

## 주식 데이터분석

주식을 데이터 분석하기전 우리나라의 지수를 먼저 파악한 후 현재 제일인기 있는 종목인 LG에너지솔루션

관련 기사를 웹크롤링해서 가장 인기 있는 검색어를 도출 하고 워드 클라우드 시각화를 해서

종목의 차트 분석 및 매수/매도 타이밍 도출이 목표.

### 1.우리나라의 지수 파악하기 kospi 최대 손실 낙폭

In [1]:

```
pip install pandas_datareader
```

```
Requirement already satisfied: pandas_datareader in c:\Users\WyjWanaconda3\envs\WpybookWlib\site-packages (0.10.0)
Requirement already satisfied: lxml in c:\Users\WyjWanaconda3\envs\WpybookWlib\site-packages (from pandas_datareader) (4.7.1)
Requirement already satisfied: pandas>=0.23 in c:\Users\WyjWanaconda3\envs\WpybookWlib\site-packages (from pandas_datareader) (1.3.5)
Requirement already satisfied: requests>=2.19.0 in c:\Users\WyjWanaconda3\envs\WpybookWlib\site-packages (from pandas_datareader) (2.27.1)
Requirement already satisfied: python-dateutil>=2.7.3 in c:\Users\WyjWanaconda3\envs\WpybookWlib\site-packages (from pandas>=0.23->pandas_datareader) (2.8.2)
Requirement already satisfied: numpy>=1.17.3 in c:\Users\WyjWanaconda3\envs\WpybookWlib\site-packages (from pandas>=0.23->pandas_datareader) (1.21.5)
Requirement already satisfied: pytz>=2017.3 in c:\Users\WyjWanaconda3\envs\WpybookWlib\site-packages (from pandas>=0.23->pandas_datareader) (2021.3)
Requirement already satisfied: six>=1.5 in c:\Users\WyjWanaconda3\envs\WpybookWlib\site-packages (from python-dateutil>=2.7.3->pandas>=0.23->pandas_datareader) (1.16.0)
Requirement already satisfied: charset-normalizer~>2.0.0 in c:\Users\WyjWanaconda3\envs\WpybookWlib\site-packages (from requests>=2.19.0->pandas_datareader) (2.0.10)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\Users\WyjWanaconda3\envs\WpybookWlib\site-packages (from requests>=2.19.0->pandas_datareader) (1.26.8)
Requirement already satisfied: idna<4,>=2.5 in c:\Users\WyjWanaconda3\envs\WpybookWlib\site-packages (from requests>=2.19.0->pandas_datareader) (3.3)
Requirement already satisfied: certifi>=2017.4.17 in c:\Users\WyjWanaconda3\envs\WpybookWlib\site-packages (from requests>=2.19.0->pandas_datareader) (2021.10.8)
Note: you may need to restart the kernel to use updated packages.
```

In [2]:

```
pip install yfinance
```

```
Requirement already satisfied: yfinance in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (0.1.69)
Requirement already satisfied: numpy>=1.15 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from yfinance) (1.21.5)
Requirement already satisfied: requests>=2.26 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from yfinance) (2.27.1)
Requirement already satisfied: multitasking>=0.0.7 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from yfinance) (0.0.10)
Requirement already satisfied: lxml>=4.5.1 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from yfinance) (4.7.1)
Requirement already satisfied: pandas>=0.24 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from yfinance) (1.3.5)
Requirement already satisfied: python-dateutil>=2.7.3 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from pandas>=0.24->yfinance) (2.8.2)
Requirement already satisfied: pytz>=2017.3 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from pandas>=0.24->yfinance) (2021.3)
Requirement already satisfied: six>=1.5 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from python-dateutil>=2.7.3->pandas>=0.24->yfinance) (1.16.0)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from requests>=2.26->yfinance) (1.26.8)
Requirement already satisfied: certifi>=2017.4.17 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from requests>=2.26->yfinance) (2021.10.8)
Requirement already satisfied: charset-normalizer~=2.0.0 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from requests>=2.26->yfinance) (2.0.10)
Requirement already satisfied: idna<4,>=2.5 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from requests>=2.26->yfinance) (3.3)
Note: you may need to restart the kernel to use updated packages.
```

In [3]:

```

#코스피의 MDD(Maximum Draw Down)고점 대비 최대 손실폭 구하기 최저점-최고점/최저점
#rolling() 함수는 시리즈에서 윈도우 크기에 해당하는 개수만큼 데이터를 추출하여 집계 함수에 해당하는
#집계함수로는 최대값max(), 평균값mean(), 최소값min(),을 사용할수 있다

from pandas_datareader import data as pdr
import yfinance as yf
yf.pdr_override()
import matplotlib.pyplot as plt

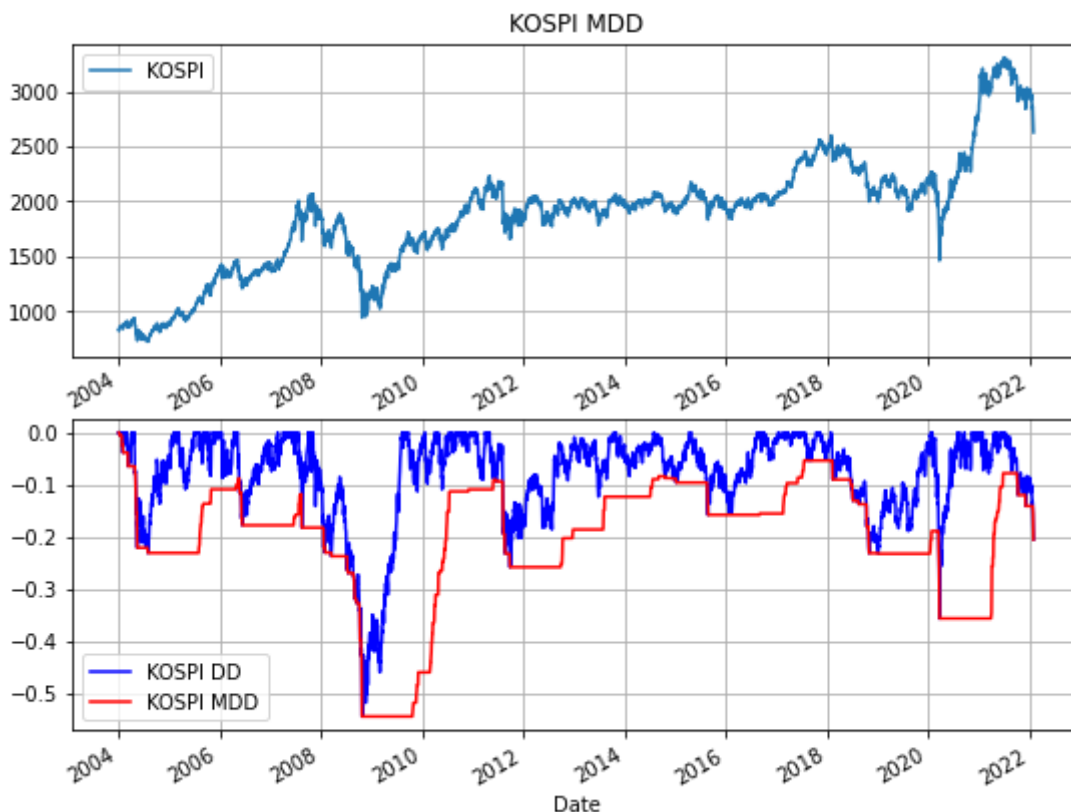
kospi=pdr.get_data_yahoo('^KS11', '2004-01-04') # 코스피지수 데이터를 다운. 코스피지수의 심볼은 '^KS11'

window=252 #산정 기간에 해당하는 window값은 1년 동안의 개장일을 252일로 어림잡아 설정함.
peak=kospi['Adj Close'].rolling(window, min_periods=1).max() #코스피 종가 칼럼에서 1년(거래일 기준)
drawdown=kospi['Adj Close']/peak -1.0 # drawdown은 최고치(peak)대비 현재 코스피 종가가 얼마나 하락했는지
max_dd=drawdown.rolling(window, min_periods=1).min() #drawdown은 1년 기간 단위로 최저치 max_dd를 구함

plt.figure(figsize=(9,7))
plt.subplot(211) #2행 1열중 1행을 그린다.
kospi['Close'].plot(label='KOSPI', title='KOSPI MDD', grid=True, legend=True)
plt.subplot(212) #2행 1열중 2행에 그린다.
drawdown.plot(c='blue', label='KOSPI DD', grid=True, legend=True)
max_dd.plot(c='red', label='KOSPI MDD', grid=True, legend=True)
plt.show()

