

Placement Empowerment Program Cloud Computing and DevOps Centre

***Installing Docker and Running Your First Container
on Windows***

Name: Vignesh V

Department : CSE

Introduction and Overview

Docker is a powerful tool that allows developers to create, deploy, and run applications in isolated environments called containers. These containers ensure that applications run consistently across different systems, eliminating compatibility issues.

In this guide, we will go through the step-by-step process of installing Docker on Windows, setting up a basic Nginx web server inside a container, and accessing it through a browser.

Objective

- ☒ To install Docker on Windows and verify its functionality.
- ☒ To learn how to pull and run a basic Nginx container.
- ☒ To understand how to access a containerized web application using a browser.
- ☒ To gain hands-on experience with containerization and Docker commands.

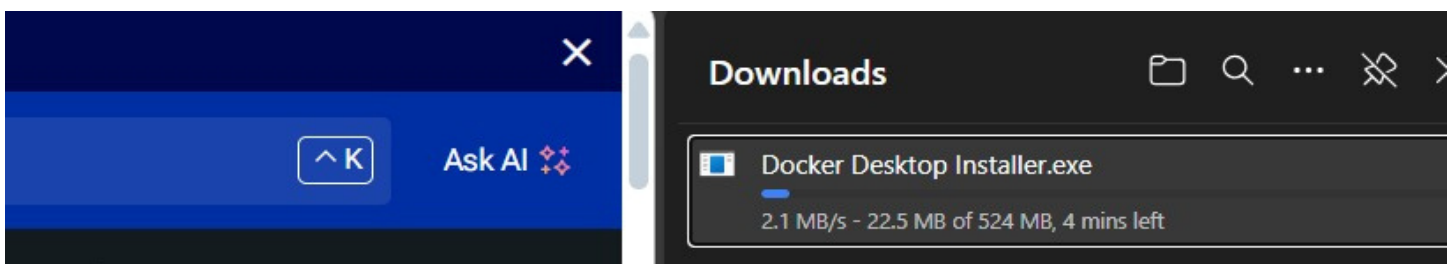
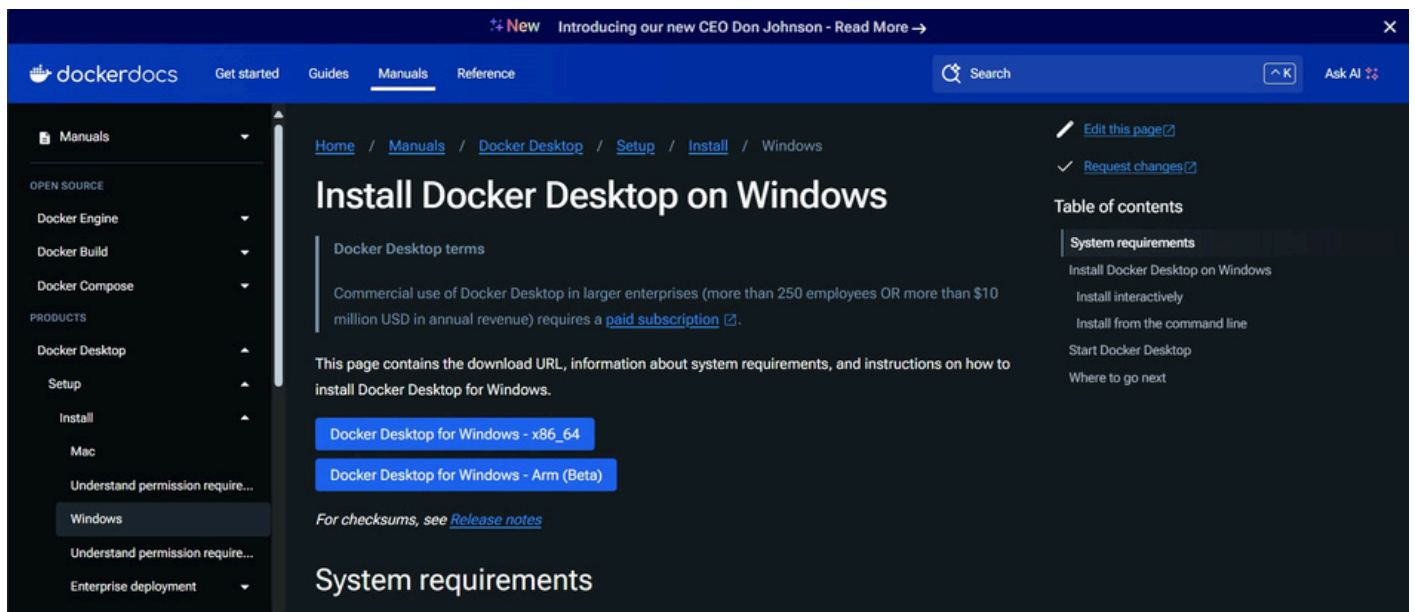
Importance

- ☒ Simplifies Deployment – Containers eliminate compatibility issues by packaging applications with all their dependencies.
- ☒ Improves Efficiency – Docker containers are lightweight and use system resources more effectively than virtual machines.
- ☒ Enhances Portability – Applications run consistently across different environments (local, cloud, or server).
- ☒ Boosts Scalability – Docker makes it easier to scale applications up or down based on demand.
- ☒ Speeds Up Development – Developers can quickly create isolated environments for testing and debugging.

