

# Jupyter notebook server

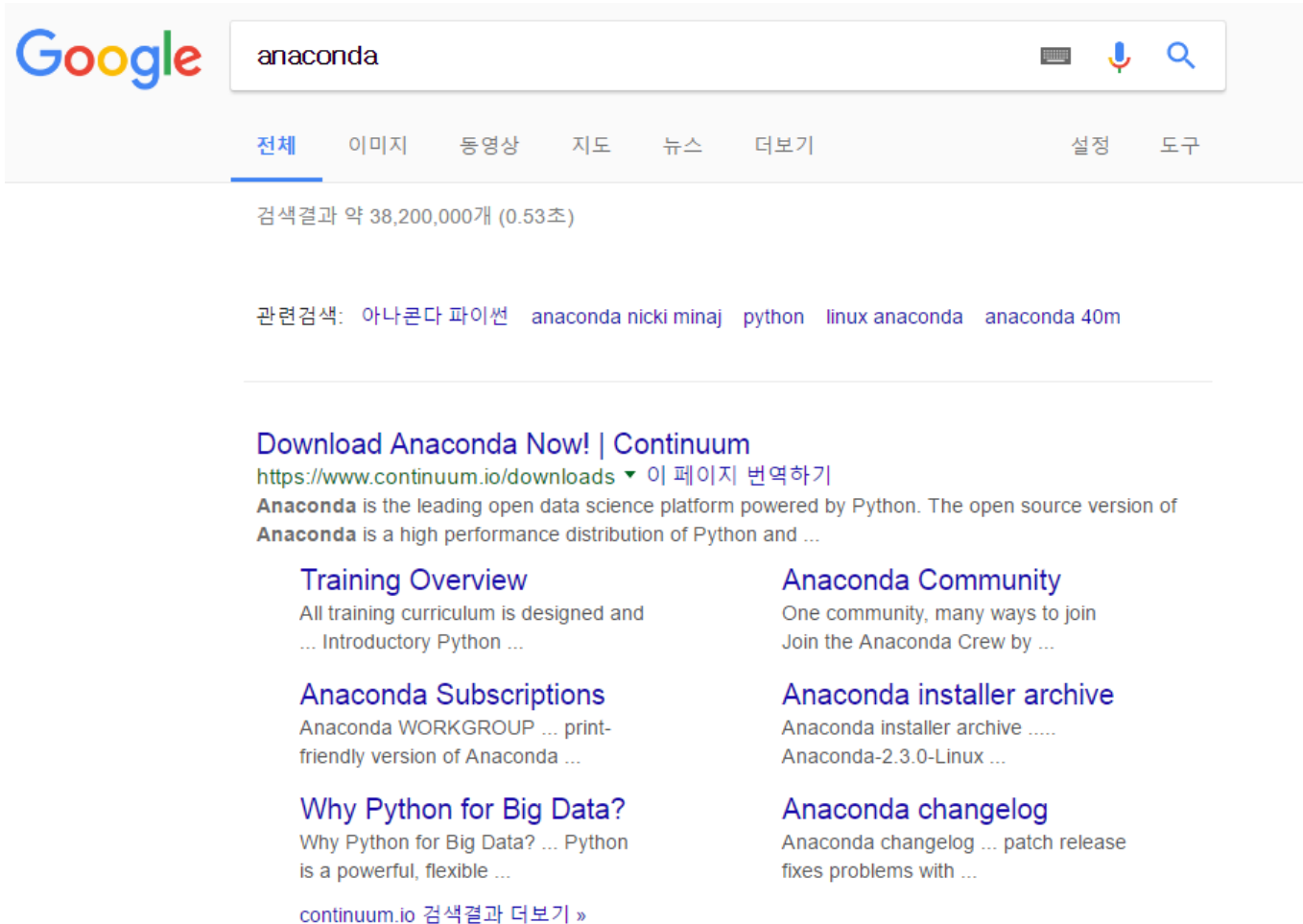
김현중 (soy.lovit@gmail.com)

# Anaconda

- Anaconda는 python 및 주로 사용되는 패키지들의 버전테스트까지 완료하여, 한 번의 설치로 주요 패키지들을 모두 이용할 수 있도록 하는 일종의 번들킷
  - Gensim, tensorflow, keras 와 같은 자연어처리나 딥러닝에 이용되는 라이브러리도 호환이 잘 됨
  - Scikit-learn, numpy, scipy, pandas와 같은 데이터분석에 필요한 라이브러리들은 자체가 깔려있음

# Anaconda

- 설치는 [www.continuum.io](http://www.continuum.io)에서 할 수 있음



The screenshot shows a Google search interface with the query 'anaconda'. The search results indicate approximately 38,200,000 results found in 0.53 seconds. Below the search bar, there are tabs for '전체' (All), '이미지' (Images), '동영상' (Videos), '지도' (Maps), '뉴스' (News), and '더보기' (More). To the right of these tabs are links for '설정' (Settings) and '도구' (Tools). The search results section includes a '관련검색' (Related searches) list with terms like '아나콘다 파이썬', 'anaconda nicki minaj', 'python', 'linux anaconda', and 'anaconda 40m'. The main search result is titled 'Download Anaconda Now! | Continuum' with a URL 'https://www.continuum.io/downloads'. Below the title, there is a brief description of Anaconda as an open data science platform powered by Python. The result also includes several sub-links: 'Training Overview', 'Anaconda Subscriptions', 'Why Python for Big Data?', 'Anaconda Community', 'Anaconda installer archive', and 'Anaconda changelog'. At the bottom of the search results, there is a link to 'continuum.io 검색결과 더보기 »'.