```

[\*\*\*\*\*100%\*\*\*\*\*] 1 of 1 completed



In [4]:

```
#서브프라임 금융 위기 당시 2008년 10월 24일에 코스피 지수가 10.57% 하락하면서 MDD가 -54.5% 를 기록함
max_dd.min()
```

Out[4]:

```
-0.5453665130144085
```

In [5]:

```
#코스피와 다우존스 지수 비교
```

```
from pandas_datareader import data as pdr
import yfinance as yf
yf.pdr_override()

dow=pdr.get_data_yahoo('^DJI', '2000-01-04')
kospi=pdr.get_data_yahoo('^KS11', '2000-01-04')

import matplotlib.pyplot as plt
plt.figure(figsize=(9,5))
plt.plot(dow.index, dow.Close, 'r--', label='Dow Jones Industrial') #다우존스 지수를 붉은 점선으로 출력
plt.plot(kospi.index, kospi.Close, 'b', label='KOSPI') #코스피를 푸른 실선으로 출력.
plt.grid(True)
plt.legend(loc='best')
plt.show()
```

```
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
```



In [6]:

```
#지수 기준값이 달라서, 어느 지수가 더 좋은 성과를 냈는지를 한눈에 알아보기 어렵다.
#일별 증가만으로는 코스피와 다우존스 지수의 상관관계를 비교하기가 어려웠다.
#이번에는 현재 증가를 특정 시점의 증가로 나누어 변동률을 구함
```

```
from pandas_datareader import data as pdr
import yfinance as yf
yf.pdr_override()
```

```
d=(dow.Close / dow.Close.loc['2000-01-04'])*100 #금일 다우존스 지수를 2000년 1월 4일 다우존스 지수
k=(kospi.Close / kospi.Close.loc['2000-01-04'])*100 #금일 코스피 지수를 2000년 1월 4일 코스피 지수로
```

```
import matplotlib.pyplot as plt
plt.figure(figsize=(9,5))
plt.plot(d.index, d, 'r--', label='Dow Jones Industrial Average') #다우존스 지수를 붉은 점선으로 출력
plt.plot(k.index, k, 'b', label='KOSPI') #코스피를 푸른 실선으로 출력.
plt.grid(True)
plt.legend(loc='best')
plt.show()
```



미국의 다우존스와 코스피 지수가 비슷하게 흘러가고 있지만, 최근에는 코스피가 훨씬

밀리고 있다. 그래서 우리나라의 시총1위인 삼성전자와 코스피를 비교해본다.

우선 LG에너지솔루션을 분석하고 싶지만 오늘 상장이라 데이터가 너무 부족해서 삼성으로함.

In [7]:

#코스피와 다우존스 지수 비교

```
from pandas_datareader import data as pdr
import yfinance as yf
yf.pdr_override()

samsung=pdr.get_data_yahoo('005380.KS', '2000-01-04')
kospi=pdr.get_data_yahoo('^KS11', '2000-01-04')

import matplotlib.pyplot as plt
plt.figure(figsize=(9,5))
plt.plot(samsung.index, samsung.Close, 'b--', label='samsung') #다우존스 지수를 붉은 점선으로 출력한
plt.plot(kospi.index, kospi.Close, 'r', label='KOSPI') #코스피를 푸른 실선으로 출력.
plt.grid(True)
plt.legend(loc='best')
plt.show()
```

```
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
```



In [8]:

```
#지수 기준값이 달라서, 어느 지수가 더 좋은 성과를 냈는지를 한눈에 알아보기 어렵다.
#일별 증가만으로는 코스피와 삼성전자의 상관관계를 비교하기가 어려웠다.
#이번에는 현재 증가를 특정 시점의 증가로 나누어 변동률을 구함
```

```
from pandas_datareader import data as pdr
import yfinance as yf
yf.pdr_override()
```

```
s=(samsung.Close / samsung.Close.loc['2000-01-04'])*100 #금일삼성전자 지수를 2000년 1월 4일 삼성전
k=(kospi.Close / kospi.Close.loc['2000-01-04'])*100 #금일 코스피 지수를 2000년 1월 4일 코스피 지수로
```

```
import matplotlib.pyplot as plt
plt.figure(figsize=(9,5))
plt.plot(s.index, s, 'b--', label='samsung') #삼성전자 지수를 붉은 점선으로 출력한다.
plt.plot(k.index, k, 'r', label='KOSPI') #코스피를 푸른 실선으로 출력.
plt.grid(True)
plt.legend(loc='best')
plt.show()
```



## 2.네이버뉴스 에서 lg에너지솔루션 관련 기사 웹크롤링 해서 워드클라우드 시각화 하기

pip install selenium

pip install beautifulsoup4

In [10]:

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

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



In [11]:

```
"""네이버 뉴스 기사 웹 크롤러 모듈"""
```

```
from bs4 import BeautifulSoup #라이브러리 불러오기
import urllib.request
```

```
# 출력 파일 명
```

```
OUTPUT_FILE_NAME = 'output.txt'
```

```
# 굵어 올 URL url주소를 백 슬래시를 사용해 두 줄로 나누었음. 이는 한 행이 80자 넘어가지 않게 하기위
# 네이버에 lg에너지 솔루션 최근 기사 하나를 불러옴.
```

```
URL='https://finance.naver.com/news/news_read.naver?article_id=0005036548&office_id=277'W
    '&mode=mainnews&type=&date=2022-01-27&page=1'
```

```
# 크롤링 함수 해당 url주소로 요청을 보내고 받아 기사 내용을 파싱해서 하나의 문자열로 저장하는 함수.
# 1.우선 urlopen메서드를 통해서 url주소에 대한 요청을 보내 source_code_from_URL 변수에 그 결과를 저장
# 2.그 후 html코드를 파싱하기 위해 source_code_from_URL를 이용, BeautifulSoup객체를 생성해 soup에 할
# 3.beautifulSoup객체 생성자의 2번째 인자로 'lxml'을 사용해 기존 'html'방식 대신 'lxml'방식으로 파싱
# 3.1 그리고 한글 내용이 포함된 기사이기 때문에 from_encoding을 이용해 키워드 인자 cp949 넣어 인코딩
# 4. 'lxml'방식으로 BeautifulSoup 객체를 생성할때 오류나면 lxml라이브러리를 설치하면됨.
# 5. soup객체에서 원하는 부분(html요소)만 가지고 와야함.
# 5.1 id가 contentarea인 div클래스 안에 담겨 있음을 알수 있음.
# 5.2 그래서 우리가 생성한 객체 soup에서 위에서의 'div'클래스를 가져오면됨.
```

```
def get_text(URL):
```

```
    source_code_from_URL = urllib.request.urlopen(URL)
```

```
    soup = BeautifulSoup(source_code_from_URL, 'lxml', from_encoding='cp949')
```

```
    text = '' #우선 본문 내용을 저장하기 위해 text에 빈 문자열을 할당함.
```

```
    for item in soup.find_all('div', id='contentarea'): #이후, soup객체의 find_all메서드를 통해 모
        text = text + str(item.find_all(text=True)) #for 문으로 뽑힌 각 요소에 item에 find_all메서
```

```
    return text #텍스트 키워드 인자에 true를 넣어 텍스트 요소만 뽑아 문자열로 치환, text문자열에
#그 후 텍스트를 반환
```

```
# 메인 함수
```

```
def main():
```

```
    open_output_file = open(OUTPUT_FILE_NAME, 'w')
```

```
    result_text = get_text(URL)
```

```
    open_output_file.write(result_text)
```

```
    open_output_file.close()
```

```
#메인 함수를 별도로 만들고 __name__을 이용해 main 함수를 실행시킴.
```

```
#메인 함수에서 위에서 할당한 output_file_name상수를 통해 txt파일을 생성, get_text함수를 사용해
```

```
# 기사내용을 result_text에 할당함.
```

```
# 이후 , 오픈한 output_file에 기사를 쓰고 닫음.
```

```
if __name__ == '__main__':
```

```
    main()
```

In [27]:

```

# 텍스트 정제 모듈
# 영어, 특수기호 모두 제거

#정규표현식을 활용하기위해 're'라이브러리를 임포트함.
import re

#정제할 파일이름을 input_file_name으로,
#결과 파일의 이름을 output_file_name 으로 할당.
# 입,출력 파일명
INPUT_FILE_NAME = 'output.txt'
OUTPUT_FILE_NAME = 'output_cleand.txt'

#문자열을 입력받아 영어와 특수문자를 제거하는 클리닝 함수정의
# 클리닝 함수
def clean_text(text):
    cleaned_text = re.sub('[a-zA-Z]', '', text) #대소문자 영어를 제거하는 코드.
    cleaned_text = re.sub('[W{W}W[W]W/?.,:|W)*~`!^W-_+<>@W#$%&WWW=W(W'W"']',
                          '', cleaned_text) #특수문자를 제거하는 코드.

    return cleaned_text

# 메인 함수
def main():
    read_file = open(INPUT_FILE_NAME, 'r')
    write_file = open(OUTPUT_FILE_NAME, 'w')
    text = read_file.read()
    text = clean_text(text)
    write_file.write(text)
    read_file.close()
    write_file.close()

#이번 모듈도 main정의, 여기서는 clean_text함수를 이용해 본문 내용을 정제함.
if __name__ == "__main__":
    main()

