DIFERENÇA IMAGEM E CONTAINER

**Imagem**

* **O que é:**  
  Um **modelo** ou **template** somente leitura que contém tudo o que é necessário para executar um software: código, bibliotecas, dependências, variáveis de ambiente e instruções de configuração.
* **Analogia:**  
  É como a receita de um bolo — ela descreve os ingredientes e o modo de preparo, mas não é o bolo em si.
* **Características:**
  + Imutável (não muda depois de criada).
  + Pode ser armazenada localmente ou no **Docker Hub**/outros registries.
  + É criada a partir de um **Dockerfile** ou de outra imagem.
  + Exemplo: nginx:latest, python:3.11, meuapp:v1.

**Container**

* **O que é:**  
  Uma **instância em execução** de uma imagem.
* **Analogia:**  
  É o bolo já pronto a partir da receita. Você pode fazer quantos bolos quiser a partir da mesma receita.
* **Características:**
  + É baseado em uma imagem, mas pode ter estado próprio (arquivos gerados, configurações alteradas enquanto está rodando).
  + É efêmero por padrão (se parar e remover, as alterações internas somem, a menos que você use volumes).
  + Cada container é isolado, mas pode se comunicar com outros via rede do Docker.
  + Exemplo: um container rodando a imagem nginx na porta 80 do seu PC.

| **Imagem** | **Container** |
| --- | --- |
| Modelo / blueprint | Instância em execução |
| Somente leitura | Pode ter estado temporário |
| Criada com Dockerfile | Criada a partir de uma imagem |
| Pode ter várias versões | Pode haver vários containers da mesma imagem |
| Não "roda" por si só | Roda processos e serviços |