

# 11.Iniciando imagem com nome

terça-feira, 15 de agosto de 2023

15:23

## Iniciando imagem com um nome

- Podemos **nomear a imagem já na sua criação**;
- Vamos utilizar a **flag -t**;
- É possível inserir o nome e a tag, na sintaxe: **nome:tag**
- Isso torna o processo de nomeação mais simples;

```
PS D:\Docker> docker build -t meunode .
[+] Building 0.9s (10/10) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 158B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/node:latest
=> [1/5] FROM docker.io/library/node
=> [internal] load build context
=> => transferring context: 78.14kB
=> CACHED [2/5] WORKDIR /app
=> CACHED [3/5] COPY package*.json .
=> CACHED [4/5] RUN npm install
=> CACHED [5/5] COPY . .
=> exporting to image
=> => exporting layers
=> => writing image sha256:4f3375e19f9c51a42e9542c08931c9749e6c4df3b1f03c86d26ec9c90ebacdb7
=> => naming to docker.io/library/meunode
```

Ou podemos criar já com a tag

```
PS D:\Docker> docker build -t minhatag:novatag .
[+] Building 2.9s (10/10) FINISHED
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 158B
=> [internal] load metadata for docker.io/library/node:latest
=> [1/5] FROM docker.io/library/node
=> [internal] load build context
=> => transferring context: 78.00kB
=> CACHED [2/5] WORKDIR /app
=> CACHED [3/5] COPY package*.json .
=> CACHED [4/5] RUN npm install
=> CACHED [5/5] COPY . .
=> exporting to image
=> => exporting layers
=> => writing image sha256:4f3375e19f9c51a42e9542c08931c9749e6c4df3b1f03c86d26ec9c90ebacdb7
=> => naming to docker.io/library/minhatag:novatag
```