Google anaconda

전체 이미지 동영상 지도 뉴스 더보기 설정 도구

검색결과 약 38,200,000개 (0.53초)

관련검색: 아나콘다 파이썬 anaconda nicki minaj python linux anaconda anaconda 40m

**Download Anaconda Now! | Continuum**  
<https://www.continuum.io/downloads> 이 페이지 번역하기

Anaconda is the leading open data science platform powered by Python. The open source version of Anaconda is a high performance distribution of Python and ...

**Training Overview**  
All training curriculum is designed and ... Introductory Python ...

**Anaconda Subscriptions**  
Anaconda WORKGROUP ... print-friendly version of Anaconda ...

**Why Python for Big Data?**  
Why Python for Big Data? ... Python is a powerful, flexible ...

**Anaconda Community**  
One community, many ways to join Join the Anaconda Crew by ...

**Anaconda installer archive**  
Anaconda installer archive ..... Anaconda-2.3.0-Linux ...

**Anaconda changelog**  
Anaconda changelog ... patch release fixes problems with ...

[continuum.io 검색결과 더보기 »](#)

# Anaconda

- Python은 버전이 2.x와 3.x가 모두 사용되지만, 한국어 분석에는 인코딩 문제가 많이 해결된 3.x가 더 좋으며, 3.x로 사용자들이 계속 옮겨오고 있음
- OS는 자동으로 선택되지만, 필요에 따라 OS 별로 다운 받을 수 있음

The screenshot shows the Anaconda 4.3.1 download page for Windows. At the top, there are three tabs: 'Download for Windows' (selected), 'Download for macOS', and 'Download for Linux'. The main content area is titled 'Anaconda 4.3.1 For Windows'. It includes a paragraph stating that Anaconda is BSD licensed and provides permission to use it commercially and for redistribution. Below this is a link to the 'Changelog'. A list of three steps guides the user through downloading the installer, verifying data integrity with MD5 or SHA-256 (with a 'More info' link), and double-clicking the .exe file to install. At the bottom, it mentions 'Behind a firewall? Use these zipped Windows installers'. On the right side, there are two sections for Python versions. The first section is for 'Python 3.6 version' and offers two download options: a green '64-BIT INSTALLER (422M)' button and a blue '32-BIT INSTALLER (348M)' link. The second section is for 'Python 2.7 version' and offers two download options: a blue '64-BIT INSTALLER (414M)' button and a blue '32-BIT INSTALLER (339M)' link.

Download for Windows   Download for macOS   Download for Linux

## Anaconda 4.3.1

### For Windows

Anaconda is BSD licensed which gives you permission to use Anaconda commercially and for redistribution.

[Changelog](#)

1. Download the installer
2. Optional: Verify data integrity with [MD5](#) or [SHA-256](#) [More info](#)
3. Double-click the **.exe** file to install Anaconda and follow the instructions on the screen

Behind a firewall? Use these [zipped Windows installers](#)

Python 3.6 version

**64-BIT INSTALLER (422M)**

[32-BIT INSTALLER \(348M\)](#)

Python 2.7 version

**64-BIT INSTALLER (414M)**

[32-BIT INSTALLER \(339M\)](#)

# Anaconda

- 각자 컴퓨터의 비트는 제어판 >> 시스템 및 보안 >> 시스템 에서 확인 가능  
Bit를 맞추지 않고 설치하면 파이썬이 제대로 작동하지 않음



# Anaconda Virtual Environment

- Anaconda의 기본 셋팅을 해치지 않으며, 각자 작업에 필요한 가상의 파이썬 환경을 생성할 수 있음
- Cmd 창에서 다음을 입력

```
id:@jupyter_server_home>conda create --name YOUR_ENV_NAME python=3.6 anaconda
```

- 아나콘다 공식홈페이지에서는 “conda create -name YOUR\_ENV\_NAME”을 입력하도록 하지만, 위처럼 python=3.6으로 파이썬 버전을 지정하고 그 뒤에 anaconda라고 입력하면 anaconda 설치시 기본으로 설치되는 모든 패키지를 다 설치함

# Anaconda Virtual Environment

- “conda info --envs”를 입력하면 설치된 모든 가상 환경 리스트가 출력

```
d:\jupyter_server_home>conda info --envs
Using Anaconda Cloud api site https://api.anaconda.org
# conda environments:
#
gensim_py3          C:\Anaconda2\envs\gensim_py3
keras_py3           C:\Anaconda2\envs\keras_py3
py3                 C:\Anaconda2\envs\py3
python3            C:\Anaconda2\envs\python3
root                * C:\Anaconda2
```

- 현재 사용중인 가상환경을 \*로 표시함

# Anaconda Virtual Environment

- “activate YOUR\_ENV\_NAME”를 입력하면 해당 환경의 이름이 [ ]사이에 나옴
  - 해당 환경을 이용하고 있다는 의미

```
d:\jupyter_server_home>activate python3
Deactivating environment "C:\Anaconda2"...
Activating environment "C:\Anaconda2\envs\python3"...

[python3] d:\jupyter_server_home>
```



```
[python3] d:\jupyter_server_home>conda info --envs
Using Anaconda Cloud api site https://api.anaconda.org
# conda environments:
#
gensim_py3          C:\Anaconda2\envs\gensim_py3
keras_py3           C:\Anaconda2\envs\keras_py3
py3                 C:\Anaconda2\envs\py3
python3             * C:\Anaconda2\envs\python3
root                C:\Anaconda2
```



# Jupyter notebook server

- Anaconda에는 Jupyter notebook server 가 기본으로 설치되어 있다.

```
>> jupyter notebook --generate-config
```

