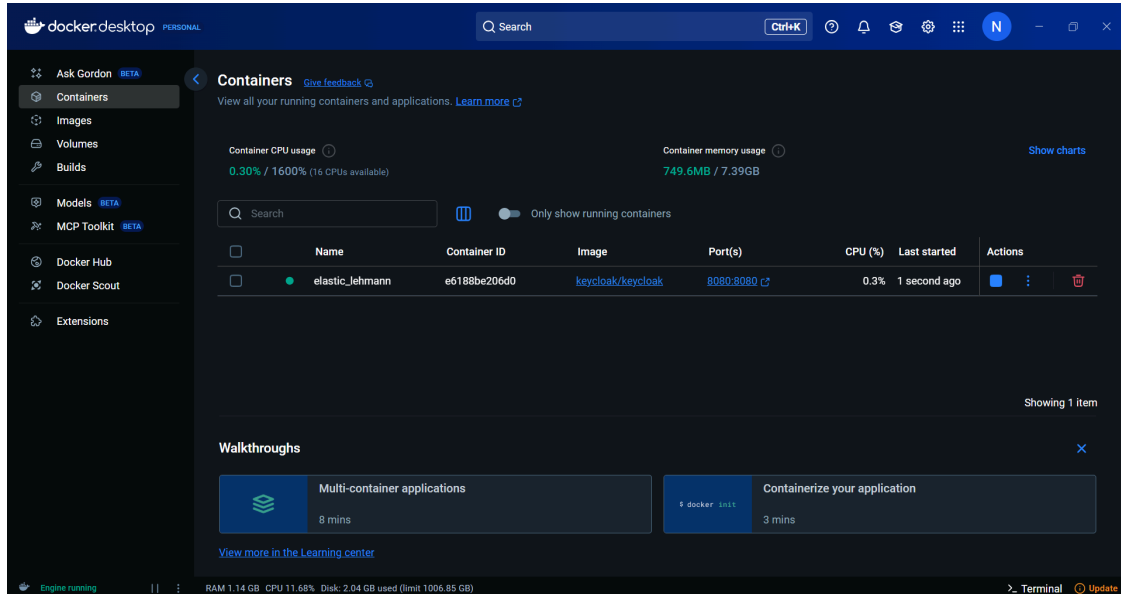
C:\Users\Nisarg>

```

4. Now you will be able to access the container through your Docker Desktop

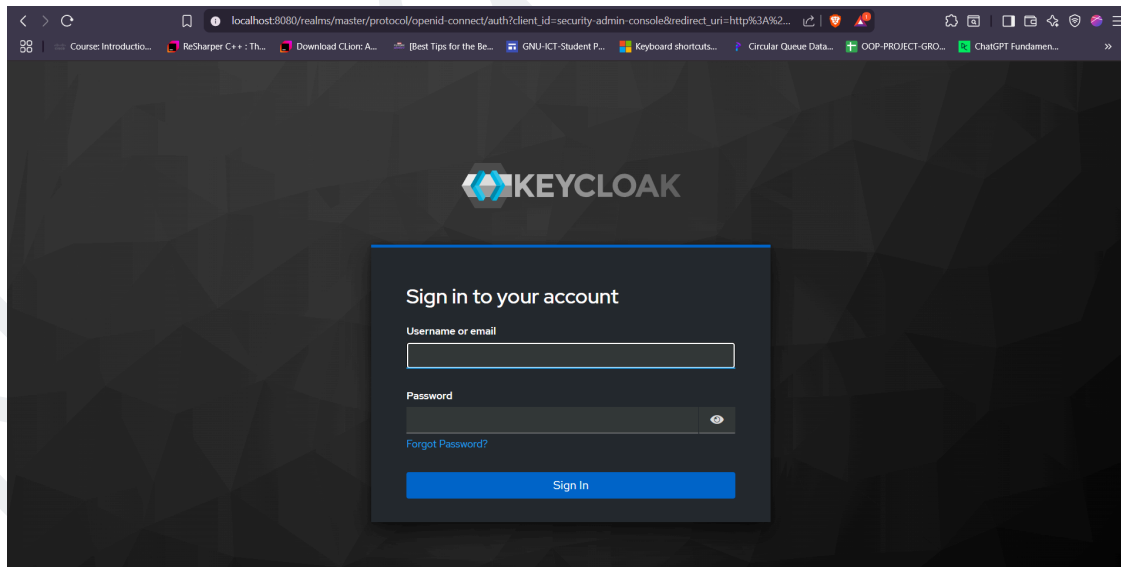
Running a Container:

