**Are you excited to do some simple and advanced projects on Docker?**

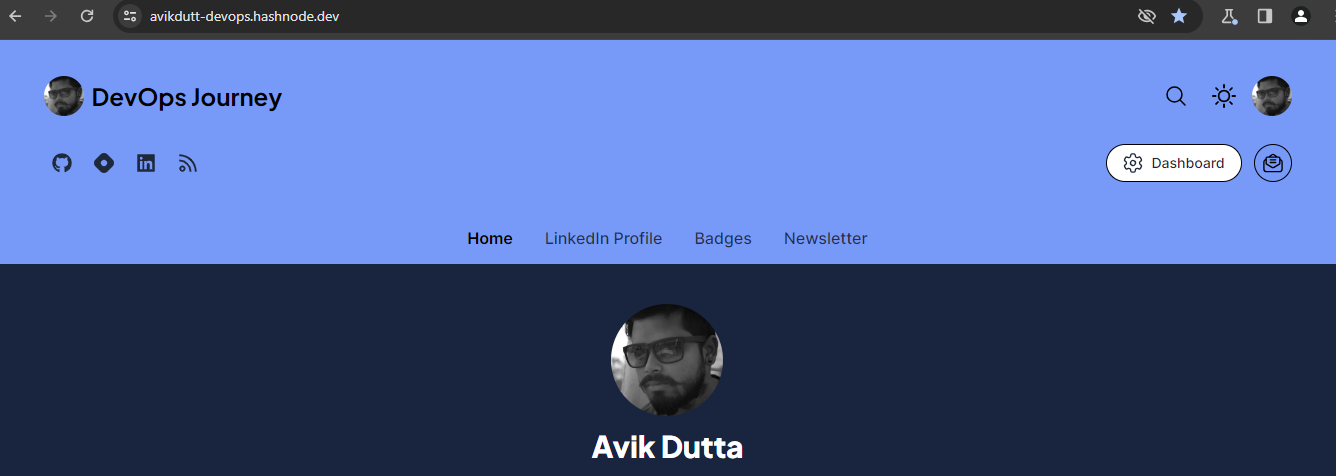
Let’s complete the below perquisites quickly and confirm in the comment box. You can share/repost so that it can reach many people who all are interested to be with your DevOps journey!

Prerequisites:

1. AWS free tier account and an EC2 instance(Ubuntu)/ [you can do it locally in your machine as well]
2. Basic Linux commands
3. GitHub account and Git basics commands

All the above topics have been covered in my Blog. You can go through it in case you have any queries. Also comment box is opened; feel free to post your doubts.

My Blog link, series [DevOps Journey](https://avikdutt-devops.hashnode.dev/): <https://avikdutt-devops.hashnode.dev/>



**Before starting the projects lets create an account on DockerHub.**

What is DockerHub?

Docker Hub is a cloud-based registry service provided by Docker, Inc. It serves as a central repository for Docker images, which are lightweight, standalone, executable packages that include everything needed to run a piece of software, including the code, runtime, libraries, and system tools. Docker Hub allows developers to share and distribute their Docker images, making it easier for others to access and deploy software in a consistent and reproducible manner.

Key features of Docker Hub include:

**Image Repositories**: Users can store and manage Docker images in repositories on Docker Hub. Repositories can be public, allowing anyone to access the images, or private, restricting access to authorized users.

**Collaboration**: Docker Hub facilitates collaboration among developers and teams by providing a platform for sharing and versioning Docker images. Multiple contributors can work on the same project and share their images through Docker Hub.

**Automated Builds**: Docker Hub supports automated builds, which allow developers to automatically build and update Docker images whenever changes are pushed to a connected source code repository (e.g., GitHub). This ensures that the images are always up-to-date with the latest code.

**Integration with Docker CLI**: Docker Hub is seamlessly integrated with the Docker command-line interface (CLI), making it easy for users to pull, push, and manage Docker images directly from the command line.

**Official Images**: Docker Hub hosts a collection of official images maintained by various organizations and software vendors. These images are considered reliable and are a good starting point for many common software stacks.

Developers commonly use Docker Hub to find, share, and distribute containerized applications, services, and development environments. It plays a crucial role in the Docker ecosystem and supports the principles of containerization, making it simpler to deploy and manage applications across different environments.

**Steps to create account on DockerHub**

Creating an account on Docker Hub is a straightforward process. Follow these steps to create your Docker Hub account:

1. Visit the Docker Hub Website: <https://hub.docker.com/>
2. Click on "Sign Up": On the Docker Hub homepage, locate the "Sign Up" button and click on it.
3. Fill in Registration Details:
   1. Enter your desired Docker ID.
   2. This is a unique username that will be associated with your Docker Hub account.
   3. Provide a valid email address.
   4. Choose a strong password for your account.
4. Agree to Terms of Service: Read the Terms of Service and Privacy Statement. If you agree, check the box to accept the terms.
5. Complete the reCAPTCHA: Complete any reCAPTCHA verification that may be required to prove that you are not a robot.
6. Verify Your Email: After submitting your registration details, Docker Hub will send a verification email to the address you provided. Check your email inbox for a message from Docker and click on the verification link within the email to confirm your registration.
7. Set Up Your Profile (Optional): Once your account is verified, you can optionally set up your Docker Hub profile. This may include adding a profile picture, a short bio, and other information.
8. Log In: Return to the Docker Hub website and click on the "Log In" button. Enter your Docker ID (or email) and password to log in.

Congratulations! You have now successfully created an account on Docker Hub. You can use this account to publish and share Docker images, as well as to explore and download images created by other users.