```

## 위에서 정제한 output\_cleand.txt를 이용해 워드 클라우드 시작하기

In [29]:

```

#워드클라우드를 이용하기위해 패키지 불러오기
#twitter는 konlpy의 여러 패키지 중 하나로 형태소를 분석해줌.
#counter는 단어를 카운팅 해주는 패키지.
from konlpy.tag import Twitter
from collections import Counter

```

In [14]:

```
#클라우드로 만들 텍스트 파일 불러오기/ 'r'파일을 읽겠다.
file = open("../220127/output_cleand.txt", "r", encoding="cp949")
#readlines로 불러오면 아래와 같이 리스트 형태로 글이 출력됨.
lists = file.readlines()
file.close()
lists
```

Out[14]:

```
[ ' 콘텐츠 로케이터 금융홍 뉴스 주요뉴스 로케이터 주요뉴스 타이틀
우측 티커 영역 10 1 10 1 0
0 1 1 1 3000 3000 속보 “가상 화폐 보관해드립니다” . . . 현대오
일뱅크 지난해 영업익 1조142 군인공제회 김해 복합스포츠허리 장덕수 품에 안긴 투
자증권 신임 맥쿼리 올해 인플레이가 투자자들의 아이이테크놀로지 영업손실 290억
이어룡 대신금융그룹 회장 나주·과 디스플레이 선명해진 불투명 넷마블 블록체인·메
타버스 도전… 라씨로 따상 대신 급락한 엔솔 로케이션바 추가 인쇄
글자크기 글꼴 본문 텍스트 작게 본문 텍스트 크게 로케이션바 추가
게시판 뷰 언론사 로고 언론사 로고 기사제목 기재일 기사제목
굿모닝 증시엔솔 데뷔로 변동성 확대·매파적인 파월…외국인 수급 변수 기사제목
기사입력 20220127 0816 기사원문 좋아요昏昏해요 3 슬퍼요 2 화나요
1 후속기사 원해요 좋아요 평가하기 공감 좋아요 0昏昏해요 0
슬퍼요 0 화나요 0 후속기사 원해요 0 댓글수표기가 숫자일때 에
클래스 추가 댓글 기사 상단 요점 기사본문 27일 국내 증시는 기업공개
사상 최대어로 꼽히는 에너지솔루션이 코스피 시장에 데뷔해 수급 변동성 확대가 불가피할
것으로 보인다 전반적인 투자심리가 외출된 점을 감안할 때 매물 출회 가능성이 높아 하락
출발이 예상된다 다만 저가 매수 유입도 기대할 수 있을 것으로 전망된다 미국 뉴욕증시
는 26일현지시간 제롬 파월 연방준비제도·연준 의장의 금리인상 예고에 약세로 돌아섰다
이날 뉴욕증시의 다우존스30 산업평균지수는 전장보다 12964포인트038 내린 3만416809에
거래를 마쳤다 스탠더드앤드푸어스 500 지수도 434993으로 652포인트015 하락했으나 기술
주 중심의 나스닥 지수는 282포인트002 오른 1만354212에 장을 마감했다 연방공개시장위원
회 정례회의 종료 직후 500포인트 이상 치솟았던 다우 지수는 이후 파월 의장이 기자회견
에서 노동시장을 위협하지 않고도 금리를 인상할 수 있는 여지가 꽤 많다고 언급한 직후
하락 반전했다 외환중개업체 오안다의 에드워드 모야는 투자자 노트를 통해 파월 의장의
회견을 들은 직후 더 많은 금리인상 가능성이 커졌다는 점이 분명해졌다고 말했다고 방송
이 전했다 아울러 연준이 금리인상 시작 후 대차대조표 축소에 나서겠다는 방침을 공표한
것도 투자 심리에 부정적인 영향을 준 것으로 보인다 연준의 이러한 매파통화긴축 선호적
움직임에 10년물 미 국채 금리는 188까지 치솟으며 금리에 민감한 기술주들에 더 부담을
줬다 ◆서상영 미래에셋증권 연구원 미국 증시가 본격적인 실적 시즌에 대한 기대 심리가
높아지며 상승한 점은 한국 증시에 우호적이다 더불어 그동안 부진한 모습을 보여 왔던 반
도체 업종이 정부의 지원책이 발표되며 강세를 보인 점도 긍정적이다 이런 가운데 미국 연
준은 를 통해 시장 우려보다는 덜 매파적인 내용을 발표한 점도 우호적이나 파월 의장이
매 회의때마다 금리를 인상할 수 있음을 언급하고 양적 긴축을 이전보다 더 빠르게 진행할
수 있다고 주장한 점은 부담이다 이는 외국인 수급에 부정적이기 때문이다 테슬라가 견고
한 실적 발표에도 불구하고 공급망 문제가 향후 부담 요인이라고 주장하자 매물 출회되며
시간외로 하락하는 등 가이던스에 대한 우려가 부각된 점도 부담이다 이를 감안한 한국 증
시는 매파적인 연준에 대한 우려로 원달러 환율이 1203원을 넘어서는 등 원화 약세폭이
확대되는 등 전반적인 투자심리가 외출된 점을 감안할 때 매물 출회 가능성을 높인다 다만
최근 하락에 따른 저가 매수 기대 또한 배제할 수 없어 시장은 하락 출발 후 외국인 수급
에 따라 변화가 예상된다 ◆한지영 키움증권 연구원 1월 성명서상 3월 금리 인상 및
그 이후의 양적긴축 논의가 이전보다 구체화됐다는 점은 이미 시장에서 예상하고 있었던
수준이었다 그러나 이후에 있었던 파월의장의 기자회견이 예상보다 더 매파적이었다는 점
이 시장의 혼란을 재차 유발한 것으로 보인다 특히 고용시장을 저해하지 않은 채 금리를
인상할 여지가 많으며 매 회의 때마다 금리인상을 하는 것도 배제하지 않겠다고 언급했다
는 점 양적 긴축도 과거보다 조기에 빠른 속도로 하는게 적절하다고 언급하는 등 1 월 중
순 상원 청문회 발언 때보다 매파 성향이 짙어진 모습이다 과거 2018 년도 당시 긴축 사
이클에 비해 고용 경제 물가 연준 자산 규모 등 환경이 다르기 때문에 연준 입장에서는 이
전보다 빠른 정책 정상화 자체가 증시에 부담이 되는 것은 사실이다 그러나 이 같은 정책
가속화의 충격은 1월 이후 일정부분 시장에 반영되어 왔으며 예측가능한 수준으로 양적 긴
축을 단행할 것이라는 점에 주목할 필요가 있다 이는 향후 연준이 시장과 지속적인 커뮤니
```

