1. 리눅스상에 파이썬 설치파일 가져오기

* Python 3.6.7.tgz
* Pscp 를 통해 설치파일을 보낸다

1. 파이썬 설치
2. 압축풀기
3. Configure 설정(C compiler 필수, GCC)
4. Err : Yum Install GCC - Missing Kernel-headers

-> <https://stackoverflow.com/questions/6513731/linux-yum-install-gcc-missing-kernel-headers>

4) yum downgrade glibc glibc-common 를 통해 최종 dependencies 문제를 해결한다.

5) yum install gcc -> local repo에 있는 gcc 설치완료 (v2.12-1.209)

- <http://wonwoo.ml/index.php/post/363> 경로설정 먼저하자

5-5) zlib, zlib-devel 설치 -> yum install zlib, yum install zlib-devel

6) ./configure -> make 설치

7) make -> 컴파일완료, make install까지

7-1) [zipimport.ZipImportError: can't decompress data; zlib not available] 경우,

yum -y install zlib-devel 한 뒤 다시 make install(setuptools, pip 설치)

8) ln -s Python설치폴더 /bin/python3 -> 바로가기 생성

1. 파이썬 가상환경 설정
2. Setuptools (v33)

* .~~deb파일을 통한 시도 ->Rpmforge 다운로드 후 설치(RPMs) ->yum install apt 실패~~
* P~~ython setup.py install~~
* python3 setup.py install

1. (보류)vim 설치(yum) -> 환경설정

|  |
| --- |
| **export PATH=$HOME/.local/bin:$PATH** |
| **if [ -f $HOME/.local/bin/virtualenvwrapper.sh ]; then**  **export WORKON\_HOME=$HOME/.virtualenvs**  **source $HOME/.local/bin/virtualenvwrapper.sh**  **fi** |

1. pip -> easy\_install pip-xxx.tar.gz ({all} error 발생)

3-1) get-pip.py 로 설치 시도(x)

3-2) pip3

1. pbr -> pip3 install pbr..
2. ~~virtualenv-clone~~
3. ~~virtualenv~~
4. ~~stevedore~~
5. ~~virtualenvwrapper~~
6. ~~pytz~~

* 파이썬 3.6 은 venv가 내장되어있다고 한다.

(<https://stackoverflow.com/questions/52119544/python-how-to-install-virtualenv-without-internet-connection>)

* venv는 python3에 기본 내장되어 있기 때문에 python3라면 virtualenv대신 venv를 사용하면 됨

python3 -m venv .venv(가상환경경로임) --prompt=myvenv

* 인터넷없이 pip를 아래와 같이 설치할 수 있다

python3 -m ensurepip

* 아래는 버전체크없이 pip를 통해 설치하는 예시

python3 -m pip install --no-index --disable-pip-version-check ./mydist.tar.gz

1. 가상환경 activate하기

* Source 가상환경경로/bin/activate

1. Deactivate

* deactivate 가상환경이름

1. 관련 라이브러리 오프라인 설치
2. Pandas
3. Numpy
4. GLIBC 2.12 -> 2.14 로 업데이트

<https://stackoverflow.com/questions/35616650/how-to-upgrade-glibc-from-version-2-12-to-2-14-on-centos>

1. Tensorflow

3-1) 관련 패키지 다수..

**Centos 7 상에서 설치환경 구축**

1. Anaconda3 설치

* Bash Anaconda…sh
* 내용을 확인하며 다음단계로 넘어간다, path설정하는 부분은 yes로 수동변경해줘야함
* 설치완료후 터미널 재시작, ‘conda’ 입력시 알아들으면 설치완료

1. 가상환경 만들기

* Conda create -n {가상환경명} python={파이썬버전, 3.6.7입력 후 진행}
* Conda activate {가상환경명}

1. 파이썬 패키지설치하기

* Pandas
* Tensorflow
* Matplotlib
* Sklearn
* Seaborn

1. 외부접속 설정

* Jupyter notebook –generate-config

-> home/~username~/“**.**jupyter” 안에 jupyter\_notebook\_config.py 생성

* Ipython or jupyter notebook 상에서 아래 커맨드 입력
* from notebook.auth import passwd
* passwd()
* 원하는 패스워드 입력
* Return 하는 암호화된 문자열을 복사해 놓는다(중요), 파이썬 종료
* Vi /.jupyter/jupyter\_notebook\_config.py
* #c.NotebookApp.ip = ‘localhost’ -> ‘\*’ 으로 바꿔준다(주석도 푼다)
* #c.NotebookApp.passwd = u’{암호화된 문자열}’ 로 변경(주석도 푼다)
* c.NotebookApp.open\_browser = False 로 변경
* c.NotebookApp.port = {희망포트번호}

1. 포트 개방하기

* Firewall-층 –zone=public –permanent –add-port={희망포트번호}/tcp
* Firewall-층 --reload

1. (root가 아닌경우) 유저 권한주기

Chown user:user /home/user/.local/share/jupyter

1. 주피터노트북 실행

* (가상환경 접속상에서) jupyter notebook