

한글 자연어처리(NL)을 위한 Konlpy 라이브러리 설치

1.

conda prompt로 nltk설치

<https://cafe.daum.net/bigmega/pJVC/7784>

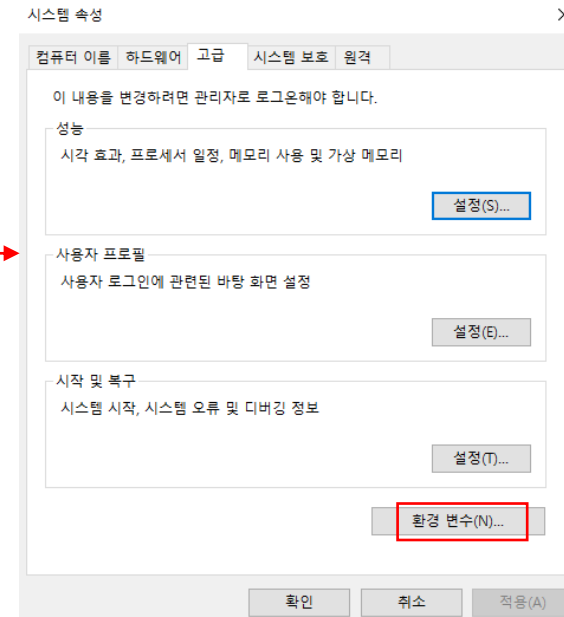
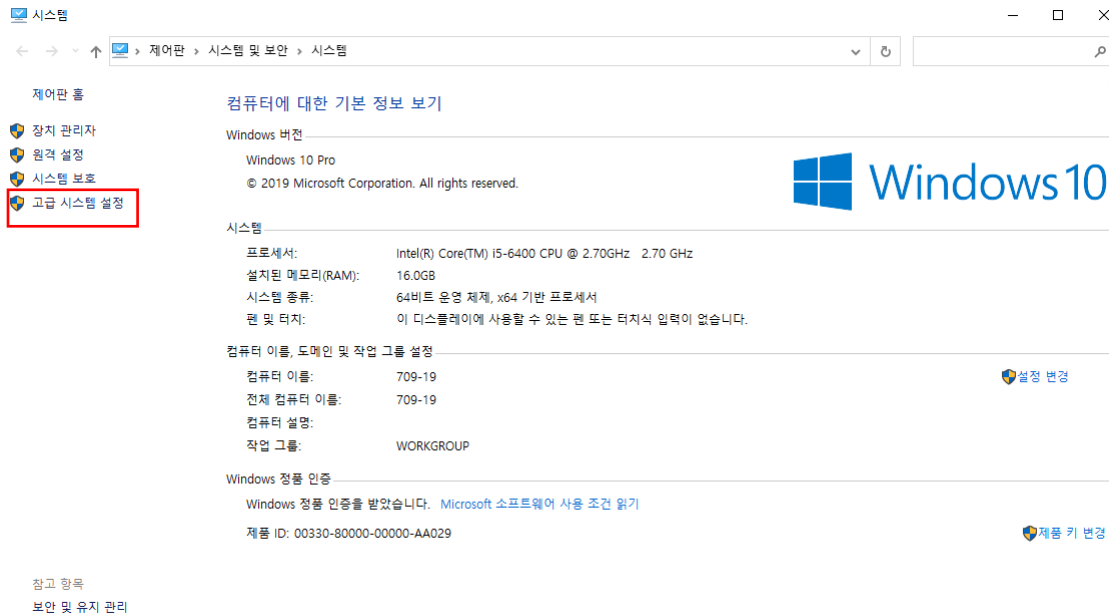