- 위 명령어를 입력하면 C:\Users\[해당사용자폴더]\.jupyter에 “jupyter\_notebook\_config.py” 파일이 생성

# Jupyter notebook server

- 비밀번호를 설정하기 위해서 cmd에서 ipython 실행.

```
[python3] d:\jupyter_server_home>ipython
Python 3.5.2 |Anaconda 4.1.1 (64-bit)| (default, Jul  5 2016, 11:41:13) [MSC v.1900 64 bit
Type "copyright", "credits" or "license" for more information.

IPython 4.2.0 -- An enhanced Interactive Python.
?                -> Introduction and overview of IPython's features.
%quickref        -> Quick reference.
help             -> Python's own help system.
object?         -> Details about 'object', use 'object??' for extra details.

In [1]:
```

- 아래 코드 실행. Out으로 나오는 부분을 한 곳에 적어두자.

```
In [1]: from notebook.auth import passwd
In [2]: passwd()
Enter password:
Verify password:
Out[2]: 'sha1:f24baff49ac5:863dd2ae747212ede58125302d227f0ca7b12bb3'
```

# Jupyter notebook server

- `\.jupyter\jupyter_notebook_config.py` 파일에서 다음의 항목을 수정하자

`c.NotebookApp.ip = 'YOUR_IP'`

`c.NotebookApp.open_browser = False`

`c.NotebookApp.password = u'out[2]에 적혀있던 그 코드'`

`c.NotebookApp.port = 8888`

`c.NotebookApp.notebook_dir = u'주피터를 시작하고 싶은 폴더 주소'`

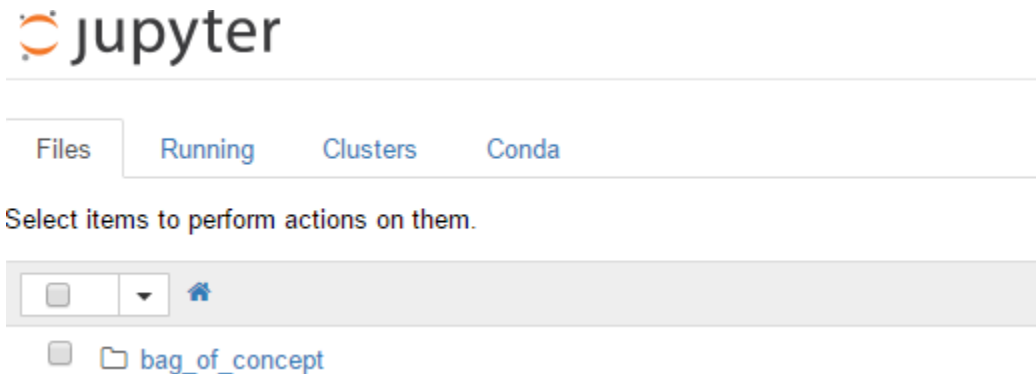
# Jupyter notebook server

- Chrome을 켜고 주소창에 아래처럼 입력한다.

YOUR\_IP:8888

Ex) 147.46.255.255:8888

- `c.NotebookApp.notebook_dir` 폴더가 나타남을 볼 수 있다.



# Jupyter notebook server

- 본인의 컴퓨터에서 접속이 되었다면, 학교 외의 무선 네트워크 (테더링, 에그 등)으로 노트북으로도 동일한 주소로 접속해보자.
- 동일하게 접속이 되었다면 8888 port가 이용가능 하다는 의미이며, 데스크탑이나 교내 인터넷으로는 접속이 되었는데, 외부 네트워크에서 접속이 안된다면 8888 port가 보안으로 막혀있다는 의미
  - 이 때에는 중앙전산원에 포트 사용 신청을 해야 한다 (각 연구실의 IP 관리자들에게 문의)
  - 기본으로 학교에서는 8888은 막혀있지 않다.

# Jupyter notebook server

- Jupyter notebook을 시작하는 folder 아래에 다음과 같은 구조로 폴더를 만들기를 추천한다

```
/PROJECT_NAME/data/raw  
/PROJECT_NAME/playground  
/PROJECT_NAME/stage
```

- ../data/raw/ 폴더 아래에 데이터를 저장하고, 그 폴더에 있는 데이터는 수정하지 않기를 권한다. 모든 수정된 데이터는 다른 이름 (예: PROJECT\_NAME/data/processed) 폴더에 넣어서 단계별로 보관하길 바란다

# KoNLPy (for Windows)

김현중 (soy.lovit@gmail.com)

# Install JDK, JPyep1 and KoNLPy

- **KoNLPy**

- 공식 홈페이지: <http://konlpy.org/ko/v0.4.4/>
- 각종 한국어 형태소 분석기를 Python에서 사용할 수 있게 만들어주는 패키지
- 많은 한국어 형태소 분석기가 Java 기반으로 되어 있기 때문에, Python 환경에서 Java 프로그램을 실행하게 해주는 “JPyep1”을 설치해야 함.

- **JPyep1**

- Python환경에서 Java 프로그램을 실행해주는 패키지
- 미리 Java Development Kit (JDK)를 설치해야만 사용 가능합니다!

## Dependencies







# Install JDK, JPytype1 and KoNLPy

- KoNLPy의 공식 설치 문서를 따라해주시기 바랍니다.
  - <http://konlpy.org/ko/v0.4.4/install/>
    - 공식 문서에는 JDK 1.7을 설치하라고 되어 있음.
    - (제 경험) 저는 JDK 1.8을 사용해야 하는 환경을 사용 중이어서 JDK 1.8을 설치하였는데 사용하는 데에 문제는 (아직까지) 없었습니다.
    - 밑에 운영체제 별로 설치할 때의 유의점을 적어 두었습니다.

