

- ❶ 도커파일에서 가장 먼저 할 일은 FROM 키워드를 사용해 기본 이미지를 지정하는 것이다. 여기서는 파이썬 이미지를 사용하며 여러 가지 버전을 사용할 수 있다. 웹 페이지⁴⁷에서 사용할 수 있는 버전을 확인할 수 있다.
- ❷ WORKDIR 키워드를 사용해 작업 디렉터리를 /app으로 설정한다. 작업 디렉터리는 이미지로 빌드될 프로젝트 구조를 정리할 때 도움이 된다.
- ❸ COPY 키워드를 사용해 requirements.txt 파일을 로컬 디렉터리에서 도커 컨테이너의 작업 디렉터리로 복사한다.
- ❹ RUN 명령을 사용해 pip 패키지를 업그레이드하고 requirements.txt 파일을 기반으로 의존 라이브러리를 설치한다.
- ❺ EXPOSE 명령을 사용해 로컬 네트워크에서 애플리케이션에 접속할 수 있는 포트 번호를 설정한다.
- ❻ 나머지 파일을 도커 컨테이너의 작업 디렉터리로 복사한다.
- ❼ CMD 명령을 사용해 애플리케이션을 실행한다.

도커파일의 각 명령은 개별 레이어로 빌드되며, 도커는 각 레이어에 캐시를 적용해서 빌드 시간을 줄이고 반복을 제거한다. 예를 들어 아주 중요한 명령으로 구성된 레이어에 변경 사항이 없다면 해당 레이어를 빌드하지 않고 이전에 빌드된 것을 사용한다. 즉, 이미지를 빌드할 때 캐시 시스템을 사용한다.

이미지를 빌드하기 전에 .dockerignore 파일을 만들자.

```
(venv)$ touch .dockerignore
```

.dockerignore 파일에 다음 내용을 추가한다.

```
venv
.env
.git
```



.dockerignore 파일

.dockerignore 파일에는 도커파일에 정의된 명령(COPY 등)을 실행할 때 제외할 파일과 폴더를 지정한다.

⁴⁷ https://hub.docker.com/_/python