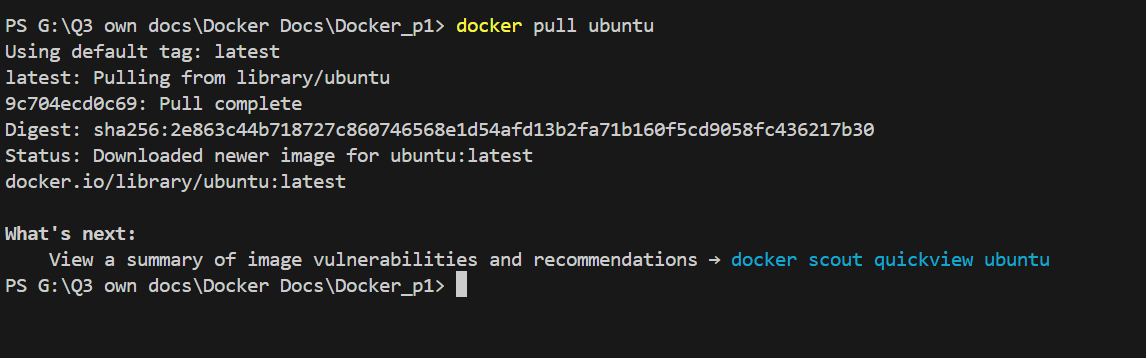
1. Go on Docker Hub
2. Choose anyone image from Docker hub that you want to run. (in our case we are choosing ubuntu OS image).
3. Copy the pull command of image.
4. Open a folder in vscode , acha yeh jo folder open kreingay isme basic kuch be install nhi hoga yeh just wesay hi bnaya hai.
5. Open terminal and paste the command that you have copied.



Now you can see jo Docker pay already created image thi wo humna apnay system may download krli. Or isme jo understand krnay wali cheez hai wo yeh haka agar **Status** wala option check toh usmay likha hai **Downloaded newer image for ubuntu,** its mean that pehla Docker Daemon check krega agar image local system may already hai toh wo usi image ko update krdega But agar image nhi hai toh usko Docker Hub say download krega.

**\*\*Docker Image Pull Process\*\*:**

1. \*\*Local Check\*\*:

- Jab aap `docker pull ubuntu` command dete hain, Docker daemon pehle aapke local system par check karega ke kya `ubuntu:latest` image pehle se available hai ya nahi.

2. \*\*Image Already Present\*\*:

- \*\*Case 1\*\*: Agar image local system par already present hoti, aur wohi version (tag) hai jo aap pull kar rahe hain, toh Docker daemon usko dobara download nahi karega.

- \*\*Case 2\*\*: Agar local system par image already hai lekin Docker Hub par us image ka naya version available hai, toh Docker daemon nayi image ko download karke existing image ko update kar dega. Aapke case mein, message "Downloaded newer image for ubuntu:latest" is baat ka indication hai ke nayi image download ki gayi hai.

3. \*\*Image Not Present Locally\*\*:

- Agar image local system par available nahi hai, toh Docker daemon Docker Hub se image ko download karega aur phir se usko local system par store kar lega taake next time direct use ho sake.

------------------------------------------------------------------------------------------------------------------

**Ab question yeh haka wo Docker Hub say kiu download krega or be toh registries hain jaha image store hogi ?**

* Toh uski waja yeh haka by Default Docker agar local system may image na milay toh uski Docker Hub say download krta hai But agar hum chahay toh we can also explicitly define registry.

**Default Behavior:**

* **Docker Hub as Default Registry**:
  + Jab aap Docker se koi image pull karte hain aur aapne koi specific registry mention nahi ki, toh Docker by default Docker Hub se image download karta hai. Yeh is liye hota hai kyunki Docker Hub Docker ka official public registry hai aur sabse zyada use hone wala registry bhi hai.

**Custom Registry Define Karna:**

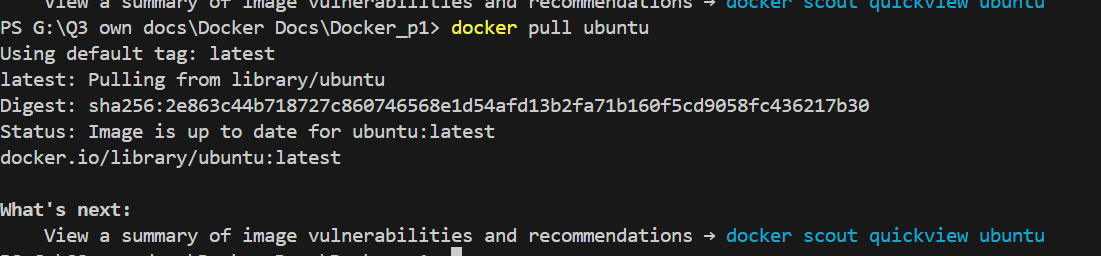
* **Specify a Different Registry**:
  + Aap Docker ko explicitly bata sakte hain ke aap kis registry se image pull karna chahte hain. Yeh useful hota hai jab aap private registries use kar rahe hain, jaise AWS ECR, Google Container Registry (GCR), ya koi aur on-premise Docker Registry.
  + Example:

bash

**docker pull myregistry.com/myimage:latest**

* + Is command mein, myregistry.com custom registry hai jahan se Docker image pull karega, instead of Docker Hub.

------------------------------------------------------------------------------------------------------------------



For checking jab mena again usi image ko pull kia toh iss dafa Docker Daemon nay image ko download nhi kia balkay update krdia.

1. Now as we know that we have to create a **Container**  for running the image. So for creating container we will use this command **docker -it imageName** , in our case we will write **docker -it ubuntu** . Remember jo container create hoga uska Docker by default koi be random name set krdega but agr hum apni marzi ka name rakhna chahtay hain toh the command is mentioned in below commands.

Some other way You can run a Docker image to create a container using the following command:

**\*\*Basic Command\*\*:**

**docker run <image-name>**

**\*\*Additional Options\*\*:**

- \*\*Detached Mode\*\*: Run in the background.

**docker run -d <image-name>**

- \*\*Interactive Mode\*\*: Keep the container interactive.

**docker run -it <image-name>**

- \*\*Map Ports\*\*: Map container ports to host ports.

**docker run -p <host-port>:<container-port> <image-name>**

- \*\*Name the Container\*\*: Give the container a specific name.

**docker run --name <container-name> <image-name>**

Also remember to run dockerfile or docker image or docker container . you must have run docker engine first , which means you must have to on Docker desktop.