# Install JDK





전체

동영상

이미지

뉴스

도서

더보기

설정

도구

검색결과 약 14,000,000개 (0.26초)

## 구글에서 “jdk” 검색 후, 아래 페이지(“JAVA SE Development Kit 8”)로 이동

관련검색: [jdk 설치](#) [jdk 다운로드](#) [eclipse](#) [mysql](#) [notepad++](#)

---

### Java SE Development Kit 8 - Downloads - Oracle

[www.oracle.com](#) › [Java](#) › [Java SE](#) ▼ [이 페이지 번역하기](#)

Download **JDK 8**, a development environment for building applications and components using the Java programming language.

이 페이지를 여러 번 방문했습니다. 최근 방문 날짜: 16. 12. 15

### Java SE - Downloads | Oracle Technology Network | Oracle

[www.oracle.com](#) › [Java](#) › [Java SE](#) ▼ [이 페이지 번역하기](#)

Java SE downloads including: **Java Development Kit (JDK)**, **Server Java Runtime Environment (Server JRE)**, and **Java Runtime Environment (JRE)**.

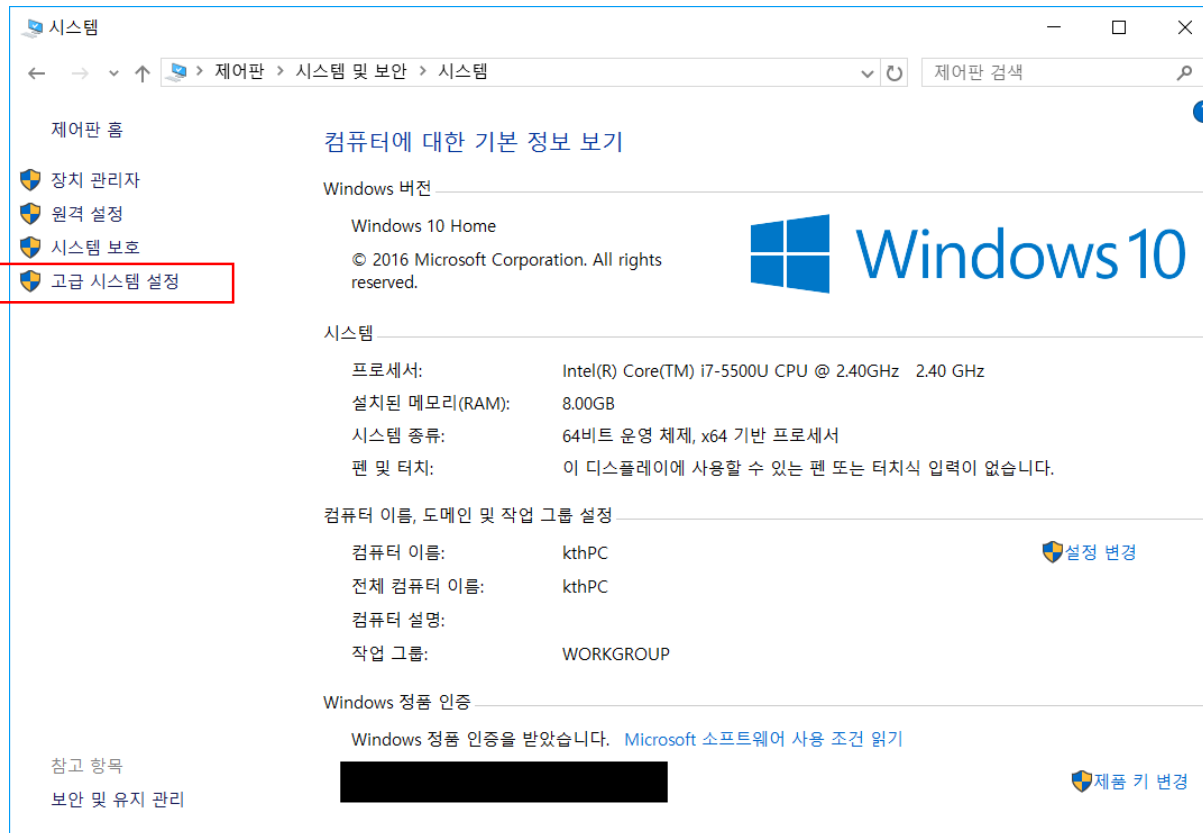
# Install JDK

- 위에서 “Accept License Agreement”를 클릭한 후, 자신의 OS와 bit 수를 고려하여 알맞은 JDK를 다운받아 설치
- JDK 8u111과 8u112 중 어떤 것도 상관없음

