

반면 `git checkout -b newbranch`는 현재 브랜치에서 `newbranch`라는 이름의 새 브랜치를 만든 다음 활성 브랜치로 설정한다. 더 자세한 내용은 Git 문서⁴를 참고하자.

Git의 기본 명령을 간단히 살펴봤다. 이제 `virtualenv`를 사용해 독립된 개발 환경을 구축해보자.

1.2 virtualenv를 사용한 개발 환경 구축

일반적으로 파이썬 애플리케이션은 가상 환경^{virtual environment}을 사용해 개발된다. 가상 환경에서 애플리케이션을 개발하면 특정 패키지를 시스템에 전역^{global}으로 설치하지 않아도 된다. 또한 서로 다른 애플리케이션을 동시에 개발할 때 발생할 수 있는 충돌을 피할 수 있다.

가상 환경은 독립된 환경이기 때문에 애플리케이션 의존성이 해당 환경 내에만 존재한다. 즉, 애플리케이션은 가상 환경 내에 있는 패키지에만 접근할 수 있다.

가상 환경 생성

파이썬 3를 설치하면 기본으로 표준 라이브러리의 `venv` 모듈이 함께 설치되는데 이 모듈이 가상 환경을 생성한다. 다음 명령을 사용하여 `todos`라는 폴더를 만들고 이 폴더 안에 가상 환경을 생성해보자.

```
$ mkdir todos && cd todos
$ python3 -m venv venv
```

윈도우에서는 다음 명령을 사용한다.⁵

```
c:\code\todos> mkdir todos && cd todos
c:\code\todos> python -m venv venv
```

⁴ [https://www.git-scm.com/doc](https://www.git-scm.com/doc(한글 매뉴얼은 https://wiki.kldp.org/Translations/html/Git-User-Manual/에서 확인할 수 있다))

⁵ 옮긴이 `python3`가 아닌 `python`을 실행해야 하는 경우도 있다. 보통 파이썬 2와 파이썬 3가 모두 설치되어 있으면 `python3`라고 버전을 명시해야 한다. 반대로 파이썬 2가 설치되어 있지 않고 파이썬 3만 있다면 `python`을 실행해도 `python3`가 실행된다.