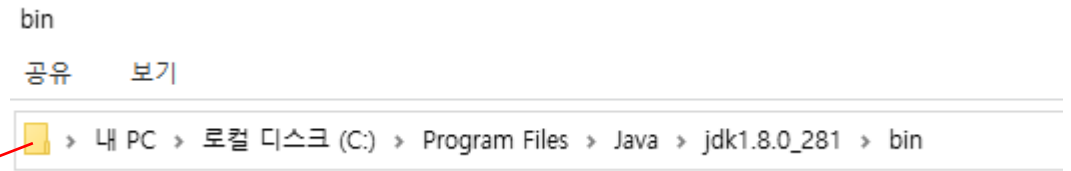
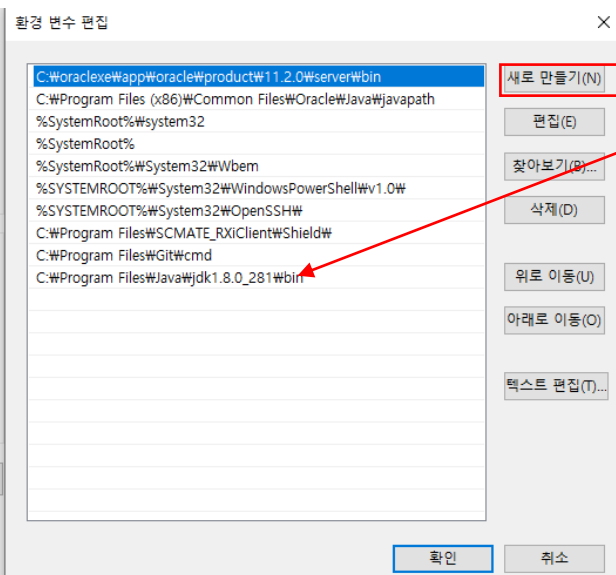
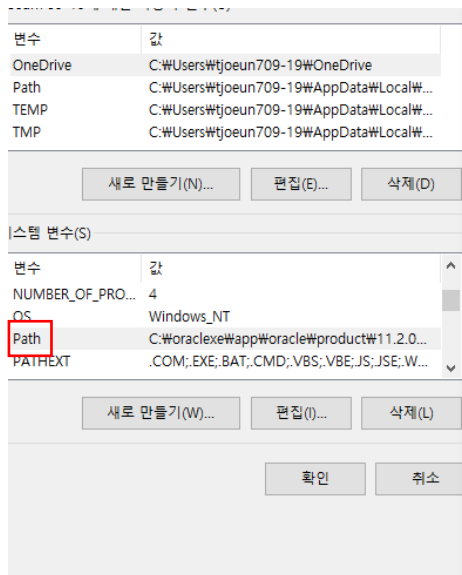
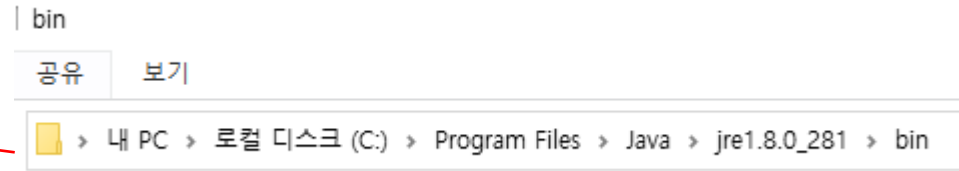
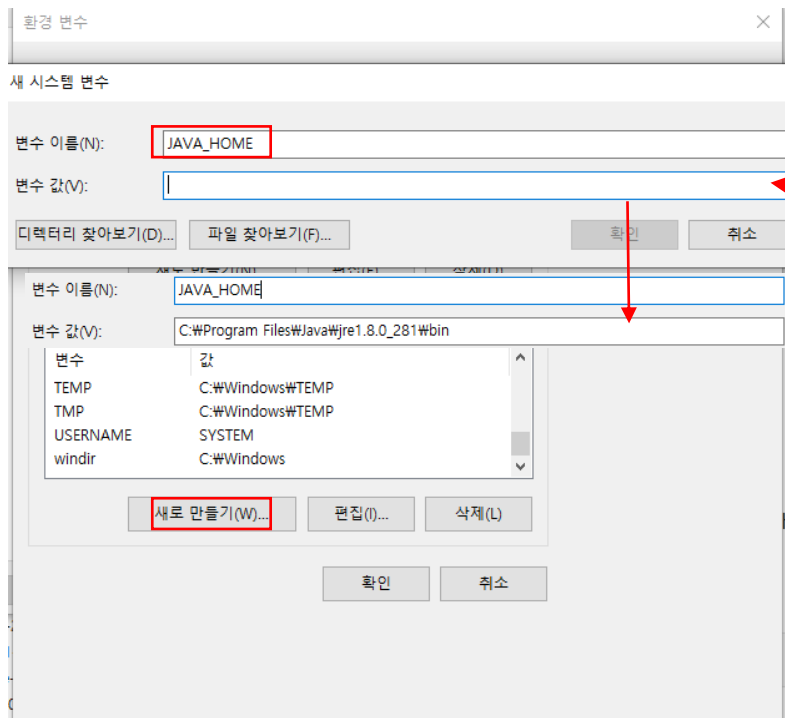
conda로 환경설정 후, 실행하는 경우 + nltk 설치