케이션을 통해 최근의 긴축 발작과 같은 시장 충격을 최소화할 것임을 시사한다 이에 1월  
 가 끝난 만큼 당분간 연준발 불확실성은 완화될 전망이다 1월 의사록 발표 이후 2월 16 일  
 시장이 해석하는 과정에서 불확실성이 재차 확대될 수 있다고 판단한다 이날 국내 증시도  
 미국발 악재에 영향을 받아 악세 압력을 받을 전망이다 미국 장 마감 후 테슬라가 어닝 서  
 프라이즈를 기록했음에도 공급난 문제가 연중 지속될 것으로 언급함에 따라 시간외 주가가  
 1 내외 하락세를 보이고 있다는 점도 국내 성장주들의 투자심리에 영향을 줄 것으로 전망  
 된다 더 나아가 국내 증시에는 에너지솔루션이 상장함에 따라 해당 종목 및 2차전지 관련  
 주 시가총액 상위주들의 수급 변동성이 확대될 것으로 예상된다 에너지솔루션의 코스피 편  
 입은 28일에 단행되므로 이날 해당 종목의 주가 변화에 따라 코스피 내 비중 37 레인지로  
 추정으로 편입하려는 기관들의 수급 변화에 따라 시장 전반적인 방향성이 달라질 것으로  
 전망된다 아시아경제 관련뉴스 해당 언론사에서 선정하며 언론사 페이지아웃링크로 이  
 동해 볼 수 있습니다 역술가 박세리에 남편 자리에 자식 같은 男 들어와 있다 박군 도경완  
 앞에서 장윤정이 이상형 밥 먹으라고 35만원 주셨다 김희철 내년 초에 결혼식 올릴 것...비  
 혼 선언하더니 돌연 결혼 계획 발표 비아그라와 함께 먹으면 효과가 4배...몇 분 뒤면 나타  
 난다 길거리서 주문 컵까지 보증금 내주라니...장사는 언제 바이라인 이선애 기자카  
 드 이선애 기록은 기억을 이긴다 관찰하고 확인하고 전달하고 기록하겠습니다  
 동학·서학·코인 개미 죄다 피눈물...패닉셀링 공포는 이제 시작종합 엔솔 개장 10분만에  
 2조 거래·외국인 매도 폭탄 20 급락 카피라이트 © 아시아경제 무단 전재 및  
 재배포 금지 프로모션 2022 임인년 나의 토정비결과 신년운세는 빅데이터·인공지능  
 ·핀테크 취업연계 국비 교육 좋아요 좋아요 0 훈훈해요 0  
 슬퍼요 0 화나요 0 후속기사 원해요 0 목록보기 주요뉴스  
 기사제공 인쇄하기 메일보내기 아시아경제 기사제공 게시판 뷰 이 언론사  
 기사는 뉴스제휴평가위원회 뉴스 제휴 및 제재 심사 규정에 따라 노출이 중단된 기사입니  
 다 이전페이지 목록보기 묶음 뉴스일 경우 추가 필요 묶음 뉴스 묶음  
 뉴스 증시 주요뉴스 오전 美 금리인상 예고에 와르르 코스피 약 14개월 만에 2700  
 선 내줬다 외환브리핑 3월 금리인상 예고한 파월...환율 굿모닝 증시엔솔 데뷔로 변동성  
 확대·매 파월 3년 만의 금리인상 예고...변동성 확 모닝브리핑 뉴욕증시 금리인상 예고  
 에 혼조 일일펀드동향 국내 채권형펀드 25일 하루에 3월 금리인상 예고에 투심 흔들변동  
 이전 페이지로 돌아가기 맨위로 콘텐츠 우측메뉴 끼워넣기 여러개가  
 뉴스홈 배너로 등록되었을 경우에는 하나의 경우만 뿌려주게 한다 가장 많이 본 뉴스  
 더보기 주식 매매 상위 종목 및 환율 2022년 1월 26일자 주요경제지표 2022년 1월 26일  
 자 음악저작권 조각투자하는 플랫폼 '뮤직카우' 증권업 될까 로보어드바이저 3사 지난  
 해 자문·일임 계약 1조 돌파...투자 대안 부각 '스톡옵션 대박' 부풀었던 네이버카카  
 오 직원 이젠 손실 걱정 파월 "물가 더 오를 수 있다" ... 015 데일리 국제금융시장 유  
 럽증시 美 결과 기다리며 상승 마감 첫 월급 비트코인으로 바꾼 뉴욕시장그런데 왜 웃음  
 거리 돼버렸나 가장 많이 본 뉴스 여러개가 뉴스홈 배너로 등록되었을 경우에는  
 하나의 경우만 뿌려주게 한다 인기검색어 더보기 인기검색어 에너지솔루션 5  
 07000 삼성전자 71500 카카오 82700 셀트리온 147500 하이닉스 1  
 13000 인기검색어 우측메뉴 끼워넣기 ']

In [15]:

#형태소나누기

twitter = Twitter()

morphs = [] #morphs=[]라는 빈 리스트를 만들고 그 리스트에 append한다.

#무엇을? lists에 있는 (sentence)의 형태소를 분석한 (twitter)결과를.

for sentence in lists:

morphs.append(twitter.pos(sentence))

print(morphs)

#조사,명사,부사 등등 단어와 형태로 나누어 리스트가 형성.

a), ('은', 'Noun'), ('은', 'Suffix'), ('은', 'Josa'), ('을', 'Noun'), ('시  
 준', 'Noun'), ('에', 'Josa'), ('대한', 'Noun'), ('기대', 'Noun'), ('심리', 'Nou  
 n'), ('가', 'Josa'), ('높아지며', 'Verb'), ('상승', 'Noun'), ('한', 'Josa'),  
 ('점', 'Noun'), ('은', 'Josa'), ('한국', 'Noun'), ('증시', 'Noun'), ('에', 'Jos  
 a'), ('우호', 'Noun'), ('적', 'Suffix'), ('이다', 'Josa'), ('더불어', 'Verb'),  
 ('그동안', 'Noun'), ('부진한', 'Adjective'), ('모습', 'Noun'), ('을', 'Josa'),  
 ('보여', 'Verb'), ('왔던', 'Verb'), ('반도체', 'Noun'), ('업종', 'Noun'), ('이',  
 'Josa'), ('정부', 'Noun'), ('의', 'Josa'), ('지', 'Modifier'), ('원', 'Modifier'),  
 ('책', 'Noun'), ('이', 'Josa'), ('발표', 'Noun'), ('되며', 'Verb'), ('강세', 'Nou  
 n'), ('를', 'Josa'), ('보인', 'Verb'), ('정도', 'Noun'), ('긍정', 'Noun'), ('적',  
 'Suffix'), ('이다', 'Josa'), ('이런', 'Adjective'), ('가운데', 'Noun'), ('미국',  
 'Noun'), ('연', 'Modifier'), ('준', 'Noun'), ('은', 'Josa'), ('를', 'Noun'), ('통  
 해', 'Noun'), ('시장', 'Noun'), ('우려', 'Noun'), ('보다는', 'Josa'), ('덜', 'Nou  
 n'), ('매', 'Modifier'), ('파적', 'Noun'), ('인', 'Josa'), ('내용', 'Noun'),  
 ('을', 'Josa'), ('발표', 'Noun'), ('한', 'Josa'), ('정도', 'Noun'), ('우호', 'Nou  
 n'), ('적', 'Suffix'), ('이나', 'Josa'), ('파월', 'Noun'), ('의장', 'Noun'),  
 ('이', 'Josa'), ('매', 'Noun'), ('회의', 'Noun'), ('때', 'Noun'), ('마다', 'Jos  
 a'), ('금리', 'Noun'), ('를', 'Josa'), ('인상', 'Noun'), ('할', 'Verb'), ('수', 'N  
 oun'), ('있', 'Adjective'), ('음', 'Noun'), ('을', 'Josa'), ('언급', 'Noun'), ('하  
 고', 'Josa'), ('양적', 'Noun'), ('긴축', 'Noun'), ('을', 'Josa'), ('이전', 'Nou

In [16]:

```
#여기서 명사,형용사,부사만 골라낼거다.
# *sentence는 for 함수안에서만 쓰이는 함수라 위에꺼랑은 상관없음.
#morph라는 리스트를 만들었고, 이 리스트 안에 쉼표로 구분된 두 가지 데이터가 담겨있다.
#ex) ('주식','Noun')
#if 태그 안에 명사,형용사,부사가 있으면 내가 만든 빈 리스트 (noun_adj_adv_list)에 앞의 단어 (word)를
# 추가(append)한다. 워드 클라우드에 필요없는 '것','나','내' 는 제외.
```

```
noun_adj_adv_list=[]

for sentence in morphs :
    for word, tag in sentence :
        if tag in ['Noun'] and ("것" not in word) and ("내" not in word)and ("나" not in word)and ("수"not in word) and("게"not in word)and("말"not in word):
            noun_adj_adv_list.append(word)

print(noun_adj_adv_list)
#내가 만든 조건에 충족하는 word만 출력된 리스트다.
```

