

pyenv

pyenv를 사용하는 이유

디렉토리별로 python의 가상환경을 만들어 주어, 하나의 시스템 안에서 여러개 python 버전을 효율적으로 관리 할수 있습니다.

reference

<https://github.com/pyenv/pyenv-installer>

<https://github.com/yyuu/pyenv/wiki/Common-build-problems>

<https://github.com/kennethreitz/autoenv>

1. 우분투의 패키지 관리 툴인 apt-get 을 update

```
$ sudo apt-get update
$ sudo apt-get upgrade
```

2. 의존성 파일 install

```
$ sudo apt-get install -y make build-essential libssl-dev zlib1g-dev libbz2-dev \
  libreadline-dev libsqlite3-dev wget curl llvm libncurses5-dev libncursesw5-dev \
  xz-utils tk-dev
```

3. git & pyenv install

```
$ sudo apt-get install git
$ curl -L https://raw.githubusercontent.com/yyuu/pyenv-installer/master/bin/pyenv-installer | bash
```

4. .bash_profile 설정

(1) .bash_profile 파일 생성
/home/ubuntu\$ vi .bash_profile

(2) .bash_profile 파일에 아래 내용 추가
export PATH="/home/ubuntu/.pyenv/bin:\$PATH"
eval "\$(pyenv init -)"
eval "\$(pyenv virtualenv-init -)"

(3) 변경된 설정 적용
\$ source .bash_profile

5. Install python versions

(1) 두개의 파이썬 버전을 설치
\$ pyenv install 3.6.4
\$ pyenv install 2.7.12

* 전체 버전에 대한 정보 확인 (\$ pyenv install --list)

(2) pyenv에 설치된 버전과 현재 사용하고 있는 버전 확인
\$ pyenv versions

(3) pyenv에 설치된 버전으로 변경
\$ pyenv shell 3.6.4

(4) global 버전 설정하기
\$ pyenv global 3.6.4

6. virtualenv 설치하기 같은 버전을 다른 가상환경에서 쓰고 싶을 때 만들.

pyenv의 가상환경을 만들어줌
\$ sudo apt-get install pyenv-virtualenv
\$ sudo apt-get install python-virtualenv

7. 가상환경 만들기

(1) 해당 파이썬 버전으로 가상환경 만들어 주기
\$ pyenv virtualenv 3.6.4 python3
\$ pyenv virtualenv 2.7.12 python2

(2) 가상환경으로 파이썬 버전 변경 가능
\$ pyenv shell python2

8. 가상환경 삭제하기

\$ pyenv uninstall python2

9. autoenv 설치하기

<https://github.com/kennethreitz/autoenv>

\$ git clone git://github.com/kennethreitz/autoenv.git ~/.autoenv
\$ echo 'source ~/.autoenv/activate.sh' >> ~/.bash_profile
\$ source .bash_profile

10. autoenv 설정하기 자동으로 버전이 바뀌게 함.

아래와 같이 실행하면 python2로 python 버전이 바뀝니다.

\$ pyenv activate python2

기본 설정으로 바꾸고 싶으면 아래와 같이 실행하면 됩니다.

\$ pyenv deactivate

특정 디렉토리에 자동으로 적용하고 싶으면 .env 파일을 만들어 위의 명령을 넣어 주면 됩니다.

```
$ mkdir python2  
$ cd python2  
python2 $ vi .env
```

```
pyenv activate python2
```

```
$ mkdir python3  
$ cd python3
```

```
python3 $ vi .env
```

```
pyenv activate python3
```

```
$ vi .env
```

```
pyenv deactivate
```

위와 같이 작성을 하면 해당 디렉토리로 갈때 마다 해당 버전이 실행 됩니다.