| Java SE Development Kit 8u111   |           |   |
|---|-----------|---|
| You must accept the <a href="#">Oracle Binary Code License Agreement for Java SE</a> to download this software. |           |   |
| <input type="radio"/> Accept License Agreement <input checked="" type="radio"/> Decline License Agreement       |           |   |
| Product / File Description  | File Size | Download  |
| Linux ARM 32 Hard Float ABI   | 77.78 MB  | <a href="#">jdk-8u111-linux-arm32-vfp-hflt.tar.gz</a> |
| Linux ARM 64 Hard Float ABI   | 74.73 MB  | <a href="#">jdk-8u111-linux-arm64-vfp-hflt.tar.gz</a> |
| Linux x86   | 160.35 MB | <a href="#">jdk-8u111-linux-i586.rpm</a>              |
| Linux x86   | 175.04 MB | <a href="#">jdk-8u111-linux-i586.tar.gz</a>           |
| Linux x64   | 158.35 MB | <a href="#">jdk-8u111-linux-x64.rpm</a>               |
| Linux x64   | 173.04 MB | <a href="#">jdk-8u111-linux-x64.tar.gz</a>            |
| Mac OS X  | 227.39 MB | <a href="#">jdk-8u111-macosx-x64.dmg</a>              |
| Solaris SPARC 64-bit  | 131.92 MB | <a href="#">jdk-8u111-solaris-sparcv9.tar.Z</a>       |
| Solaris SPARC 64-bit  | 93.02 MB  | <a href="#">jdk-8u111-solaris-sparcv9.tar.gz</a>      |
| Solaris x64   | 140.38 MB | <a href="#">jdk-8u111-solaris-x64.tar.Z</a>           |
| Solaris x64   | 96.82 MB  | <a href="#">jdk-8u111-solaris-x64.tar.gz</a>          |
| Windows x86   | 189.22 MB | <a href="#">jdk-8u111-windows-i586.exe</a>            |
| Windows x64   | 194.64 MB | <a href="#">jdk-8u111-windows-x64.exe</a>             |

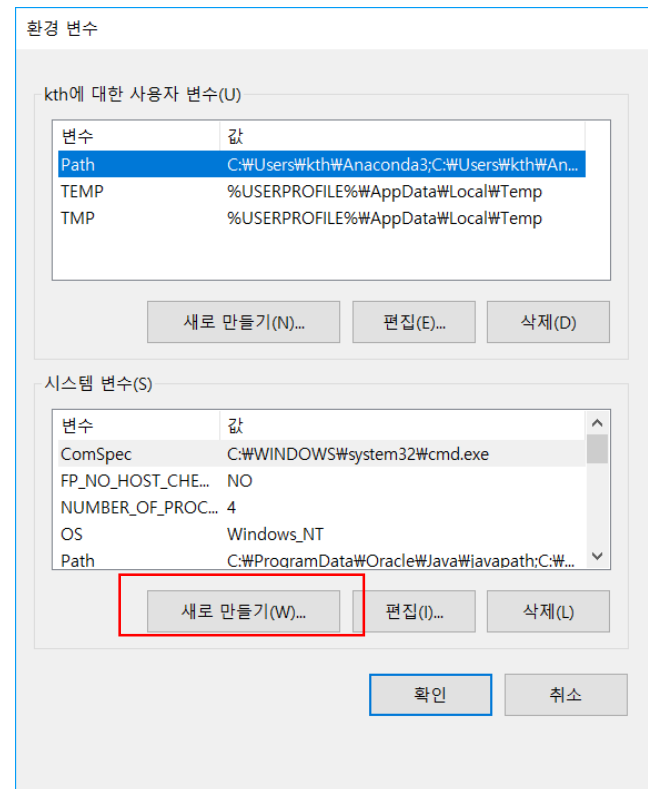
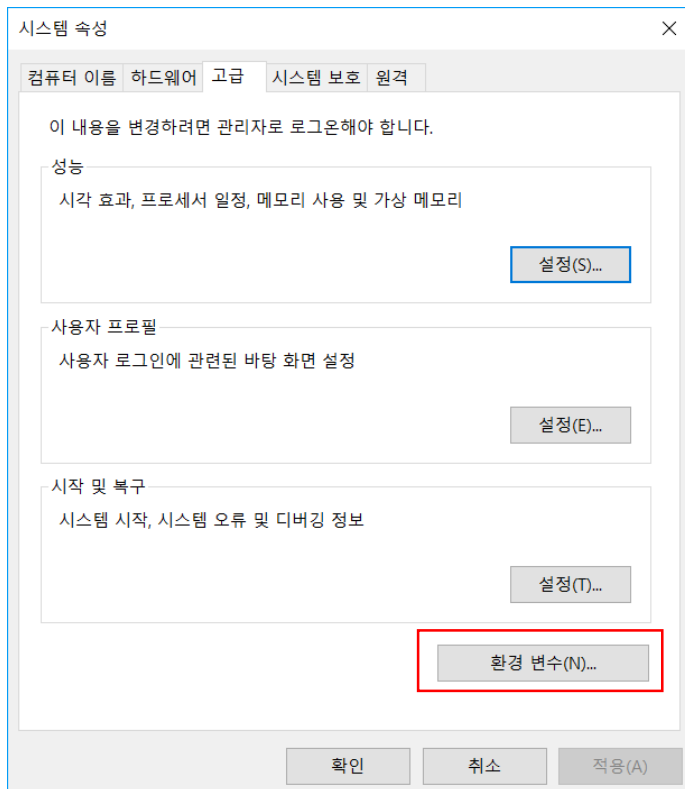
# Install JDK

- 주의사항: Windows를 사용 중인 경우,
  - JAVA\_HOME 환경 변수 만드는 방법 (<http://prolite.tistory.com/175> 참고)
    - 제어판 > 시스템 및 보안 > 시스템
    - 좌측 메뉴에서 “고급 시스템 설정” 클릭



# Install JDK

- 주의사항: Windows를 사용 중인 경우,
  - JAVA\_HOME 환경 변수 만드는 방법
    - “환경 변수” 클릭
    - [시스템 변수] 탭에서 “새로 만들기” 클릭