시스템변수에 자바 경로와 경로 2개 잡아줌. nltk import시 에러나면 numpy를 지우고, 다시 numpy설치

cafe.daum.net

2. 환경변수설정





3. <https://support.microsoft.com/en-us/topic/the-latest-supported-visual-c-downloads-2647da03-1eea-4433-9aff-95f26a218cc0>

<https://support.microsoft.com/en-us/topic/the-latest-supported-visual-c-downloads-2647da03-1eea-4433-9aff-95f26a218cc0>



The latest supported Visual C++ downloads

Notice Some of the downloads that are mentioned in this article are currently available on My.VisualStudio.com. Make sure to log in by using a Visual Studio Subscription...

support.microsoft.com

The latest supported Visual C++ x



l-visual-c-downloads-2647da03-1eea-4433-9aff-95f26a218cc0

Visual Studio 2015, 2017 and 2019

Download the [Microsoft Visual C++ Redistributable for Visual Studio 2015, 2017 and 2019](#). The following updates are the latest supported Visual C++ redistributable packages for Visual Studio 2015, 2017 and 2019. Included is a baseline version of the Universal C Runtime see [MSDN](#) for details.

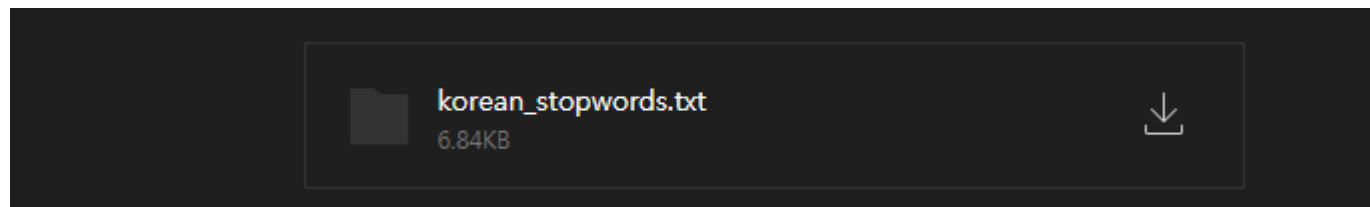
- x86: [vc_redist.x86.exe](#)
- x64: [vc_redist.x64.exe](#)
- ARM64: [vc_redist.arm64.exe](#)

Note Visual C++ 2015, 2017 and 2019 all share the same redistributable files.

For example, installing the Visual C++ 2019 redistributable will affect programs built with Visual C++ 2015 and 2017 also. However, installing the Visual C++ 2015 redistributable will not replace the newer versions of the files installed by the Visual C++ 2017 and 2019 redistributables.

This is different from all previous Visual C++ versions, as they each had their own distinct runtime files, not shared with other versions.