['컨텐츠', '로', '케이', '터', '금융', '홈', '뉴스', '뉴스', '로', '케이', '터', '뉴스', '타이틀', '우측', '티', '영역', '속보', '가상', '화폐', '보관', '뱅크', '지난해', '영업', '익', '군인공제회', '김해', '복합', '스포츠', '레저', '품', '투자', '증권', '신임', '맥', '쿼리', '올해', '인플레', '투자자', '아이', '테크놀로지', '영업', '손실', '어룡', '대신', '금융', '그룹', '회장', '괴', '디스플레이', '불투명', '넷마블', '블록', '체인', '메타', '버스', '도전', '라씨', '상', '대신', '급락', '엔솔', '로케이션', '바', '추가', '인쇄', '글자', '크기', '글꼴', '본문', '텍스트', '본문', '텍스트', '로케이션', '바', '추가', '뷰', '언론사', '로고', '언론사', '로고', '기사', '제목', '재일', '기사', '제목', '굿모닝', '증시', '솔', '데뷔', '변동성', '확대', '파적', '파월', '외국인', '기사', '제목', '기사', '입력', '기사', '원문', '후속', '기사', '평가', '공감', '후속', '기사', '댓글', '표기', '숫자일', '때', '클래스', '추가', '댓글', '기사', '상단', '요점', '기사', '본문', '증시', '기업', '공개', '사상', '최대', '에너지', '솔루션', '코스피', '시장', '데뷔', '변동성', '확대', '전반', '투자', '심리', '외측', '점', '감안', '때', '매물', '회', '가능성', '하락', '출발', '예상', '다만', '저가', '유입', '전망', '미국', '뉴욕증시', '제롬', '파월', '연방', '준비', '제도', '준', '의장', '금리', '인상', '예고', '약세', '날', '뉴욕증시', '다우존스', '산업', '평균', '전장', '포인트', '거래', '스탠더드', '앤드', '어스', '포인트', '하락', '기술', '주', '중심', '포인트', '오른', '장', '마감', '연방', '공개시장', '위원회', '정례', '회의', '종료', '직후', '포인트', '이상', '다우', '이후', '파월', '의장', '기자회견', '노동시장', '위협', '금리', '인상', '여지', '패', '언급', '직후', '하락', '반전', '외환', '중개', '업체', '에드워드', '모야', '투자자', '노트', '통해', '파월', '의장', '회견', '직후', '더', '금리', '인상', '가능성', '점', '방송', '전', '준', '금리', '인상', '시작', '후', '대차대조표', '축소', '방침', '공표', '투자', '심리', '부정', '영향', '준', '준', '매파', '통화', '긴축', '선호', '움직임', '국채', '금리', '금리', '기술', '주', '더', '부담', '서상영', '미래에셋', '증권', '연구원', '미국', '증시', '본격', '실적', '시즌', '대한', '기대', '심리', '상승', '점', '한국', '증시', '우호', '그동안', '모습', '반도체', '업종', '정부', '책', '발표', '강세', '점도', '긍정', '가운데', '미국', '준', '를', '통해', '시장', '우려', '덜', '파적', '발표', '점도', '우호', '파월', '의장', '매', '회의', '때', '금리', '인상', '음', '언급', '양적', '긴축', '이전', '더', '진행', '주장', '점', '부담', '외국인', '부정', '이기', '때문', '테슬라', '실적', '발표', '불구', '공급망', '문제', '향후', '부담', '요인', '주장', '매물', '회', '시간', '외로', '하락', '등', '가이던스', '대한', '우려', '부각', '점도', '부담', '감안', '한국', '증시', '파적', '준', '대한', '우려', '달러', '환율', '등', '원화', '폭', '확대', '등', '전반', '투자', '심리', '외측', '점', '감안', '때', '매물', '회', '가능성', '다만', '최근', '하락', '저가', '기대', '또한', '배제', '시장', '하락', '출발', '후', '외국인', '변화', '예상', '한지역', '증권', '연구원', '성명', '상', '금리', '인상', '맞', '그', '이후', '양적', '긴축', '논의', '이전', '구체화', '점', '시장', '예상', '이후', '파월',

'의장', '기자회견', '예상', '더', '파적', '점', '시장', '혼란', '재차', '유발', '고용', '시장', '저해', '채', '금리', '인상', '여지', '매', '회의', '때', '금리', '인상', '배제', '언급', '점', '양적', '긴축', '과거', '조기', '속도', '언급', '등', '월', '중순', '상원', '청문회', '발언', '때', '매파', '성향', '모습', '과거', '년도', '당시', '긴축', '사이클', '고용', '경제', '물가', '준', '자산', '규모', '등', '환경', '때문', '준', '입장', '이전', '정책', '정상화', '자체', '증시', '부담', '사실', '이', '정책', '가속', '화의', '충격', '이후', '일정', '부분', '시장', '반영', '예측', '양적', '긴축', '단행', '점', '주목', '필요', '향후', '준', '시장', '지속', '커뮤니케이션', '통해', '최근', '긴축', '발작', '시장', '충격', '최소', '시사', '이', '만큼', '당분간', '준발', '불확실', '성은', '완화', '전망', '의사록', '발표', '이후', '일', '시장', '해석', '과정', '불확실', '재차', '확대', '판단', '날', '증시', '미국', '발', '악재', '영향', '악세', '압력', '전망', '미국', '장', '마감', '후', '테슬라', '어닝', '서프라이즈', '기록', '공급', '문제', '연중', '지속', '언급', '함', '시간', '외', '주가', '하락', '세', '보이', '정도', '장주', '투자', '심리', '영향', '줄', '전망', '더', '증시', '에너지', '솔루션', '상장', '함', '해당', '종목', '및', '관련', '주', '시가총액', '상', '위주', '변동성', '확대', '예상', '에너지', '솔루션', '코스피', '편입', '단행', '날', '해당', '종목', '주가', '변화', '코스피', '비중', '레인지', '추정', '편입', '기관', '변화', '시장', '전반', '방향성', '전망', '아시아', '경제', '관련', '뉴스', '해당', '언론사', '선정', '언론사', '페이지', '아웃', '링크', '로', '이동해', '볼', '역술', '박세리', '남편', '자리', '자식', '도경완', '앞', '장윤정', '이상형', '밥', '김희철', '초', '결혼식', '비혼', '선언', '돌연', '결혼', '계획', '발표', '비아그라', '효과', '배', '몇', '분', '뒤', '길거리', '컵', '보증금', '장사', '언제', '바이라인', '이선', '애', '기자', '카드', '이선', '애', '기록', '기억', '관찰', '확인', '전달', '기록', '동학', '서학', '코인', '개미', '피', '눈물', '패닉', '셀링', '공포', '이제', '시작', '종합', '엔솔', '개장', '거래', '외국인', '매도', '폭탄', '급락', '카피라이트', '아시아', '경제', '무단', '재', '및', '재', '배포', '금지', '프로모션', '임인년', '토정비결', '신년', '운세', '빅데이터', '인공', '지능', '핀', '테크', '취업', '계', '국비', '교육', '후속', '기사', '목록', '보기', '뉴스', '기사', '제공', '인쇄', '메일', '아시아', '경제', '기사', '제공', '뷰', '이', '언론사', '기사', '뉴스', '제휴', '평가', '위원회', '뉴스', '제휴', '및', '제재', '심사', '규정', '노출', '중단', '기사', '이전', '페이지', '목록', '보기', '묶음', '뉴스', '일', '경우', '추가', '필요', '묶음', '뉴스', '묶음', '뉴스', '증시', '뉴스', '오전', '금리', '인상', '예고', '와르르', '코스피', '악', '개월', '선', '외환', '브리핑', '금리', '인상', '예고', '파월', '환율', '굿모닝', '증시', '솔', '데뷔', '변동성', '확대', '매', '파월', '금리', '인상', '예고', '변동성', '확', '모닝', '브리핑', '뉴욕증시', '금리', '인상', '예고', '혼', '일일', '펀드', '동향', '채권', '펀드', '하루', '금리', '인상', '예고', '투심', '변동', '이전', '페이지', '위로', '컨텐츠', '우측', '메뉴', '널기', '개', '뉴스', '홈', '배너', '등록', '경우', '경우', '가장', '뉴스', '더', '보기', '주식', '매매', '상위', '종목', '및', '환율', '경제지표', '음악', '저작권', '투자', '플랫폼', '유직', '증권', '업', '로', '보어', '드', '바이', '저', '사', '지난해', '자문', '일임', '계약', '돌파', '투자', '대안', '부각', '스톡옵션', '대박', '네이버', '카카오', '직원', '젠', '손실', '걱정', '파월', '물가', '더', '데', '일리', '국제', '금융시장', '유럽', '증시', '결과', '상승', '마감', '첫', '월급', '비트코인', '뉴욕시', '왜', '웃음', '거리', '가장', '뉴스', '개', '뉴스', '홈', '배너', '등록', '경우', '경우', '인기', '검색어', '더', '보기', '인기', '검색어', '에너지', '솔루션', '삼성', '전자', '카카오', '셀트리온', '하이닉스', '인기', '검색어', '우측', '메뉴', '널기']

In [17]:

```
#빈도수대로 정렬
#counter는 카운팅 해주는 패키지.
#이를 내가 만든 최종 리스트(noun_adj_adv_list)에 적용시킨다.
count = Counter(noun_adj_adv_list)
```

In [18]:

```
#단어와 빈도수가 함께 나오는 딕셔너리 생성.  
words = dict(count.most_common())  
words
```

Out[18]:

```
{'뉴스': 15,  
 '금리': 15,  
 '기사': 14,  
 '인상': 13,  
 '증시': 11,  
 '시장': 11,  
 '파월': 9,  
 '점': 9,  
 '준': 9,  
 '더': 8,  
 '투자': 7,  
 '하락': 7,  
 '긴축': 7,  
 '확대': 6,  
 '때': 6,  
 '예고': 6,  
 '언론사': 5,  
 '변동성': 5.
```

In [19]:

```
pip install nltk
```

Requirement already satisfied: nltk in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (3.6.7)

Requirement already satisfied: regex>=2021.8.3 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from nltk) (2022.1.18)

Requirement already satisfied: tqdm in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from nltk) (4.62.3)

Requirement already satisfied: click in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from nltk) (8.0.3)

Requirement already satisfied: joblib in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from nltk) (1.1.0)

Requirement already satisfied: colorama in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from click->nltk) (0.4.4)

Note: you may need to restart the kernel to use updated packages.