# Install JDK

- 주의사항: Windows를 사용 중인 경우,

- JAVA\_HOME 환경 변수 만드는 방법

- [변수 이름]에 **JAVA\_HOME** 입력

- [변수 값]에 **JDK 설치 경로**를 입력

- 각자 설치한 JDK의 버전에 따라 다를 수 있으므로 꼭 확인하시고 자신의 컴퓨터 상황에 맞게

입력해주세요 (보통 C:\Program Files\Java\jdk1.x.x.xxx 라는 디렉토리에 있음)

새 시스템 변수

변수 이름(N): JAVA\_HOME

변수 값(V): C:\Program Files\Java\jdk1.8.0\_102

디렉터리 찾아보기(D)...    파일 찾아보기(F)...    **확인**    취소

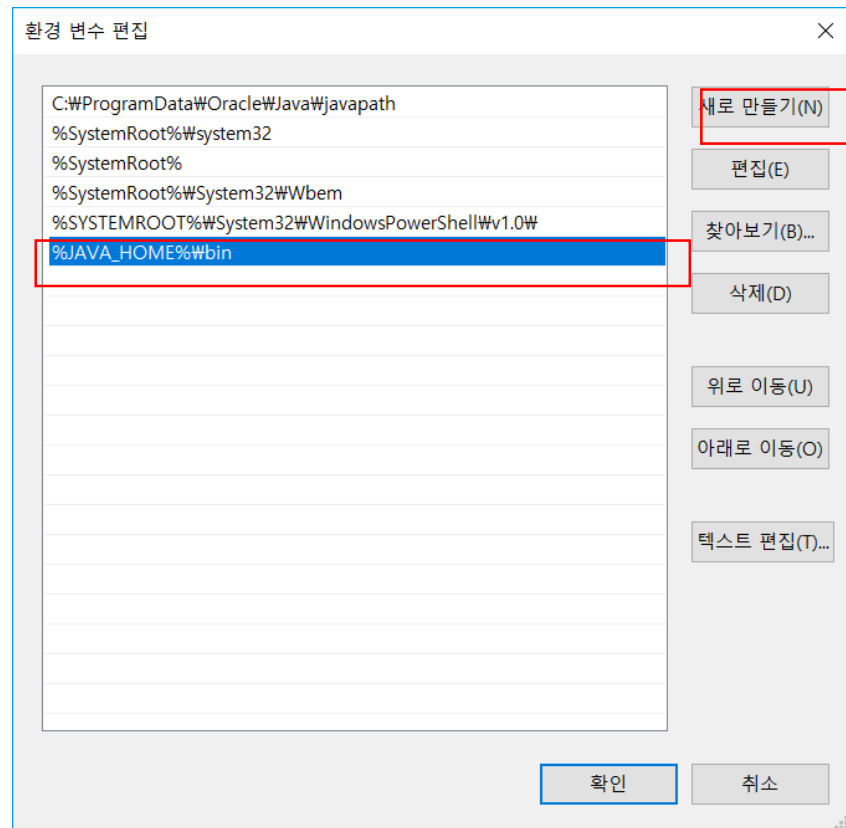
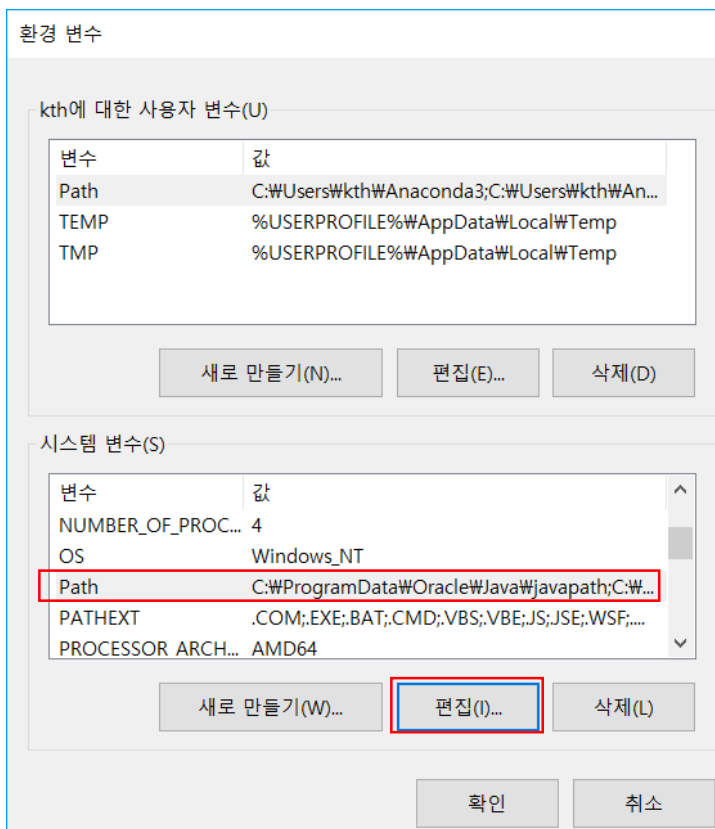
# Install JDK

- 주의사항: Windows를 사용 중인 경우,

- JAVA\_HOME 환경 변수 만드는 방법

- 시스템 변수 목록에 Path 를 찾아 선택 후 “편집”을 누릅니다.