4. 불용어 다운



공유

보기

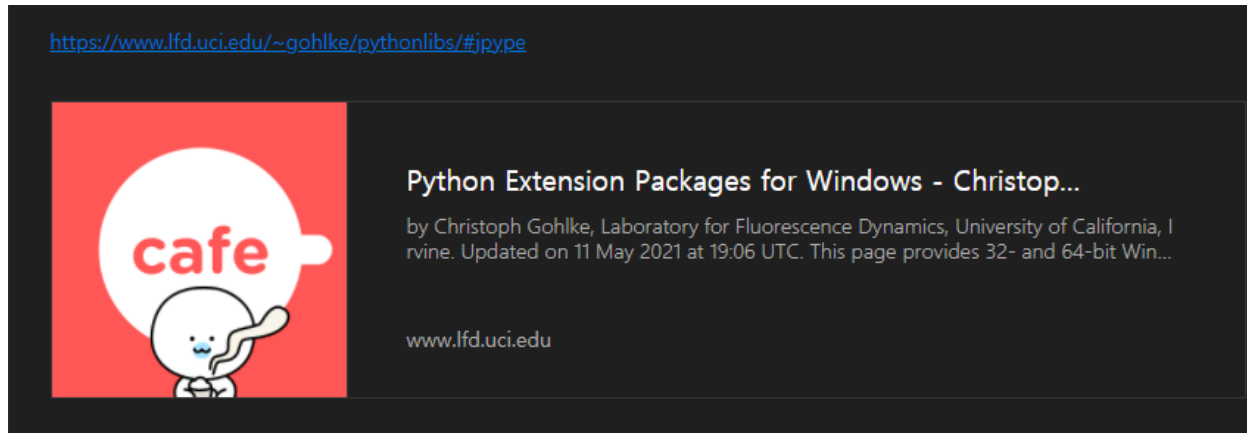
각자 data 폴더에 넣기

📁 > 내 PC > 학생방 (D:) > swo > data

data 검색

이름	수정한 날짜	유형	크기
📄 korean_stopwords.txt	2021-05-12 오후 3:33	텍스트 문서	7KB

5. <https://www.lfd.uci.edu/~gohlke/pythonlibs/#jpype>



JPype: allows full access to Java class libraries.

[JPype1-1.2.0-cp39-cp39-win_amd64.whl](#)

[JPype1-1.2.0-cp39-cp39-win32.whl](#)

[JPype1-1.2.0-cp38-cp38-win_amd64.whl](#)

[JPype1-1.2.0-cp38-cp38-win32.whl](#)

[JPype1-1.2.0-cp37-cp37m-win_amd64.whl](#)

3.7

[JPype1-1.2.0-cp37-cp37m-win32.whl](#)

[JPype1-1.2.0-cp36-cp36m-win_amd64.whl](#)

[JPype1-1.2.0-cp36-cp36m-win32.whl](#)

[JPype1-1.1.2-cp39-cp39-win_amd64.whl](#)

[JPype1-1.1.2-cp39-cp39-win32.whl](#)

[JPype1-1.1.2-cp38-cp38-win_amd64.whl](#)

[JPype1-1.1.2-cp38-cp38-win32.whl](#)

[JPype1-1.1.2-cp37-cp37m-win_amd64.whl](#)

[JPype1-1.1.2-cp37-cp37m-win32.whl](#)

[JPype1-1.1.2-cp36-cp36m-win_amd64.whl](#)

[JPype1-1.1.2-cp36-cp36m-win32.whl](#)

[JPype1-0.7.1-cp35-cp35m-win_amd64.whl](#)

[JPype1-0.7.1-cp35-cp35m-win32.whl](#)

[JPype1-0.7.1-cp27-cp27m-win_amd64.whl](#)

[JPype1-0.7.1-cp27-cp27m-win32.whl](#)

[JPype1-0.6.3-cp34-cp34m-win_amd64.whl](#)

[JPype1-0.6.3-cp34-cp34m-win32.whl](#)

관리자: Anaconda Prompt - python

```
(base) C:\Users\#tjoeun709-19\Downloads>cd ..
```

```
(base) C:\Users\#tjoeun709-19>python
```

```
Python 3.7.0 (default, Jun 28 2018, 08:04:48) [MSC v.1912 64 bit (AMD64)] :: Anaconda, Inc. on win32  
Type "help", "copyright", "credits" or "license" for more information.  
>>>
```

파이썬 버전에 맞는 파일 다운

6.

1. 아래와 같은 파이썬 패키지를 설치합니다. **konlpy** 버전을 반드시 **0.5.1**로 해주어야 합니다. 2020년 현재(9월) 기준, 최신 버전 **konlpy**에서는 일부 개발환경에서 자바 에러가 발생합니다.

- (env_name) `pip install konlpy==0.5.1`
- (env_name) `pip install jpyype1`
- (env_name) `pip install Jpyype1-py3`

```
(base) C:\Users\joeun709-19>cd Downloads
(base) C:\Users\joeun709-19\Downloads>dir J*
C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: 22D3-203D

C:\Users\joeun709-19\Downloads 디렉터리

2021-05-11 오후 07:08      5,583,074 Java (1).pdf
2021-05-10 오후 07:25      5,526,590 Java.pdf
2021-02-02 오전 09:47     15,242,189 JAVA0102의 사본.pdf
2021-02-02 오전 09:22     175,084,680 jdk-8u281-windows-x64.exe
2021-05-12 오후 03:41     1,591,180 JPyype1-1.2.0-cp37-cp37m-win_amd64.whl
                    5개 파일      203,027,713 바이트
                    0개 디렉터리  717,319,929,856 바이트 남음

(base) C:\Users\joeun709-19\Downloads>
```

5번 과정에서 다운 받은 파일 확인

설치

```
(base) C:\Users\joeun709-19\Downloads>pip install JPyype1-1.2.0-cp37-cp37m-win_amd64.whl
```

<Step3. 시각화> : 워드 클라우드 시각화

[pytagcloud 사용하기]

- 아래 코드 실행을 위해, anaconda prompt 혹은 Terminal에서 아래와 같은 패키지들을 설치해 줍니다.