1. Open your Docker Desktop and go to the container tab.
2. Now, click on the start button of the container you just created.

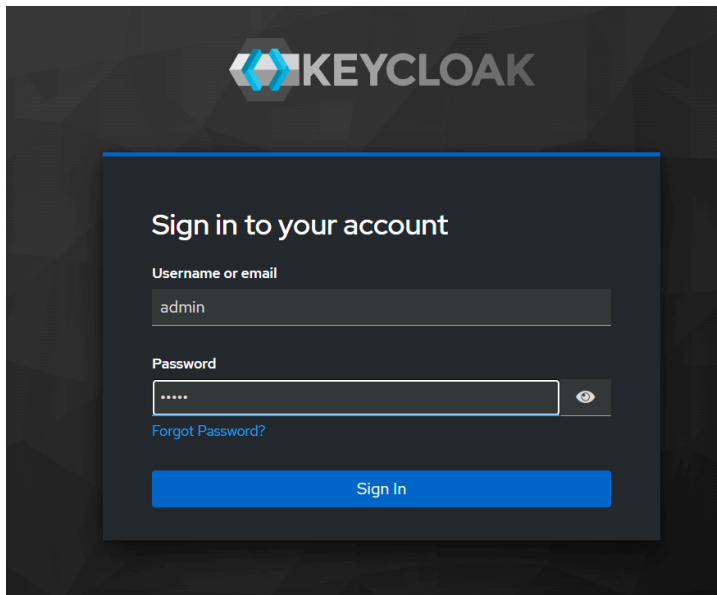


3. Open your browser and type following url:

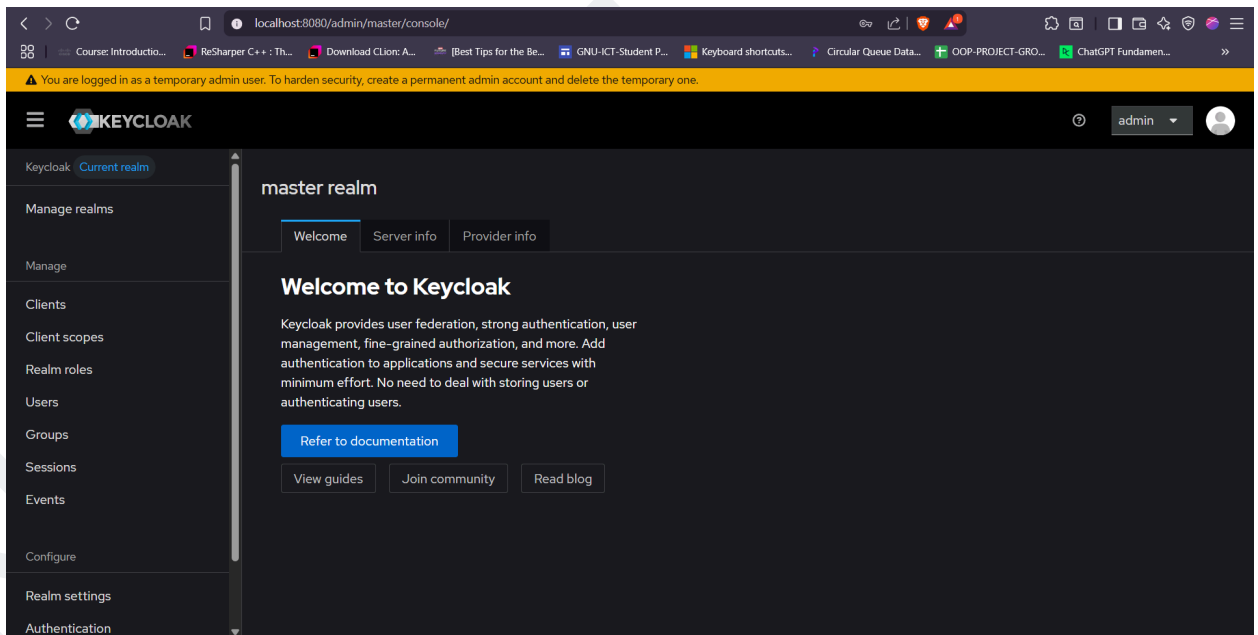
'localhost:8080'



4. Use the admin login credentials, that you declared during creating the container. (In this case, username: admin and password: admin).



5. Now, you will be able to access admin control of the system.



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

Features	Docker	Hypervisor
Type	Container-based virtualization	Hardware-based (VM) virtualization
Function	Runs applications in isolated containers	Runs entire OS instances (VMs)
Abstraction Level	OS-level virtualization	Hardware-level virtualization
What It Virtualizes	Only the application and its dependencies	Entire operating system and hardware
Host OS Shared?	Yes (containers share the same host OS kernel)	No (each VM runs its own OS instance)
Guest OS Required	No	Yes
Resource Usage	Lightweight (no guest OS overhead)	Heavy (each VM needs its own OS)
Performance	Near-native	Slightly lower due to VM overhead
Isolation Level	Process-level (less isolated)	Full OS-level (more isolated)
Security	Shared kernel (risk if compromised)	Better security due to full isolation
Portability	Very portable (container images)	Less portable (VM images are larger)
Typical Use Case	Microservices, DevOps, CI/CD, lightweight apps	Full OS testing, legacy app support, multiple OS