


Build a Docker image from source code

Link For Docker Hub



https://hub.docker.com/repository/docker/ankitojha2705/plant_identifier_app/general



ankitojha2705 / Repositories / plant_identifier_app / General Using 0 of 1 private repositories.

General Tags Builds Collaborators Webhooks Settings

ankitojha2705/plant_identifier_app 

Last pushed 1 day ago

This repository does not have a description   INCOMPLETE

This repository does not have a category   INCOMPLETE

Docker commands



To push a new tag to this repository:

```
docker push ankitojha2705/plant_identifier_app:tagname
```

[Public View](#)

Tags


This repository contains 1 tag(s).

Tag	OS	Type	Pulled	Pushed
 latest		Image	a day ago	a day ago

[See all](#)

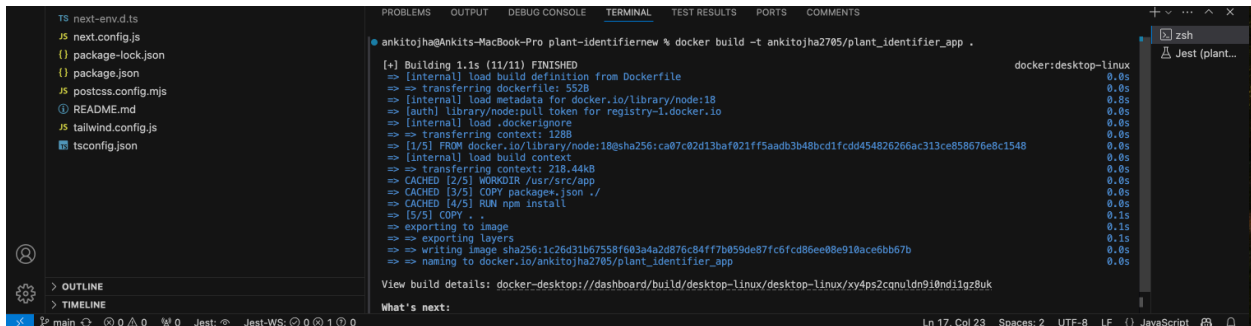
Automated Builds

Manually pushing images to Hub? Connect your account to GitHub or Bitbucket to automatically build and tag new images whenever your code is updated, so you can focus your time on creating.

Available with Pro, Team and Business subscriptions. [Read more about automated builds](#) .

[Upgrade](#)

Step 1: Build Docker Image

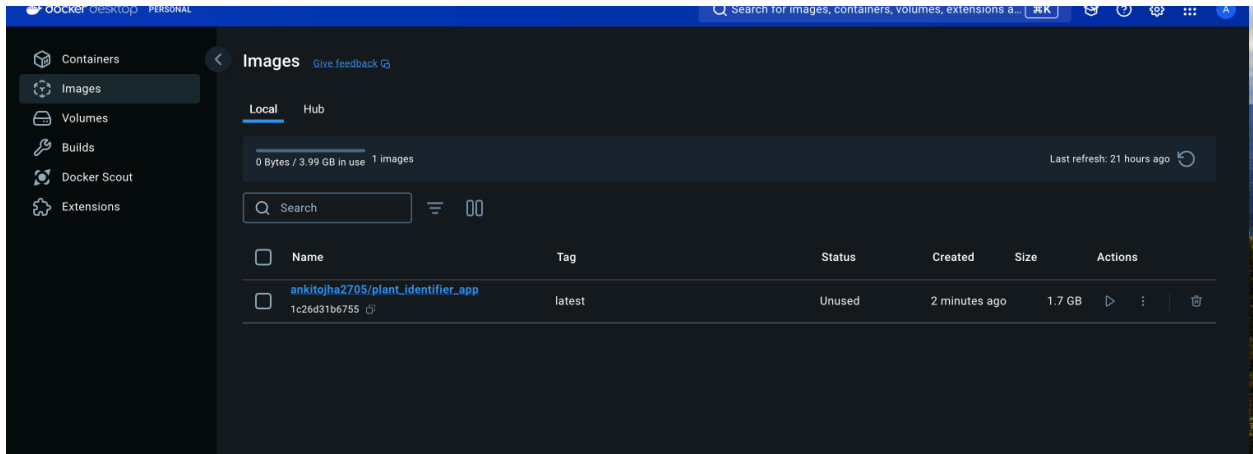


```
ankitojha@Ankits-MacBook-Pro plant-identifier$ docker build -t ankitojha2705/plant_identifier_app .
[+] Building 1.1s (11/11) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 552B
=> [internal] load metadata for docker.io/library/node:18
=> [auth] library/node:pull token for registry-1.docker.io
=> [internal] load .dockerignore
=> => transferring context: 128B
=> [1/5] FROM docker.io/library/node:18@sha256:ca07c02d13baef021ff5aadb3b48bcd1fcd454826266ac313ce858676e8c1548
=> [internal] load build context
=> => transferring context: 218.44kB
=> CACHED [2/5] WORKDIR /usr/src/app
=> CACHED [3/5] COPY package*.json ./
=> CACHED [4/5] RUN npm install
=> [5/5] COPY . .
=> exporting to image
=> writing image sha256:1c26d31b67558f683a4a2d876c84ff7b059de87f6cfd86ee08e910acebb67b
=> naming to docker.io/ankitojha2705/plant_identifier_app

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/xy4ps2cqnuldn9i0ndilgz8uk

What's next:
```