(env_name) `pip install pytagcloud pygame simplejson`

- 그리고 아래와 같은 경로에 한글 폰트(예: NanumBarunGothic.ttf) 파일을 옮깁니다.

- Mac OS : `{anaconda_path}/envs/{env_name}/lib/python3.7/site-packages/pytagcloud/fonts`
- Windows OS : `{anaconda_path}\envs{env_name}\Lib\site-packages\pytagcloud\fonts`
- 폰트 다운로드 : `http://hangeul.naver.com/webfont/NanumGothic/NanumGothic.ttf`

- 파일을 옮긴 후, 파이썬 가상환경을 재실행 하여 주피터를 다시 실행해줍니다. 그래도 아래의 코드가 실행되지 않는 경우는 다음과 같은 작업을 수행합니다.

- 위의 경로에서 fontjson 파일을 편집합니다.
- 아래와 같은 코드를 추가하고 fontjson 파일을 저장합니다.

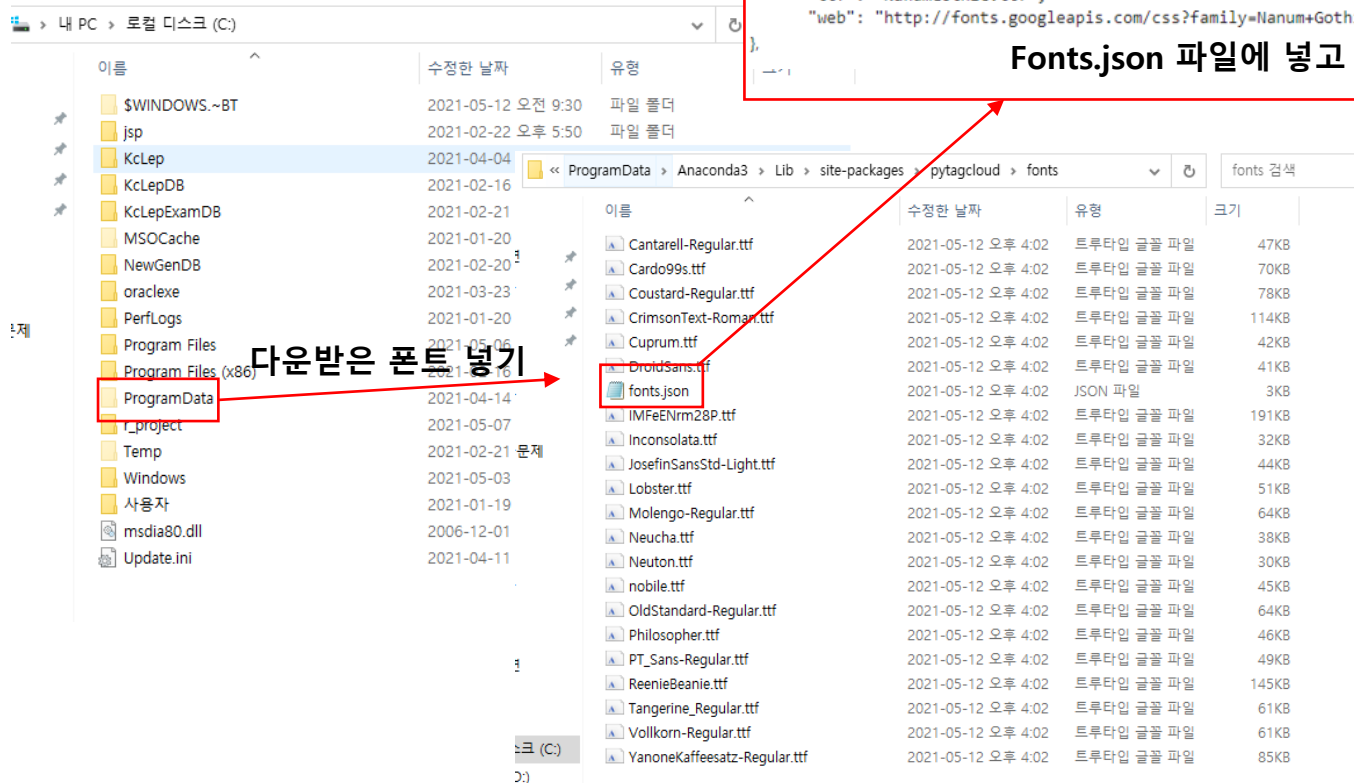
```
{
  "name": "NanumGothic",
  "ttf": "NanumGothic.ttf",
  "web": "http://fonts.googleapis.com/css?family=Nanum+Gothic"
},
```

```
(base) C:\Users\#tjoeun709-19\Downloads>pip install pytagcloud pygame simplejson
```

