Colab 환경에서의 자연어 처리 시작하기

1-1 한글 폰트 설정

1-2 한글 적용 확인

1-3 konlpy 설치

1-4 한글 엔진을 이용한 간단한 예제

In [1]:

```
%matplotlib inline
import matplotlib as mpl          # 기본 설정 만지는 용도
import matplotlib.pyplot as plt       # 그래프 그리는 용도
import matplotlib.font_manager as fm # 폰트 관련 용도
```

colab 환경에서 한글 적용을 위한 나눔 고딕 설치

In [2]:

```
### 나눔 고딕 설치
!apt-get update -qq   # 설치를 업데이트   -qq  : 로그를 최소한으로
!apt-get install fonts-nanum* -qq # 설치한다. fonts-nanum*   => ttf-nanum, ttf-nanum-coding, ttf-n
```

```
Selecting previously unselected package fonts-nanum.
(Reading database ... 160772 files and directories currently installed.)
Preparing to unpack .../fonts-nanum_20170925-1_all.deb ...
Unpacking fonts-nanum (20170925-1) ...
Selecting previously unselected package fonts-nanum-eco.
Preparing to unpack .../fonts-nanum-eco_1.000-6_all.deb ...
Unpacking fonts-nanum-eco (1.000-6) ...
Selecting previously unselected package fonts-nanum-extra.
Preparing to unpack .../fonts-nanum-extra_20170925-1_all.deb ...
Unpacking fonts-nanum-extra (20170925-1) ...
Selecting previously unselected package fonts-nanum-coding.
Preparing to unpack .../fonts-nanum-coding_2.5-1_all.deb ...
Unpacking fonts-nanum-coding (2.5-1) ...
Setting up fonts-nanum-extra (20170925-1) ...
Setting up fonts-nanum (20170925-1) ...
Setting up fonts-nanum-coding (2.5-1) ...
Setting up fonts-nanum-eco (1.000-6) ...
Processing triggers for fontconfig (2.12.6-0ubuntu2) ...
```

In [4]:

```
path = '/usr/share/fonts/truetype/nanum/NanumGothicEco.ttf'
font_name = fm.FontProperties(fname=path, size=10).get_name()
print(font_name)
plt.rc('font', family=font_name)

# 우선 fm._rebuild() 를 해주고 # 폰트 매니저 재빌드가 필요하다.
fm._rebuild()
```

NanumGothic Eco

런타임 재기동 후,

- (방법 1) CTRL + M . 을 실행
- (방법 2) 메뉴의 런타임 선택 후, 런타임 다시 시작 선택
- 데이터 준비
- 라이브러리 import
- 폰트 설정 후, 확인

In [1]:

```
%matplotlib inline
import matplotlib as mpl # 기본 설정 만지는 용도
import matplotlib.pyplot as plt # 그래프 그리는 용도
import matplotlib.font_manager as fm # 폰트 관련 용도
import numpy as np

path = '/usr/share/fonts/truetype/nanum/NanumGothicEco.ttf' # 설치된 나눔글꼴중 원하는 녀석의 전체
font_name = fm.FontProperties(fname=path, size=10).get_name()
print(font_name)
plt.rc('font', family=font_name)

## 음수 표시되도록 설정
mpl.rcParams['axes.unicode_minus'] = False
```
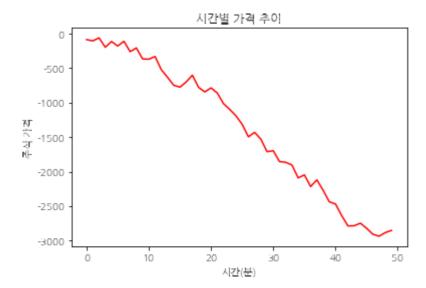
NanumGothic Eco

In [3]:

```
# 데이터 준비
data = np.random.randint(-200, 100, 50).cumsum()
data
```

Out[3]:

```
array([  -85,  -104,   -60,  -196,  -113,  -178,  -110,  -260,  -206,
        -365,  -369,  -330,  -520,  -631,  -746,  -775,  -699,  -601,
        -778,  -843,  -785,  -859, -1013, -1097, -1190, -1313, -1494,
       -1430, -1527, -1710, -1696, -1853, -1863, -1903, -2089, -2046,
       -2215, -2118, -2267, -2436, -2468, -2637, -2786, -2783, -2746,
       -2820, -2906, -2935, -2882, -2851])
```

In [4]:

```python
# 그래프를 그려보자. 이번에는 정상
plt.plot(range(50), data, 'r')
plt.title('시간별 가격 추이')
plt.ylabel('주식 가격')
plt.xlabel('시간(분)')
plt.style.use('seaborn-pastel')
plt.show()
```


1-3 konlpy 소개 및 설치

- 설치 : pip install konlpy
- 웹 사이트 : https://konlpy.org/ko/latest/ (https://konlpy.org/ko/latest/)
- KoNLPy는 파이썬 프로그래밍 언어로 사용이 가능.