Step-by-Step Overview

Step 1: Install Docker Desktop.

1. Download Docker Desktop for Windows from the official website:
<https://www.docker.com/products/docker-desktop/>
2. Run the installer and follow the on-screen instructions.
3. Ensure WSL 2 is enabled (Docker requires this for Windows).
4. Restart your PC and launch Docker Desktop.



Step 2: Verify Docker Installation

- ☒ Open PowerShell and check if Docker is installed by running the version command.
- ☒ Verify that Docker is running properly by checking its system information.

- If there are any errors, ensure Docker Desktop is open and running in the background

```
Windows PowerShell
PS C:\Users\madhu> docker images
REPOSITORY    TAG       IMAGE ID   CREATED   SIZE
PS C:\Users\madhu> docker version
Client:
 Version:      27.5.1
 API version:  1.47
 Go version:   go1.22.11
 Git commit:   9f9e405
 Built:        Wed Jan 22 13:41:44 2025
 OS/Arch:      windows/amd64
 Context:      desktop-linux

Server: Docker Desktop 4.38.0 (181591)
Engine:
 Version:      27.5.1
 API version:  1.47 (minimum version 1.24)
 Go version:   go1.22.11
 Git commit:   4c9b3b0
 Built:        Wed Jan 22 13:41:17 2025
 OS/Arch:      linux/amd64
 Experimental: false
 containerd:
 Version:      1.7.25
 GitCommit:    bcc810d6b9066471b0b6fa75f557a15a1cbf31bb
 runc:
 Version:      1.1.12
 GitCommit:    v1.1.12-0-g51d5e946
 docker-init:
 Version:      0.19.0
 GitCommit:    de40ad0
```

Step 3: Pull the Nginx Docker Image

- Use the Docker pull command to download the latest Nginx image from Docker Hub.
- Once the image is downloaded, verify it by listing all available images in Docker.

```
Command Prompt
Microsoft Windows [Version 10.0.22631.4751]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin> docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES

C:\Users\Admin> docker pull nginx
Using default tag: latest
latest: Pulling from library/nginx
103f50cb3e9f: Download complete
513c3649bb14: Download complete
7cf63256a31a: Download complete
943ea0f0c2e4: Download complete
bf9acace214a: Download complete
d014f92d532d: Download complete
9dd21ad5a4a6: Download complete
Digest: sha256:9d6b58feebd2dbd3c56ab5853333d627cc6e281011cfd6050fa4bcf2072c9496
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest

C:\Users\Admin> docker run -d -p 8080:80 --name my-nginx nginx
095c417d71e0b08df2cd4c2ed54191151defde03043255ca75095ea6fd51e9b5

C:\Users\Admin>
```

Step 4: Run the Nginx Container

- ❓ Start an Nginx container by running it in detached mode and mapping it to port 8080.
- ☒ Verify that the container is running by listing all active containers.

```
C:\Users\Admin> docker run -d -p 8080:80 --name my-nginx nginx
095c417d71e0b08df2cd4c2ed54191151defde03043255ca75095ea6fd51e9b5
C:\Users\Admin>
```

Step 5: Access the Nginx Web Page

- Open a web browser and go to <http://localhost:8080>.
- If everything is set up correctly, the default Nginx welcome page should appear.

```
PS C:\Users\Admin> docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                   NAMES
6af08de211dd   nginx    "/docker-entrypoint..." 50 seconds ago Up 48 seconds  0.0.0.0:8080->80/tcp    my-nginx
PS C:\Users\Admin>
```

```
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                   NAMES
6af08de211dd   nginx    "/docker-entrypoint..." 50 seconds ago Up 48 seconds  0.0.0.0:8080->80/tcp    my-nginx
PS C:\Users\Admin> docker inspect my-nginx | Select-String '"HostPort": "8080"'
"HostPort": "8080"
"HostPort": "8080"
```

localhost:8080

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

Step 6: Stop and Remove the Container

- ☒ If you no longer need the container, stop it using the stop command.
- ☒ Remove the stopped container from Docker.
- ☒ Optionally, remove the Nginx image if you want to free up space.

```
PS C:\Users\Admin> docker stop my-nginx|
```

```
PS C:\Users\Admin> docker start my-nginx|
```

```
PS C:\Users\Admin> docker rmi nginx|
```

```
PS C:\Users\Admin> docker rmi nginx
Untagged: nginx:latest
Deleted: sha256:91734281c0ebfc6f1aea979cffe5079cfe786228a71cc6f1f46a228cde6e34
PS C:\Users\Admin> |
```

Expected Outcome

- ☒ Successful installation of Docker Desktop on Windows.
- ☒ Verification that Docker is running correctly through PowerShell commands.
- ☒ Pulling and running an Nginx container without errors.
- ☒ Accessing the Nginx default welcome page in a web browser at <http://localhost:8080>.
- ☒ Understanding basic Docker commands like pull, run, stop, and remove.