폰트 다운 받은 후

Fonts.json 파일에 넣고 저장

다운받은 폰트 넣기



<Step1. 크롤링> : 크롤링으로 웹 데이터 가져오기

[BeautifulSoup을 이용한 웹 크롤링]

- 아래 코드 실행을 위해, anaconda prompt 혹은 Terminal에서 아래와 같은 패키지들을 설치해 줍니다.
 - (env_name) `pip install requests`
 - (env_name) `pip install beautifulsoup4`
 - (env_name) `pip install lxml`
- 혹은 아래의 코드로 라이브러리를 설치합니다.

```
!pip3 install requests beautifulsoup4 lxml
```

위에 있는 3개를 한 번에 설치 해줌

```
(base) C:\Users\wtjoeun709-19\Downloads>pip3 install requests beautifulsoup4 lxml
Requirement already satisfied: requests in c:\programdata\anaconda3\lib\site-packages (2.19.1)
Requirement already satisfied: beautifulsoup4 in c:\programdata\anaconda3\lib\site-packages (4.6.0)
Requirement already satisfied: lxml in c:\programdata\anaconda3\lib\site-packages (4.2.5)
Requirement already satisfied: certifi>=2017.4.17 in c:\programdata\anaconda3\lib\site-packages (from requests) (2018.8.24)
Requirement already satisfied: urllib3<1.24,>=1.21.1 in c:\programdata\anaconda3\lib\site-packages (from requests) (1.24.2)
Requirement already satisfied: chardet<3.1.0,>=3.0.2 in c:\programdata\anaconda3\lib\site-packages (from requests) (3.0.4)
Requirement already satisfied: idna<2.8,>=2.5 in c:\programdata\anaconda3\lib\site-packages (from requests) (2.7)
```

9. 모든 패키지&라이브러리 설치 후 수행

텍스트 마이닝을 위한 라이브러리 импорт 및 설정

```
In [2]: import konlpy
```

```
In [3]: # -*- coding: utf-8 -*-

%matplotlib inline

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

import warnings
warnings.filterwarnings("ignore")
```

```
In [4]: import requests
from bs4 import BeautifulSoup
import re

# 크롤링할 사이트 주소를 정의합니다.
source_url = "https://namu.wiki/RecentChanges"

# 사이트의 html 구조에 기반하여 크롤링을 수행합니다.
req = requests.get(source_url)
html = req.content
soup = BeautifulSoup(html, 'lxml')
contents_table = soup.find(name="table")
table_body = contents_table.find(name="tbody")
table_rows = table_body.find_all(name="tr")
```

```
In [5]: from konlpy.tag import Okt
from collections import Counter

# konlpy의 형태소 분석기로 명사 단위의 키워드를 추출합니다.
nouns_tagger = Okt()
```

```
In [6]: import random
import pytagcloud
import webbrowser

pygame 2.0.1 (SDL 2.0.14, Python 3.7.0)
Hello from the pygame community. https://www.pygame.org/contribute.html
```

```
In [7]: from IPython.display import Image
```

```
import konlpy
```

```
# -*- coding: utf-8 -*-
```

```
%matplotlib inline
```

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```
import warnings
warnings.filterwarnings("ignore")
```

```
import requests
from bs4 import BeautifulSoup
import re
```

```
# 크롤링할 사이트 주소를 정의합니다.
source_url = "https://namu.wiki/RecentChanges"
```

```
# 사이트의 html 구조에 기반하여 크롤링을 수행합니다.
req = requests.get(source_url)
html = req.content
soup = BeautifulSoup(html, 'lxml')
contents_table = soup.find(name="table")
table_body = contents_table.find(name="tbody")
table_rows = table_body.find_all(name="tr")
```

```
from konlpy.tag import Okt
from collections import Counter
```

```
# konlpy의 형태소 분석기로 명사 단위의 키워드를 추출합니다.
nouns_tagger = Okt()
```

```
import random
import pytagcloud
import webbrowser
```

```
from IPython.display import Image
```