- (For Windows 10) “새로 만들기”를 누른 후, **%JAVA\_HOME%\bin**를 추가

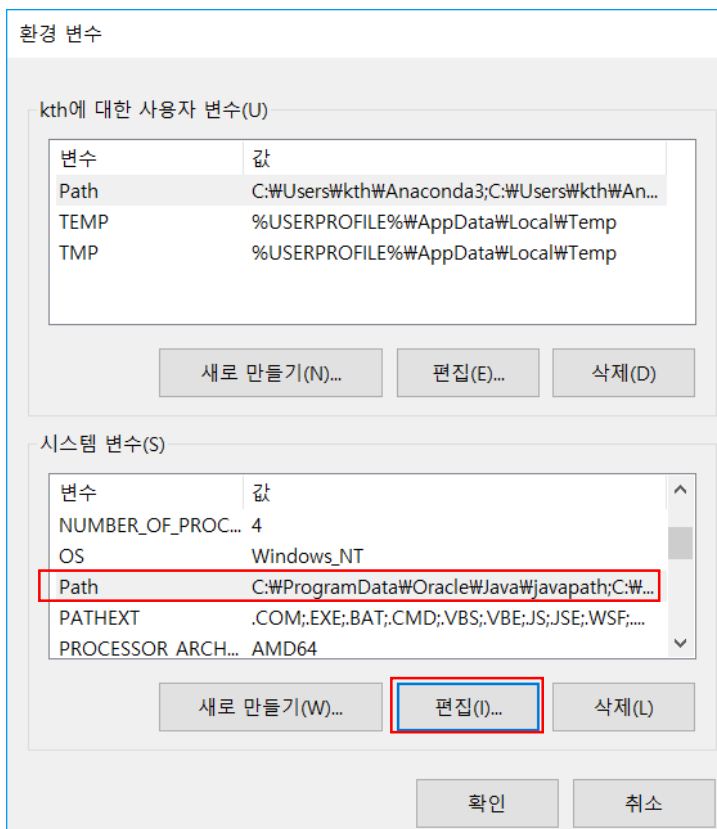


# Install JDK

- 주의사항: Windows를 사용 중인 경우,

- JAVA\_HOME 환경 변수 만드는 방법

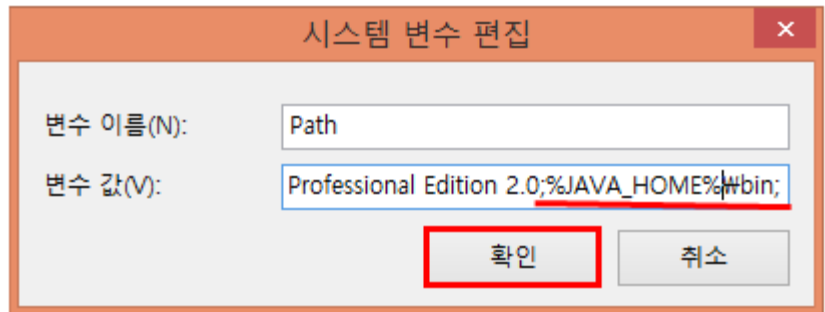
- (For Windows 7, 8) “새로 만들기”를 누른 후, 경로 끝에 `;%JAVA_HOME%\bin;`를 추가



16) 변수 값 `;%JAVA_HOME%\bin;`

여기서 변수 값은 맨 끝쪽에 씁니다.

반드시 `;` 표시가 양옆에 있어야 합니다. 확인을 누릅니다.

