In [7]:

```
import nltk
import matplotlib
import numpy as np
import pandas as pd
from PIL import Image
from konlpy.tag import Okt, Mecab
from collections import Counter
from nltk.corpus import stopwords
import requests, os, re, time, json
from wordcloud import WordCloud, STOPWORDS
import matplotlib.pyplot as plt
%matplotlib inline
from IPython.display import set_matplotlib_formats
matplotlib.rc('font',family = 'Malgun Gothic')
set_matplotlib_formats('retina')
matplotlib.rc('axes',unicode_minus = False)
```

In [15]:

```
site = 'https://openapi.naver.com/v1/search/news.json'

params = {
        'query' : '고속도로',
        'start' : 1,
        'display' : 100,
        'sort' : 'sim'}

headers = {
          'X-Naver-Client-Id' : '_HV4Mav6gAsfgons6mTA',
          'X-Naver-Client-Secret' : 'esqEqzOGLv'}

while True :
    time.sleep(1)
    data_dict = {
                'title'       : [],
                'description' : []
                }
    print(f'{params["start"]} 수집중')
    res = requests.get(site, params=params, headers=headers)
    json_root = json.loads(res.text)
    items = json_root['items']

    for item in items :
        title = item['title']
        description = item['description']
        data_dict['title'].append(title)
        data_dict['description'].append(description)


    df1 = pd.DataFrame(data_dict)

    if os.path.exists('고속국도.csv') == False :
        df1.to_csv('고속국도.csv', encoding='utf-8-sig', index=False)
    else :
        df1.to_csv('고속국도.csv', encoding='utf-8-sig', index=False, header=False, mode='a')

    Start = json_root['start']
    Start += 100
    if Start < 1000 :
        params['start'] = Start
    else : break

print('수집완료')
```

```
1 수집중
101 수집중
201 수집중
301 수집중
401 수집중
501 수집중
601 수집중
701 수집중
801 수집중
901 수집중
수집완료
```

In [17]:

```
df = pd.read_csv('고속국도.csv')
df.head()
```

Out[17]:

	title	description
0	인천공항고속도로 차량 사고후 화재…음주여부 조사	인천공항고속도로에서 재규어 차량을 몰던 30대가 빗길에 미끄러져 중앙분…

1	美 당국, 테슬라 '오토파일럿' 추돌사고 11건 조사 착수	18년부터 11건사고로 1명 사망, 17명 부상 '완전한 자율운행으로 ...
2	인천공항고속도로 승용차 사고로 전소...음주운전 정황 확인 중	오늘(17일) 새벽 1시 50분쯤 인천국제공항고속도로 서울 방면 김포공...
3	美 교통당국, 테슬라 자율주행 충돌 사고 11건 조사 착수	미국 교통안전 규제당국이 전기차 테슬라의 오토파일럿(자율주행)과 연관된 11건의 충...
4	고속도로 승용차-화물차 추돌…교통사고 잇따라	밤사이 사건사고, 임상재 기자입니다. ◀ 리포트 ▶ 화물차 앞부분이 형...

In [18]:

```
def text_cleaning(text) :
    hangul = re.compile('[^ㄱ-ㅣ가-힣]+')
    result = hangul.sub('', str(text))
    return result

df['title']       = df['title'].apply(lambda x : text_cleaning(x))
df['description'] = df['description'].apply(lambda x : text_cleaning(x))

title_corpus       = ''.join(df['title'].tolist())
description_corpus = ''.join(df['description'].tolist())
```

In [19]:

```
# tagger = Okt()
tagger = Mecab('C:\Mecab\mecab-ko-dic')
title_nouns = tagger.nouns(title_corpus)
description_nouns = tagger.nouns(description_corpus)

title_count = Counter(title_nouns)
description_count = Counter(description_nouns)

with open('korean_stopwords.txt', encoding='utf-8') as fp :
    stopwords = fp.readlines()

stopwords = [x.strip() for x in stopwords]

title_dict = {}
description_dict = {}

for key in title_count :
    if len(key) > 1 :
        title_dict[key] = title_count[key]

for key in description_count :
    if len(key) > 1 :
        description_dict[key] = description_count[key]

remove_title_count = Counter(title_dict)
remove_description_count = Counter(description_dict)

title_dict = {}
for key in remove_title_count :
    if key not in stopwords :
        title_dict[key] = remove_title_count[key]

description_dict = {}
for key in remove_description_count :
    if key not in stopwords :
        description_dict[key] = remove_description_count[key]

remove_title_count = Counter(title_dict)
remove_description_count = Counter(description_dict)
```

In [20]:

```
del remove_title_count['고속국도']
del remove_description_count['고속국도']
```

In [22]:

```
wc = WordCloud(stopwords=spwords, font_path="c:/Windows/Fonts/malgun.ttf", background_color='white', width=500, c
               height=500)
wc.generate_from_frequencies(dict(tag))
plt.figure(figsize=(12, 12))
plt.imshow(wc, interpolation="bilinear")
plt.axis('off')
plt.show()
```

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js