Step 2: Image Created



Step 3: Docker Run

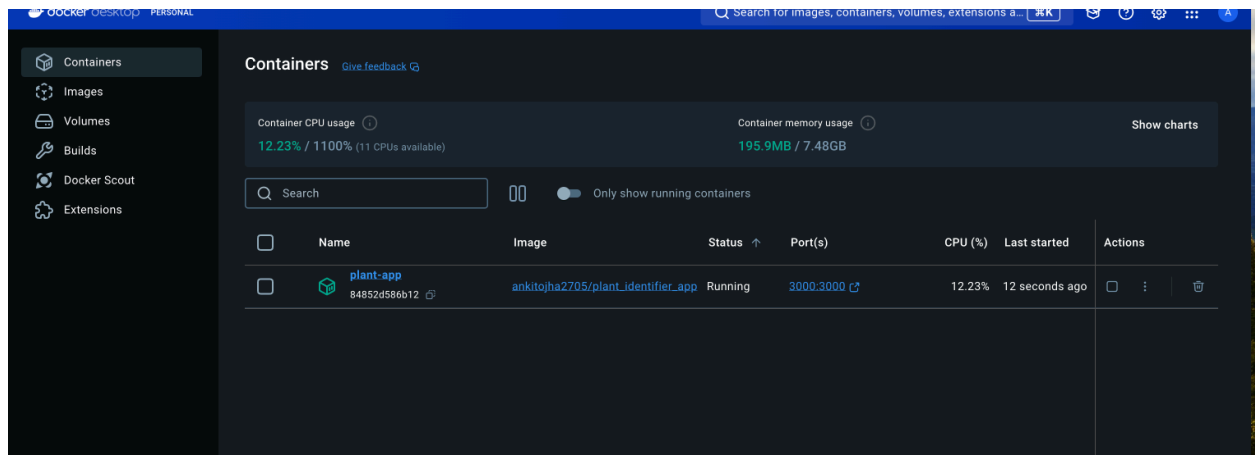
```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  TEST RESULTS  PORTS  COMMENTS

=> => exporting layers                                0.1s
=> => writing image sha256:1c26d31b67558f603a4a2d876c84ff7b059de87fc6fcd86ee08e910ace6bb67b  0.0s
=> => naming to docker.io/ankitojha2705/plant_identifier_app  0.0s

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/xy4ps2cqnuldn9i0ndi1gz8uk

What's next:
  View a summary of image vulnerabilities and recommendations → docker scout quickview
● ankitojha@Ankits-MacBook-Pro plant-identifiernew % docker run -d -p 3000:3000 --name plant-app ankitojha2705/plant_identifier_app
84852d586b1214fab640b4bb289254719b6ad3b2b2a22955130d9075050ac26
○ ankitojha@Ankits-MacBook-Pro plant-identifiernew % docker build -t ankitojha2705/plant_identifier_app
```

Step 4: Container Created



Step 5: Application Running