In [20]:

```
#워드클라우드 생성하기 위해 패키지 설치
#nltk는 형태소 분석 패키지
from wordcloud import WordCloud
import matplotlib.pyplot as plt
import nltk
from nltk.corpus import stopwords

#밑에는 그래프의 디스플레이 설정에 관한 코드
%matplotlib inline
import matplotlib
from IPython.display import set_matplotlib_formats
matplotlib.rc('font',family = 'NanumGothic')

set_matplotlib_formats('retina')
# 제일 밑에있는거는 그래프에 음수의 수치를 보여줄 때 오류가 나는 것을 방지하는 코드
matplotlib.rc('axes',unicode_minus = False)
```

In [21]:

```
wordcloud = WordCloud(font_path='NanumGothic',
background_color='white',colormap = "Accent_r",
width=1500,
height=1000).generate_from_frequencies(words)

plt.imshow(wordcloud)
plt.axis('off')
plt.show()
```



In [22]:

```
!pip install pillow
```

Requirement already satisfied: pillow in c:\Users\Wyj\anaconda3\envs\pybook\lib\site-packages (8.4.0)

In [23]:

```
# 구글에 wordcloud mask image

import numpy as np
from PIL import Image
from wordcloud import STOPWORDS

starwars=np.array(Image.open("starwars.png"))

stopwords=set(STOPWORDS)
stopwords.add("said")
```

In [24]:

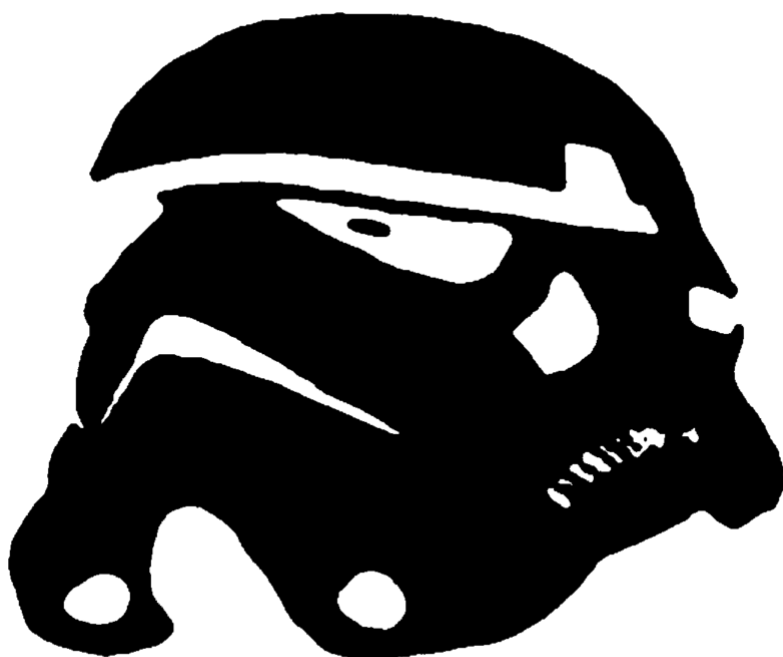
```
starwars
```

Out[24]:

```
array([[255, 255, 255, ..., 255, 255, 255],
       [255, 255, 255, ..., 255, 255, 255],
       [255, 255, 255, ..., 255, 255, 255],
       ...,
       [255, 255, 255, ..., 255, 255, 255],
       [255, 255, 255, ..., 255, 255, 255],
       [255, 255, 255, ..., 255, 255, 255]], dtype=uint8)
```

In [25]:

```
plt.figure(figsize=(8,8))
plt.imshow(starwars, cmap=plt.cm.gray, interpolation='bilinear')
plt.axis("off")
plt.show()
```



```

wc=WordCloud(font_path='NanumGothic',background_color="red", max_words=2000, mask=starwars,stopword:
gen = wc.generate_from_frequencies(words)
plt.figure(figsize=(12,12))
plt.imshow(gen,interpolation='bilinear')
plt.axis("off")
plt.show()

```



위에처럼 LG에너지솔루션 관련 웹크롤링 및 워드클라우드 시각화를 통해  
관련 언급이 파월,인상,금리,하락,증시,긴축 등등이 크게 나온다  
이제 위에 언급한 LG에너지솔루션 종목을 분석해본다.

In [44]:

```
#1. 야후증권 데이터를 불러오기
from pandas_datareader import data as pdr #야후증권 데이터를 요청하기 위한 pandas_datareader
import yfinance as yf #야후증권 데이터 모듈을 임포트
yf.pdr_override() #야후증권 데이터사용을 위한 허가

#2. 증권 데이터 요청시 종목명/종목코드를 불러오기
#ex)국내주식변수=pdr_data_yahoo('종목코드',start='시작일',end='종료일')

#국내종목은 6자리 종목코드 뒤에 .코스피(KS) 코스닥이면(KQ) EX)삼성전자는 005930.KS로 조회.
#LG에너지솔루션의 종목코드를 이용해 데이터 불러오기.
LG=pdr.get_data_yahoo('373220.KS', start='2022-01-26')

print(LG)

#open:장 시작가격 high:장중 상한가 low:장중 하한가 close:장 종료가격 volume:거래량 adj close: 수정종
```

```
[*****100%*****] 1 of 1 completed
      Open      High      Low      Close  Adj Close   Volume
Date
2022-01-27  597000.0  598000.0  450000.0  505000.0   505000.0  15790742
```

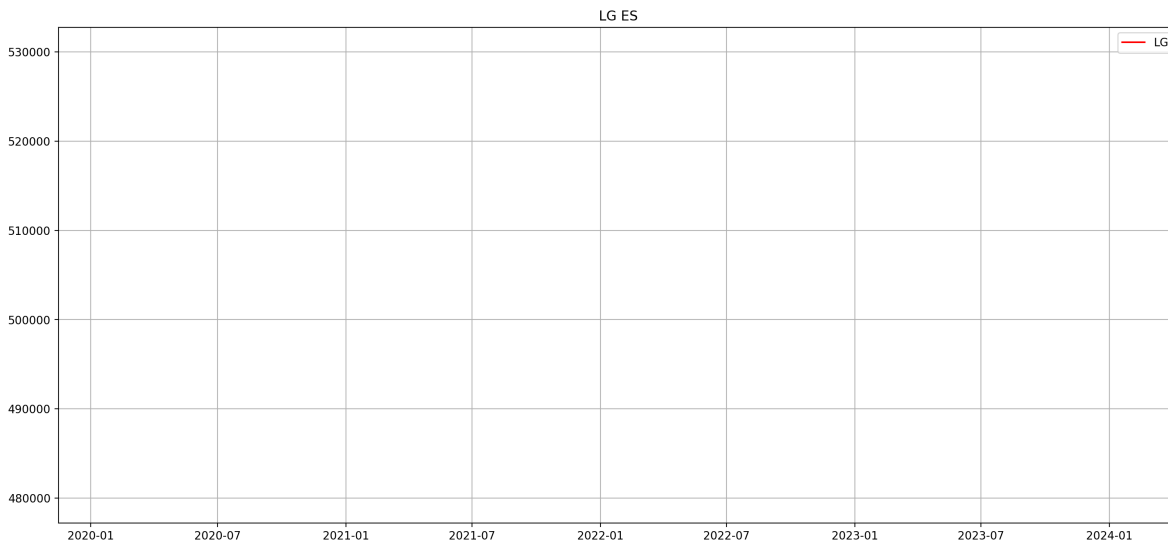
**위에 보면 오늘 상장하다보니 아침 시작가는 597000원이고 장 종료가  
격은 505000원으로 끝남.**

**이거를 시각화해봄.**

In [52]:

```
import matplotlib.pyplot as plt #데이터 시각화
plt.figure(figsize=(18,8))
#plot(x,y,마커 형태,label='')
#plot(x좌표는 현대 데이터프레임의 날짜 인덱스, y좌표는 현대 데이터프레임의 종가(close), 마커형태는 볼
plt.plot(LG.index, LG.Close, 'r', label='LG')

plt.title('LG ES')
plt.legend(loc='best') #범례표시( loc=best하면 제일 좋은자리에 표시)
plt.grid(True) # 배경 점선
plt.show()
#open:장 시작가격 high:장중 상한가 low:장중 하한가 close:장종료가격 volume:거래량 adj close:수정종
#x좌표는 데이터프레임의 날짜 인덱스, y좌표는 종가데이터, 색상, loc=best는 그래프가 표시되지 않는 부분을
```



위에 보면 lg에너지솔루션이 22/1/27일 상장하다보니 아직 차트에 안나옴.