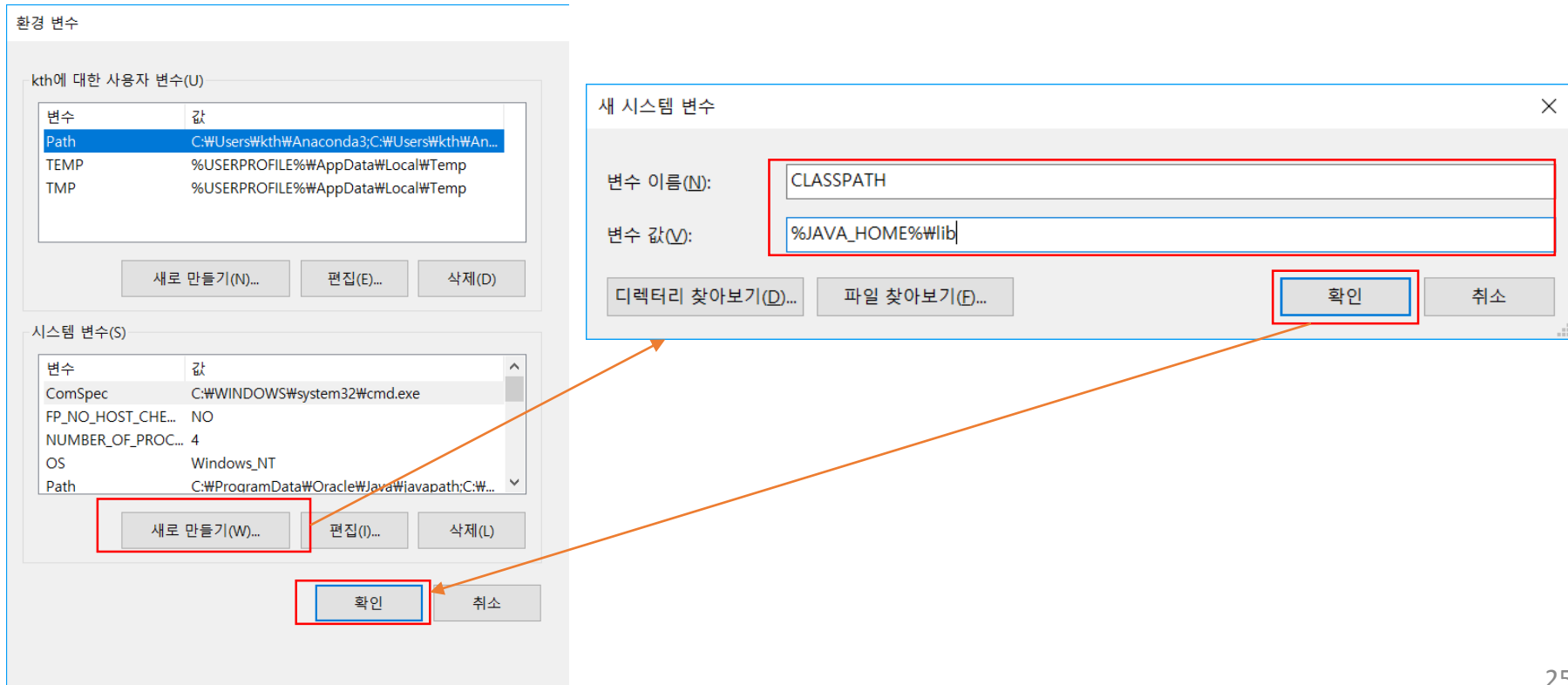


<http://prolite.tistory.com/175> 참고



# Install JDK, JPyte1 and KoNLPy

- 주의사항: Windows를 사용 중인 경우,
  - JAVA\_HOME 환경 변수 만드는 방법
    - 다시 “새로 만들기” 클릭
    - 변수 이름에 **CLASSPATH**, 변수 값으로 **%JAVA\_HOME%\lib** 입력 후 “확인” 누르기
    - [환경 변수] 창에 대해 “확인” 누르기



# Install JDK, JPyte1 and KoNLPy

- 주의사항: Windows를 사용 중인 경우,

- JAVA\_HOME 환경 변수 만드는 방법

- 명령 프롬프트를 실행하여 **javac** 실행 → 잘 실행되는지 확인

- 명령 프롬프트에서 **where javac** 입력 → 아까 설치한 jdk 디렉토리 안의 파일인지 확인

```
명령 프롬프트
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\Wkth>javac
Usage: javac <options> <source files>
where possible options include:
  -g               Generate all debugging info
  -g:none          Generate no debugging info
  -g:{lines,vars,source}  Generate only some debugging info
  -nowarn          Generate no warnings
  -verbose         Output messages about what the compiler is doing
  -deprecation     Output source locations where deprecated API is used
  -classpath <path>  Specify where to find user class files and
                    Specify where to find user class files and
  -cp <path>        Specify where to find input source files
  -sourcepath <path> Specify where to find input source files
  -bootclasspath <path>  Override location of bootstrap class files
  -extdirs <dirs>      Override location of installed extensions
  -endorseddirs <dirs>  Override location of endorsed standards path
  -proc:{none,only}  Control whether annotation processing and/or
                    -processor <class1>[,<class2>,<class3>...] Names of the annotation processors
  -processorpath <path> Specify where to find annotation processors
  -parameters      Generate metadata for reflection on method parameters
  -d <directory>    Specify where to place generated class files
  -s <directory>    Specify where to place generated source files
  -h <directory>    Specify where to place generated native header files
  -implicit:{none,class} Specify whether or not to generate class files
  -encoding <encoding> Specify character encoding used by source files
  -source <release>  Provide source compatibility with specified release
  -target <release>  Generate class files for specific VM version
```

```
명령 프롬프트
C:\Users\Wkth>where javac
C:\Program Files\Java\jdk1.8.0_102\bin\javac.exe

C:\Users\Wkth>
```

# Install JPyype1

- (for Windows) Download JPyype1 and install

- <http://www.lfd.uci.edu/~gohlke/pythonlibs/#jpype> 에 들어가서 Python 버전과 사용 중인 bit 수에 맞는 JPyype1에 대한 whl 파일을 다운로드
  - 예) Python 3.5 64bit를 사용하는 경우, 빨간색 네모의 파일을 다운로드
- 명령 프롬프트에서 파일을 다운로드한 폴더에 받은 패키지 설치
  - **activate tm** (앞서 만든 conda 환경 활성화)
  - **pip install JPyype1-0.6.1-xxxxxx-xxxxx.whl** (자신이 필요한 버전에 맞게)

JPyype allows full access to Java class libraries.

[JPyype1-0.6.1-cp27-none-win32.whl](#)

[JPyype1-0.6.1-cp27-none-win amd64.whl](#)

[JPyype1-0.6.1-cp34-none-win32.whl](#)

[JPyype1-0.6.1-cp34-none-win amd64.whl](#)

[JPyype1-0.6.1-cp35-none-win32.whl](#)

[JPyype1-0.6.1-cp35-none-win amd64.whl](#)

# [참고] Windows에서 Python 환경을 구성할 때,

- **Unofficial Windows Binaries for Python Extension Packages**

- URL: <http://www.lfd.uci.edu/~gohlke/pythonlibs>
- 주요 Python 패키지에 대해 윈도우 32bit와 64bit에 대한 지원을 해주는 repository
- Python 커뮤니티의 개발자들이 Windows에 그다지 친절하지 않습니다...
- 만약 특정 Python package를 설치할 때, Windows에서 잘 안 되는 문제가 있다면 이 사이트에서 whl파일을 다운로드 받은 후 pip로 설치해주시기 바랍니다.

# Install KoNLPy and other packages

- 명령 프롬프트에서 다음을 입력하여 KoNLPy를 설치
  - `pip install konlpy`
- 그 외에 더 설치할 패키지들 (wordcloud, Pillow, gensim)
  - `pip install wordcloud`
    - 윈도우의 경우, <http://www.lfd.uci.edu/~gohlke/pythonlibs/#wordcloud> 에서 whl 파일을 다운로드
  - `conda install Pillow` / `pip install Pillow`
  - `pip install gensim`