



イメージの作成

イメージを作成するには、プロジェクトのルートに新しいファイルを用意する必要があります。

このファイルは拡張子なしで "Dockerfile" という名前にする必要があります。

最低限必要な構成

このファイルには少なくとも次の内容が必要です：

```
FROM php:8.2-apache
```

`FROM` は、どのイメージを元にするかを指定します。 `scratch`：ベースイメージが不要な場合には "scratch" キーワードを使うこともできます。

コマンドとコピー

```
FROM php:8.2-apache
COPY . /var/www/html
RUN apt update && apt upgrade
```

- `COPY` は、どのファイルやフォルダをどこにコピーするかを指定します。
- `RUN` は、コンテナ上でコマンドを実行するための命令です。
- 複数のコマンドを実行したい場合は、`&&` でつなぐか、複数の `RUN` 命令を書くことができます。

より見やすくするために、バックスラッシュ (`\`) で改行をエスケープすることも可能です。

上記の例では、`php:8.2-apache` イメージを使用し、現在のプロジェクト内のすべてのファイルとフォルダ (`.`) を `/var/www/html` にコピーしています。

なお、`ADD` 命令は `COPY` と似ていますが、URL を扱える点が異なります。

ポートと環境変数の指定

デフォルトで公開するポートや、環境変数も定義できます：

```
FROM mariadb
EXPOSE 3306
ENV MARIADB_ROOT_PASSWORD=root
```

- `EXPOSE` はデフォルトで公開するポート番号を指定します。
- `ENV` は環境変数を設定します。
- 複数の環境変数を定義したい場合は、スペースで区切るか複数の `ENV` を使用します。

Docker イメージのビルド

Dockerfile を作成したら、以下のコマンドでイメージを作成できます：

```
docker build -t nomImage:optionnelTag .
```

- `build` はイメージを構築するためのコマンドです。
- `-t` はイメージに名前を付けるオプションです。必要であれば名前の後に「:」でタグを追加することもできます。
- 最後の「.」は、ビルド対象のプロジェクトディレクトリを示しています（ここでは「.」は現在のディレクトリを意味します）。