그래서 같은 2차전지 업종인 삼성sdi의 주가를 불러와봄.

In [53]:

```
from pandas_datareader import data as pdr
import yfinance as yf #야후증권 데이터 모듈을 임포트
yf.pdr_override() #야후증권 데이터사용을 위한 허가
```

```
samsung=pdr.get_data_yahoo('006400.KS', start='2021-12-01')
```

```
print(samsung)
```

```
#open:장 시작가격 high:장중 상한가 low:장중 하한가 close:장 종료가격 volume:거래량 adj close: 수정종
```

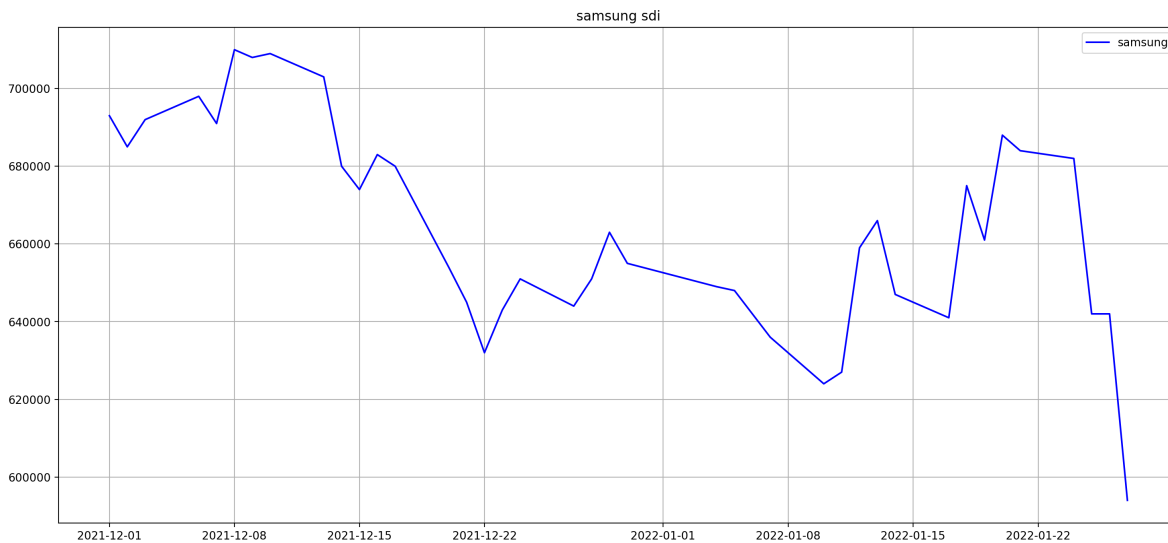
```
[*****100%*****] 1 of 1 completed
```

	Open	High	Low	Close	Adj Close	Volume
Date						
2021-12-01	694000.0	698000.0	680000.0	693000.0	693000.0	226286
2021-12-02	688000.0	696000.0	680000.0	685000.0	685000.0	291867
2021-12-03	682000.0	699000.0	681000.0	692000.0	692000.0	260528
2021-12-06	692000.0	703000.0	683000.0	698000.0	698000.0	162519
2021-12-07	701000.0	702000.0	684000.0	691000.0	691000.0	172060
2021-12-08	700000.0	719000.0	695000.0	710000.0	710000.0	264783
2021-12-09	714000.0	717000.0	705000.0	708000.0	708000.0	313022
2021-12-10	705000.0	711000.0	697000.0	709000.0	709000.0	183639
2021-12-13	709000.0	720000.0	702000.0	703000.0	703000.0	159194
2021-12-14	697000.0	698000.0	680000.0	680000.0	680000.0	293090
2021-12-15	674000.0	678000.0	661000.0	674000.0	674000.0	327141
2021-12-16	680000.0	691000.0	678000.0	683000.0	683000.0	223939
2021-12-17	671000.0	686000.0	671000.0	680000.0	680000.0	242779
2021-12-20	675000.0	676000.0	653000.0	654000.0	654000.0	352862
2021-12-21	653000.0	654000.0	628000.0	645000.0	645000.0	444824
2021-12-22	646000.0	649000.0	629000.0	632000.0	632000.0	368128
2021-12-23	641000.0	646000.0	638000.0	643000.0	643000.0	259442
2021-12-24	648000.0	662000.0	646000.0	651000.0	651000.0	215205
2021-12-27	651000.0	651000.0	642000.0	644000.0	644000.0	129558
2021-12-28	647000.0	651000.0	640000.0	651000.0	651000.0	229379
2021-12-29	660000.0	667000.0	655000.0	663000.0	663000.0	238556
2021-12-30	657000.0	660000.0	651000.0	655000.0	655000.0	147850
2022-01-04	653000.0	656000.0	645000.0	649000.0	649000.0	209259
2022-01-05	644000.0	650000.0	635000.0	648000.0	648000.0	336895
2022-01-06	642000.0	658000.0	638000.0	642000.0	642000.0	235833
2022-01-07	648000.0	649000.0	635000.0	636000.0	636000.0	458320
2022-01-10	637000.0	637000.0	623000.0	624000.0	624000.0	261638
2022-01-11	640000.0	640000.0	622000.0	627000.0	627000.0	272004
2022-01-12	632000.0	663000.0	632000.0	659000.0	659000.0	437843
2022-01-13	674000.0	674000.0	660000.0	666000.0	666000.0	313767
2022-01-14	654000.0	665000.0	643000.0	647000.0	647000.0	245855
2022-01-17	646000.0	658000.0	638000.0	641000.0	641000.0	257927
2022-01-18	642000.0	682000.0	640000.0	675000.0	675000.0	487240
2022-01-19	676000.0	680000.0	659000.0	661000.0	661000.0	320317
2022-01-20	657000.0	692000.0	652000.0	688000.0	688000.0	375814
2022-01-21	687000.0	691000.0	669000.0	684000.0	684000.0	261918
2022-01-24	676000.0	685000.0	667000.0	682000.0	682000.0	281941
2022-01-25	674000.0	676000.0	640000.0	642000.0	642000.0	463077
2022-01-26	642000.0	642000.0	642000.0	642000.0	642000.0	0
2022-01-27	637000.0	646000.0	590000.0	594000.0	594000.0	593544

In [54]:

```
import matplotlib.pyplot as plt #데이터 시각화
plt.figure(figsize=(18,8))
#plot(x,y,마커 형태,label='')
#plot(x좌표는 현대 데이터프레임의 날짜 인덱스, y좌표는 현대 데이터프레임의 종가(close), 마커형태는 불
plt.plot(samsung.index,samsung.Close, 'b', label='samsung')

plt.title('samsung sdi')
plt.legend(loc='best') #범례표시( loc=best하면 제일 좋은자리에 표시)
plt.grid(True) # 배경 점선
plt.show()
#open:장 시작가격 high:장중 상한가 low:장중 하한가 close:장종료가격 volume:거래량 adj close:수정종
#x좌표는 데이터프레임의 날짜 인덱스, y좌표는 종가데이터, 색상, loc=best는 그래프가 표시되지 않는 부분을
```



일단 차트를 보면 오늘 지수도 많이 떨어지다보니 삼성sdi도 많이 떨어  
진걸 볼수있음.

이제 좀 더 편하게 분석하기위해 캔들 차트로 변경해봄.