In [6]:

```
!pip install konlpy
```

Collecting konlpy
  Downloading https://files.pythonhosted.org/packages/85/0e/f385566fec837c0b83f216b2
da65db9997b35dd675e107752005b7d392b1/konlpy-0.5.2-py2.py3-none-any.whl (https://file
s.pythonhosted.org/packages/85/0e/f385566fec837c0b83f216b2da65db9997b35dd675e1077520
05b7d392b1/konlpy-0.5.2-py2.py3-none-any.whl) (19.4MB)
     |████████████████████████████████| 19.4MB 1.5MB/s
Collecting JPype1>=0.7.0
  Downloading https://files.pythonhosted.org/packages/98/88/f817ef1af6f794e8f11313dc
d1549de833f4599abcec82746ab5ed086686/JPype1-1.3.0-cp37-cp37m-manylinux_2_5_x86_64.ma
nylinux1_x86_64.whl (https://files.pythonhosted.org/packages/98/88/f817ef1af6f794e8f
11313dcd1549de833f4599abcec82746ab5ed086686/JPype1-1.3.0-cp37-cp37m-manylinux_2_5_x8
6_64.manylinux1_x86_64.whl) (448kB)
     |████████████████████████████████| 450kB 33.0MB/s
Collecting colorama
  Downloading https://files.pythonhosted.org/packages/44/98/5b86278fbbf250d239ae0ecb
724f8572af1c91f4a11edf4d36a206189440/colorama-0.4.4-py2.py3-none-any.whl (https://fi
les.pythonhosted.org/packages/44/98/5b86278fbbf250d239ae0ecb724f8572af1c91f4a11edf4d
36a206189440/colorama-0.4.4-py2.py3-none-any.whl)
Collecting beautifulsoup4==4.6.0
  Downloading https://files.pythonhosted.org/packages/9e/d4/10f46e5cfac773e22707237b
fcd51bbffeaf0a576b0a847ec7ab15bd7ace/beautifulsoup4-4.6.0-py3-none-any.whl (https://
files.pythonhosted.org/packages/9e/d4/10f46e5cfac773e22707237bfcd51bbffeaf0a576b0a84
7ec7ab15bd7ace/beautifulsoup4-4.6.0-py3-none-any.whl) (86kB)
     |████████████████████████████████| 92kB 8.2MB/s
Requirement already satisfied: lxml>=4.1.0 in /usr/local/lib/python3.7/dist-packages
(from konlpy) (4.2.6)
Requirement already satisfied: tweepy>=3.7.0 in /usr/local/lib/python3.7/dist-packag
es (from konlpy) (3.10.0)
Requirement already satisfied: numpy>=1.6 in /usr/local/lib/python3.7/dist-packages
 (from konlpy) (1.19.5)
Requirement already satisfied: typing-extensions; python_version < "3.8" in /usr/loc
al/lib/python3.7/dist-packages (from JPype1>=0.7.0->konlpy) (3.7.4.3)
Requirement already satisfied: requests-oauthlib>=0.7.0 in /usr/local/lib/python3.7/
dist-packages (from tweepy>=3.7.0->konlpy) (1.3.0)
Requirement already satisfied: requests[socks]>=2.11.1 in /usr/local/lib/python3.7/d
ist-packages (from tweepy>=3.7.0->konlpy) (2.23.0)
Requirement already satisfied: six>=1.10.0 in /usr/local/lib/python3.7/dist-packages
(from tweepy>=3.7.0->konlpy) (1.15.0)
Requirement already satisfied: oauthlib>=3.0.0 in /usr/local/lib/python3.7/dist-pack
ages (from requests-oauthlib>=0.7.0->tweepy>=3.7.0->konlpy) (3.1.1)
Requirement already satisfied: chardet<4,>=3.0.2 in /usr/local/lib/python3.7/dist-pa
ckages (from requests[socks]>=2.11.1->tweepy>=3.7.0->konlpy) (3.0.4)
Requirement already satisfied: idna<3,>=2.5 in /usr/local/lib/python3.7/dist-package
s (from requests[socks]>=2.11.1->tweepy>=3.7.0->konlpy) (2.10)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.7/dist-p
ackages (from requests[socks]>=2.11.1->tweepy>=3.7.0->konlpy) (2021.5.30)
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in /usr/loca
l/lib/python3.7/dist-packages (from requests[socks]>=2.11.1->tweepy>=3.7.0->konlpy)
 (1.24.3)
Requirement already satisfied: PySocks!=1.5.7,>=1.5.6; extra == "socks" in /usr/loca
l/lib/python3.7/dist-packages (from requests[socks]>=2.11.1->tweepy>=3.7.0->konlpy)
 (1.7.1)
Installing collected packages: JPype1, colorama, beautifulsoup4, konlpy
  Found existing installation: beautifulsoup4 4.6.3
    Uninstalling beautifulsoup4-4.6.3:

```
      Successfully uninstalled beautifulsoup4-4.6.3
Successfully installed JPype1-1.3.0 beautifulsoup4-4.6.0 colorama-0.4.4 konlpy-0.5.2
```

In [7]:

```python
import konlpy
```

In [8]:

```python
import nltk
import matplotlib.pyplot as plt
import numpy as np
```

꼬꼬마를 이용한 분석

- 문장 분석
- 명사 분석
- 형태소 분석

문장 분석

In [11]:

```python
from konlpy.tag import Kkma
k = Kkma()
k.sentences("안녕하세요! 오늘은 한글 분석을 시작합니다.")
```

Out[11]:

```
['안녕하세요!', '오늘은 한글 분석을 시작합니다.']
```

명사 분석

In [12]:

```python
k.nouns("안녕하세요! 오늘은 한글 분석을 시작합니다.")
```

Out[12]:

```
['안녕', '오늘', '한글', '분석']
```

형태소 분석

- http://kkma.snu.ac.kr/documents/index.jsp?doc=postag (http://kkma.snu.ac.kr/documents/index.jsp?doc=postag) : 한글 형태소 분석기 품사 태그표
- (예) NNG : 일반 명사, XSV : 동사 파생 접미사, EFN : 평서형 종결 어미

In [13]:

```
k.pos("안녕하세요! 오늘은 한글 분석을 시작합니다.")
```

Out[13]:

```
[('안녕', 'NNG'),
 ('하', 'XSV'),
 ('세요', 'EFN'),
 ('!', 'SF'),
 ('오늘', 'NNG'),
 ('은', 'JX'),
 ('한글', 'NNG'),
 ('분석', 'NNG'),
 ('을', 'JKO'),
 ('시작하', 'VV'),
 ('ㅂ니다', 'EFN'),
 ('.', 'SF')]
```

여러가지 엔진 사용해 보기

- Hannanum
- Okt (예전 - Twitter)
- Kkma
- Mecab
- Komoran

In [16]:

```
from konlpy.tag import Hannanum
hannanum = Hannanum()
hannanum.pos("아버지가방에 들어가신다")
```

Out[16]:

```
[('아버지가방', 'N'), ('에', 'J'), ('들', 'P'), ('어', 'E'), ('가', 'P'), ('시ㄴ다',
'E')]
```

In [17]:

```
from konlpy.tag import Kkma
k = Kkma()
k.pos("아버지가방에 들어가신다")
```

Out[17]:

```
[('아버지', 'NNG'),
 ('가방', 'NNG'),
 ('에', 'JKM'),
 ('들어가', 'VV'),
 ('시', 'EPH'),
 ('ㄴ다', 'EFN')]
```

In [19]:

```python
from konlpy.tag import Komoran
k = Komoran()
k.pos("아버지가방에 들어가신다")
```

Out[19]:

```
[('아버지', 'NNG'),
 ('가방', 'NNP'),
 ('에', 'JKB'),
 ('들어가', 'VV'),
 ('시', 'EP'),
 ('ㄴ다', 'EC')]
```

In [22]:

```python
from konlpy.tag import Okt
okt = Okt()
okt.pos("아버지가방에 들어가신다")
```

Out[22]:

```
[('아버지', 'Noun'), ('가방', 'Noun'), ('에', 'Josa'), ('들어가신다', 'Verb')]
```

- Mecab는 설치후 진행해야 함.

(실습) 나는 밥을 먹는다 를 품사 태깅을 해보자

말뭉치 사용해 보기

말뭉치(corpus) 사용

- kolaw : 한국 법률 말뭉치 : constitution.txt
- kobill : 대한민국 국회 의안 말뭉치. 파일 ID는 의안 번호를 의미
  - 1809890.txt ~ 1809899.txt

In [26]:

```python
from konlpy.corpus import kolaw
c = kolaw.open('constitution.txt').read()
print( c[:15])
```

대한민국헌법

유구한 역사와

In [27]:

```python
from konlpy.corpus import kobill
d = kobill.open('1809890.txt').read()
print( d[:20])
```

지방공무원법 일부개정법률안

(정의화

Ref

- KoNLPy : 파이썬 한국어 NLP

- konlpy를 사용한 사용 예시
    - https://konlpy.org/ko/latest/examples/ (https://konlpy.org/ko/latest/examples/)
- 사전 : https://konlpy.org/ko/latest/data/ (https://konlpy.org/ko/latest/data/)