In [55]:

```
pip install --upgrade mplfinance
```

Requirement already satisfied: mplfinance in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (0.12.8b6)  
 Requirement already satisfied: matplotlib in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from mplfinance) (3.5.1)  
 Requirement already satisfied: pandas in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from mplfinance) (1.3.5)  
 Requirement already satisfied: python-dateutil>=2.7 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from matplotlib->mplfinance) (2.8.2)  
 Requirement already satisfied: packaging>=20.0 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from matplotlib->mplfinance) (21.3)  
 Requirement already satisfied: cyclor>=0.10 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from matplotlib->mplfinance) (0.11.0)  
 Requirement already satisfied: fonttools>=4.22.0 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from matplotlib->mplfinance) (4.28.5)  
 Requirement already satisfied: pyparsing>=2.2.1 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from matplotlib->mplfinance) (3.0.6)  
 Requirement already satisfied: kiwisolver>=1.0.1 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from matplotlib->mplfinance) (1.3.2)  
 Requirement already satisfied: numpy>=1.17 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from matplotlib->mplfinance) (1.21.5)  
 Requirement already satisfied: pillow>=6.2.0 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from matplotlib->mplfinance) (8.4.0)  
 Requirement already satisfied: six>=1.5 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from python-dateutil>=2.7->matplotlib->mplfinance) (1.16.0)  
 Requirement already satisfied: pytz>=2017.3 in c:\Users\Wyj\Anaconda3\envs\Wpybook\lib\site-packages (from pandas->mplfinance) (2021.3)  
 Note: you may need to restart the kernel to use updated packages.

엠피엘파이낸스의 가장 큰 장점은 OHLC데이터 칼럼과 날짜시간 인덱스(datetimeindex)를 포함한 데이터 프레임만 있으면 기존에 사용자들이 수동으로 처리했던 데이터 변환 작업을 모두 자동화해줌.

```
import mplfinance as mpf
```

```
mpf.plot(OHLC 데이터프레임[,title=제목] [,type=차트형태][,mav=이동평균선][,volume=거래량 표시][,ylabel=y축레이블])
```



In [56]:

```

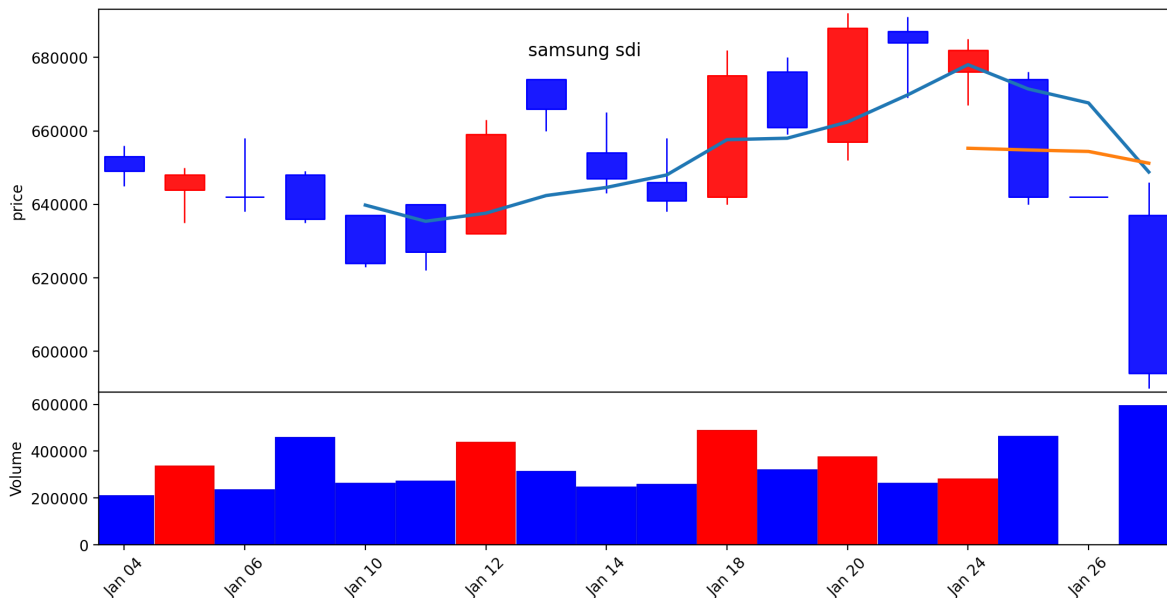
from pandas_datareader import data as pdr
import matplotlib.pyplot as plt
import mplfinance as mpf
import yfinance as yf
#국내종목은 6자리 종목코드 뒤에 .코스피(KS) 코스닥이면(KQ) EX)삼성전자는 005930.KS로 조회.

samsung=pdr.get_data_yahoo('006400.KS', start='2022-01-01')

#make_marketcolors로 차트의 색상들을 지정할수있음.
#wick는 하루의 고가,저가를 나타내는 선의 색 지정.
#inherit는 거래량의 색상도 차트색상과 동일하게변경
mc=mpf.make_marketcolors(up='r',down='b',wick={'up': 'red', 'down': 'blue'})
,inherit=True)
s=mpf.make_mpf_style(marketcolors=mc)
#피그라티오는 그래프의 크기 설정, 타이트-레이아웃은 좌우 공백 제거
mpf.plot(samsung,type='candle',mav=(5,15,60),volume=True,title='samsung sdi',ylabel='price',xrotati
style=s)

```

[\*\*\*\*\*100%\*\*\*\*\*] 1 of 1 completed



위에 보면 알겠지만 이동평균선 5일선과 20일선을 깨버리고 바닥으로 내려옴

다음주 추가하락 불가피

In [51]:

```

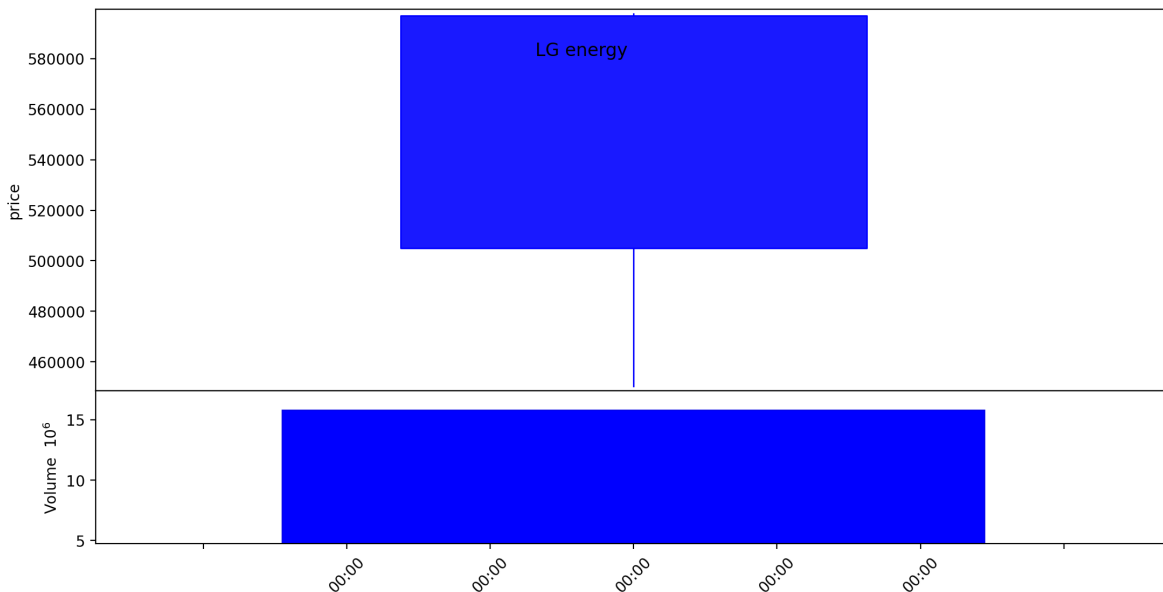
from pandas_datareader import data as pdr
import matplotlib.pyplot as plt
import mplfinance as mpf
import yfinance as yf
#국내종목은 6자리 종목코드 뒤에 .코스피(KS) 코스닥이면(KQ) EX)삼성전자는 005930.KS로 조회.

LG=pdr.get_data_yahoo('373220.KS', start='2022-01-01')

#make_marketcolors로 차트의 색상들을 지정할수있음.
#wick는 하루의 고가,저가를 나타내는 선의 색 지정.
#inherit는 거래량의 색상도 차트색상과 동일하게변경
mc=mpf.make_marketcolors(up='r',down='b',wick={'up': 'red', 'down': 'blue'})
,inherit=True)
s=mpf.make_mpf_style(marketcolors=mc)
#피그래티오는 그래프의 크기 설정, 타이트-레이아웃은 좌우 공백 제거
mpf.plot(LG,type='candle',mav=(5,15,60),volume=True,title='LG energy',ylabel='price',xrotation=45,
style=s)

```

[\*\*\*\*\*100%\*\*\*\*\*] 1 of 1 completed



위에 lg에너지솔루션이 그냥 차트는 안나오지만 캔들 차트는 나옴

역시나 하락이다 . 처음에 lg에너지솔루션 키워드를 분석한결과

파월,금리,인상 등등 을 통해 주식이 하락한걸로 결론을 도출가능함.

In [ ]: