**오늘의 종목 : 어머 이건 꼭 사야해!**

**20150514 김건우**

|  |
| --- |
| 1. 시가 총액 상위 50위 KOSPI / KOSDAQ 시장 분석:  - BeautifulSoup, requests 모듈 사용, re정규식을 통한 정보 추출 |
| 2. 원하는 종목 정보 제공 (데이터 시각화)  -matplotlib, pandas, numpy를 이용 |
| 3. 20거래일간 상승률 및 기관 순매수, 외인 순매수 정렬  -lambda, sort함수를 이용, pandas를 통한 DataFrame 및 axes객체 생성 후 시각화 |

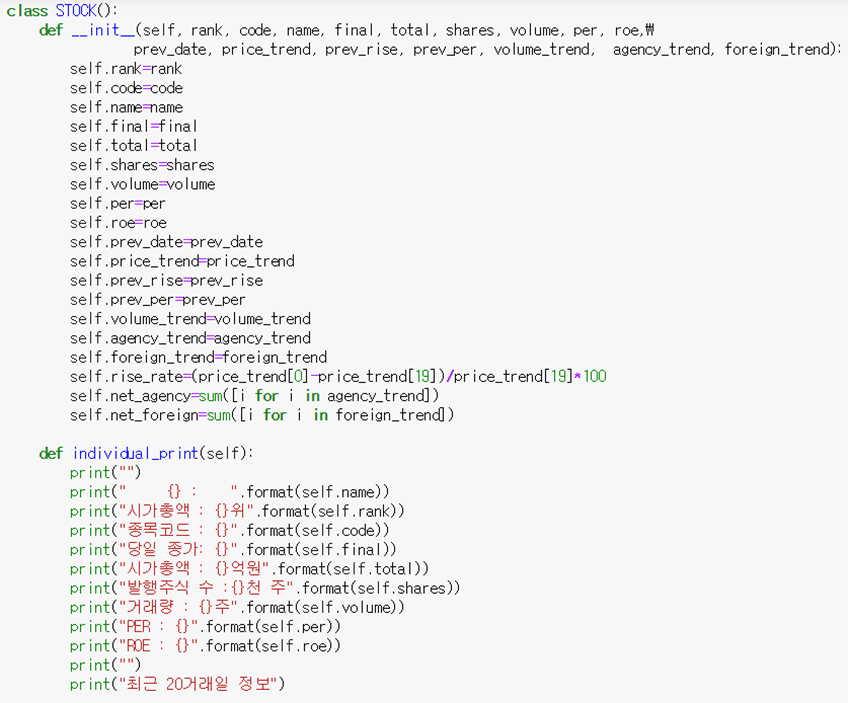
1. **KOSPI/KOSDAQ 시장 분석**

**A. 분석을 위한 데이터 크롤링**



위 사진은 본 프로그램 실행에 필요한 모듈 import와 matplotlib 한글 패치를 위한 코드이다.

**B. class 정의 (STOCK, KOSPI, KOSDAQ 클래스 생성)**



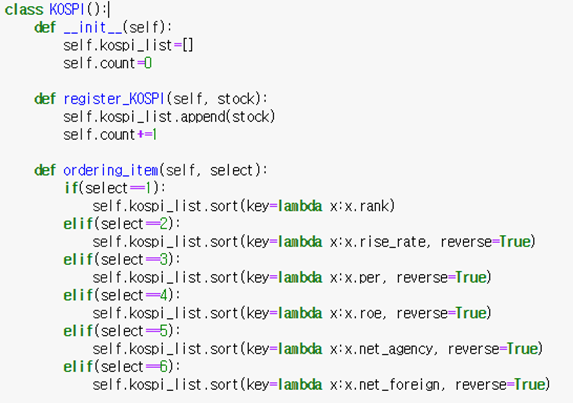
STOCK 클래스는 각 종목에 대한 모든 정보를 담기 위한 클래스이다. 멤버 변수로

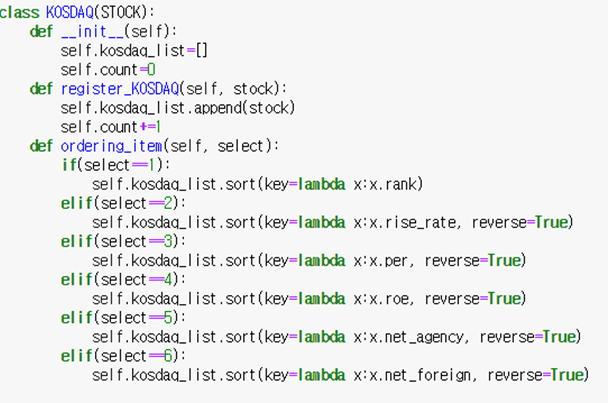
rank : 시가 총액 순위, code : 종목코드, name : 종목 이름 , final : 최근 종가, total : 시가 총액, shares : 발행주식수, volume : 거래량, per : PER(주가수익률), ROE(자기자본이익률), net\_agency : 20 거래일간 기관 순매수량, net\_foreign : 20거래일간 외인 순매수량

리스트로는

prev\_date : 최근 20거래일, price\_trend : 종가 동향, prev\_rise : 전일비, prev\_per : 전일대비 등락률, volume\_trend : 거래량 동향, agency\_trend : 기관 순매수량 동향, foreign\_trend : 외인 순매수량 동향, rise\_rate : 20 거래일간 상승률

이 있다.





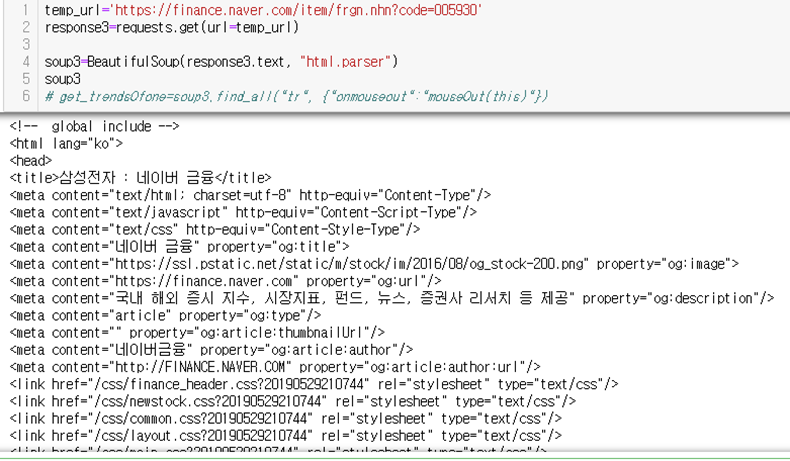
KOSPI와 KOSDAQ 클래스는 STOCK 객체를 담는 클래스로 멤버 함수로 KOSPI50, KOSDAQ50 객체의 리스트에 등록할 수 있는 register함수, 기준을 선택하여 정렬할 수 있는 ordering\_item함수가 있다.



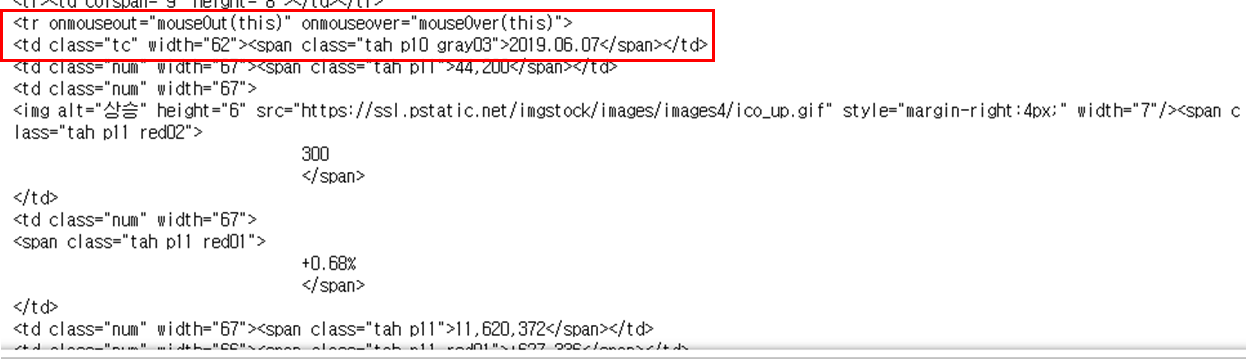
마지막으로 예외 처리를 위한 클래스를 생성하였다.

**C. def trands\_get(code): 코드번호를 인자로 넘겨받아 60 거래일간의 모든 동향을 크롤링하는 함수**

※ BeautifulSoup 모듈을 이용하여 웹페이지를 크롤링하는 법



Bs4모듈을 이용하여 웹페이지를 크롤링하면 위와 같은 텍스트가 생성된다. 이는 해당 페이지 html문서로 이를 분석하기 위해선 위 텍스트를 천천히 살펴보아야 한다.

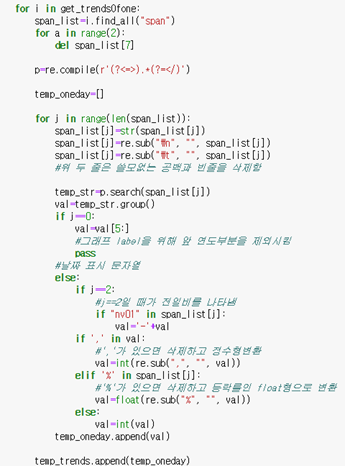


살펴보다 보면, 위 빨간 박스 부분과 같은 규칙성을 찾을 수 있다. 즉, 저 tr태그를 이용하여 한 줄에 있는 정보를 구분하고, onmousout으로 구분되는 부분을 찾을 수 있다. 이를 이용하여 크롤링을 진행해보면 다음과 같다.

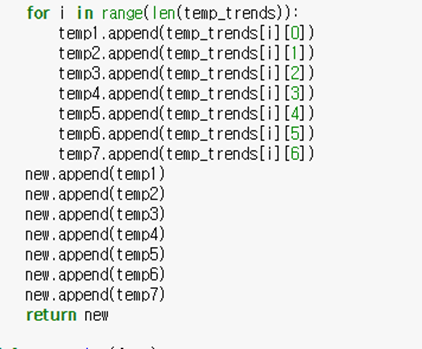
**1.** BeautifulSoup을 이용하면 위 코드에서 soup2에는 해당 페이지에 대한 소스 코드가 텍스트로 담기게 된다. 여기서 find\_all(’tr’)과 같은 식의 명령을 내리면 해당 태그로 구분되는 원소를 리스트에 담아 반환한다. 한 페이지에는 20거래일에 대한 정보가 담겨있기 때문에 60거래일까지 탐색하도록 다음과 같은 코드를 추가하였다.



**2.** 따라서 get\_trendsOfone에는 한 종목에 대해 종가, 전일비, 등락률, 기관.외인 순매수량 등의 정보가 원소로 담겨있는 리스트가 된다. 따라서 이를 각각 추출하기 위해 아래와 같은 정규식 및 문자열 처리 코드를 추가하였다.



**3.** 각각의 정보가 담긴 정보를 하나의 리스트에 모두 담아 리턴한다.

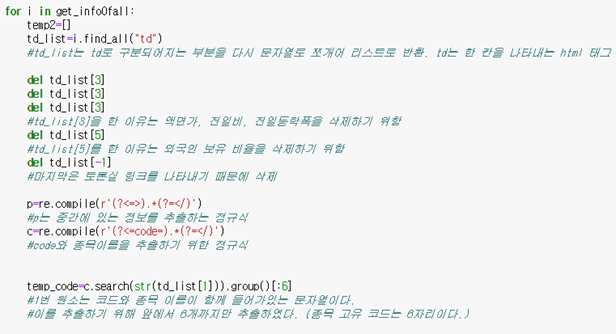


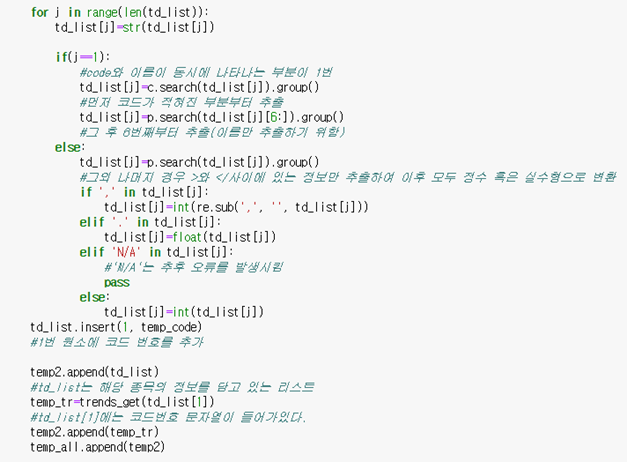
**D. def main(): 네이버 증권 시가총액 순위별로 코스피/코스닥 50종목 정보를 크롤링 (동향이 아니라 기업 정보를 포함)**

**1.** 실행시간을 출력하기 위함



**2.** get\_infoOfall에는 50개의 종목에 대한 기업 정보가 담겨있음. 이를 ‘td’로 구분하여 리스트로 반환한 후에 정보를 추출하는 정규식을 이용하여 리스트에 추가하였다.

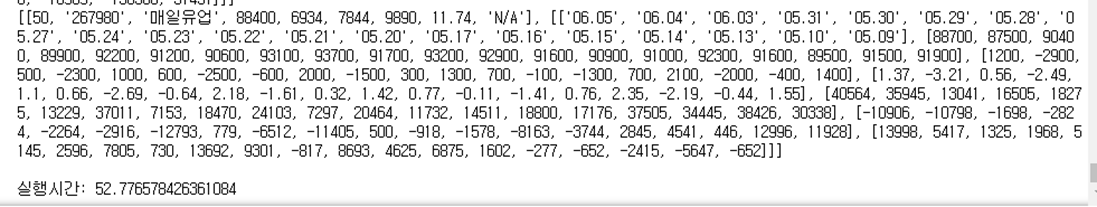




**3.** 모든 정보가 담긴 리스트를 STOCK 객체에 담아 KOSPI50 혹은 KOSDAQ50객체의 리스트에 추가함.

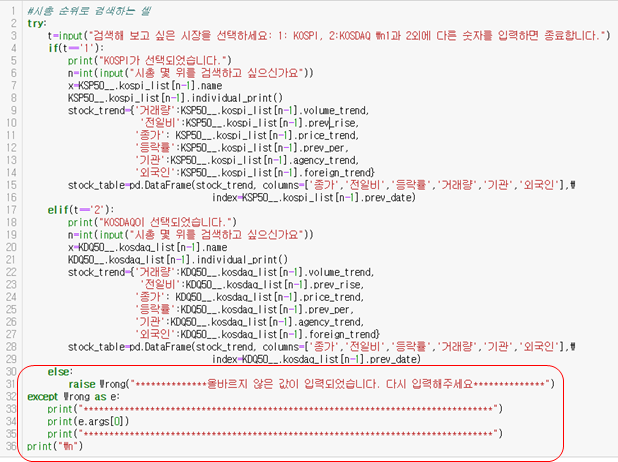


**4.** 각 리스트에 담긴 정보를 출력. 실행 시간 출력

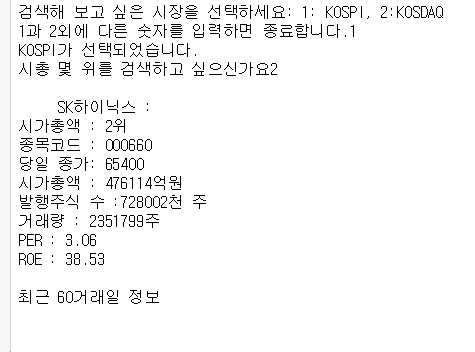


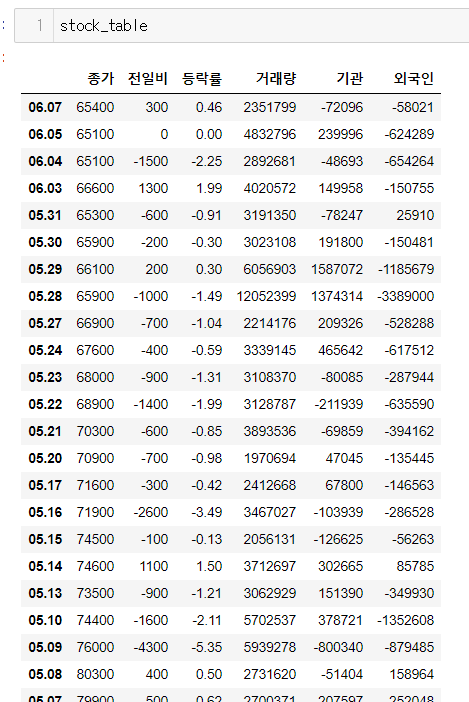
1. **데이터 시각화 : 시가 총액 순위로 검색**

**1.** 시가 총액 순위로 검색하기 위해 stock\_trend라는 사전 객체와 stock\_table이라는 DataFrame객체 생성. 1, 2 이외의 값을 입력할 경우 에러 발생. 예외 처리.



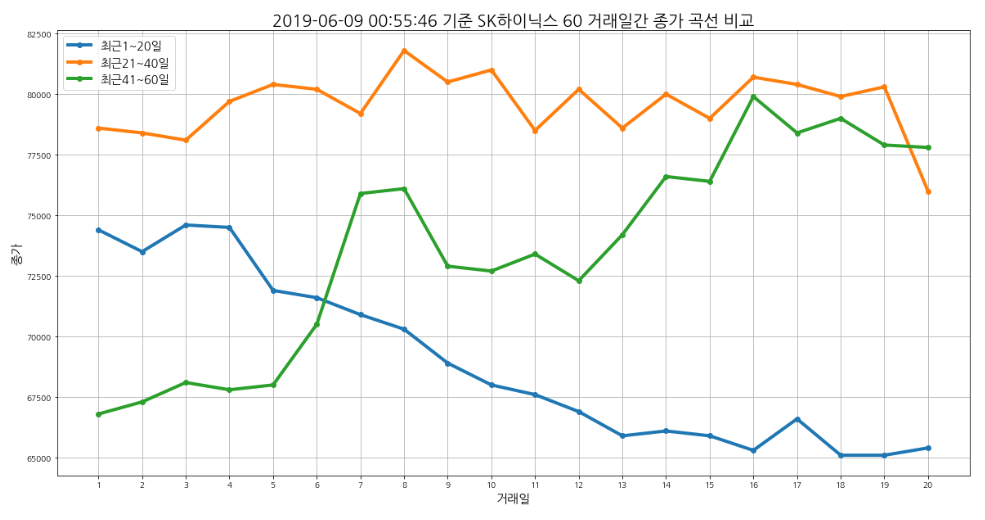
**2.** 시가 총액 순위 입력 및 데이터 시각화





**3.** 위 입력을 바탕으로 해당 기업 주가 동향 20거래일씩 구분한 곡선 출력 및 60거래일간 종가 곡선 출력







1. **데이터 시각화 : 종목 이름으로 검색**

**1.** 100개의 종목 명단을 출력

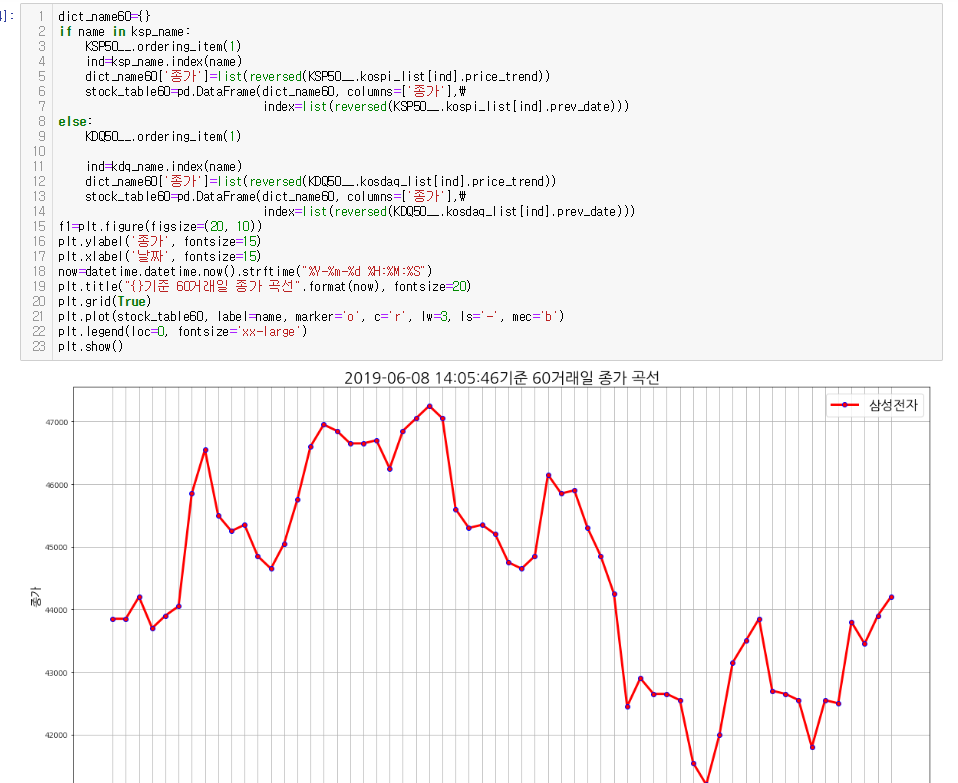


**2.** main실행 시 종목 이름들을 담아놓았던 사전을 이용하여 Data Frame출력

\* 입력한 이름의 종목이 존재하지 않을 경우 KeyError가 발생하여 예외 처리.

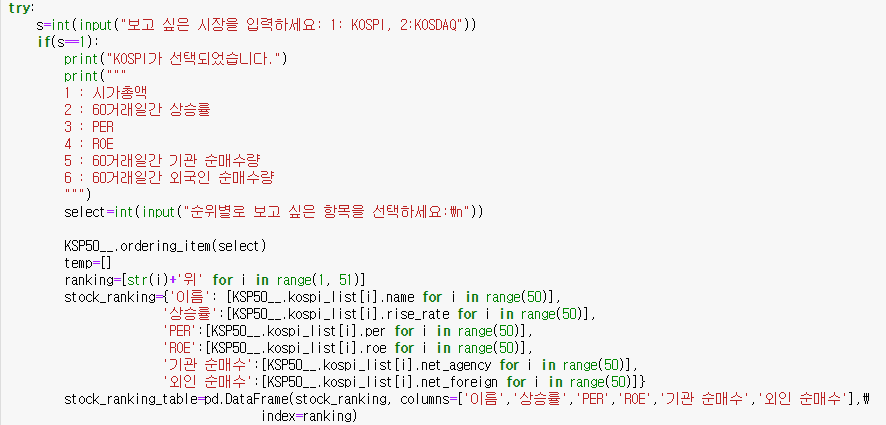


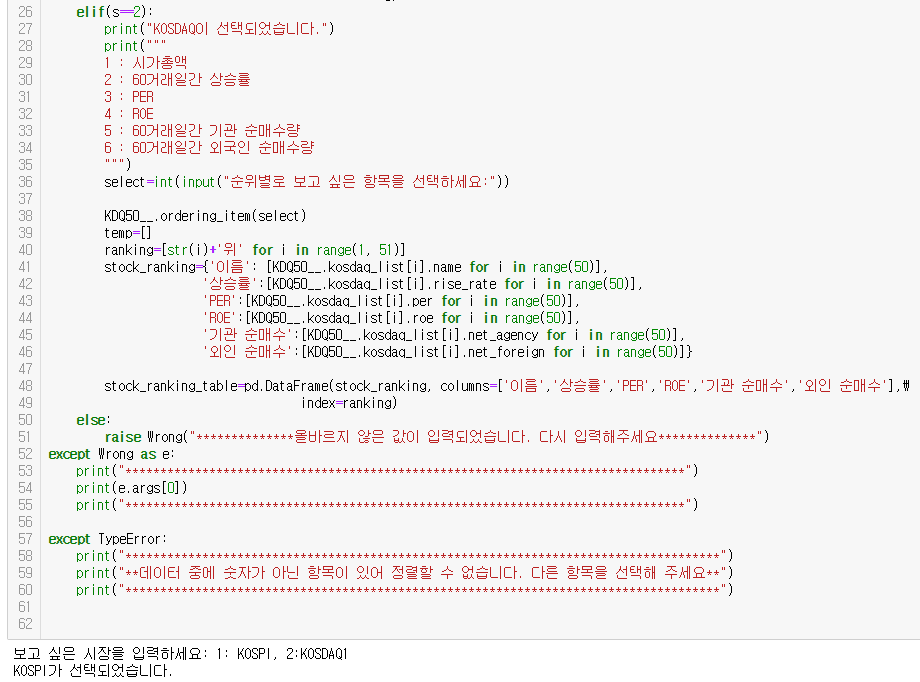


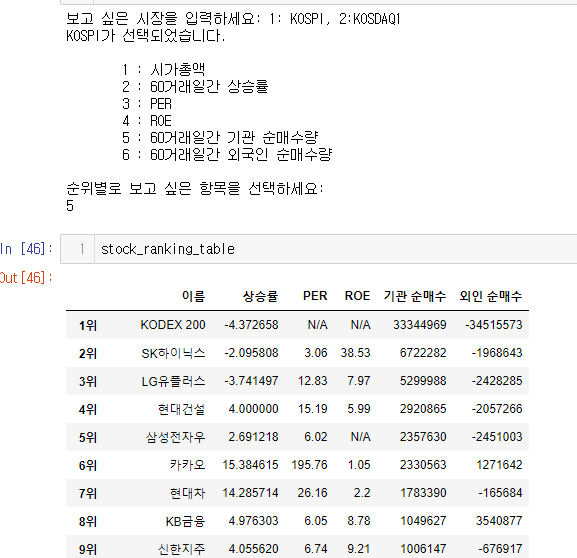


**IV. 데이터 시각화 : 원하는 항목으로 정렬하여 시각화**

**1.** KOSPI / KOSDAQ 중 하나를 입력받은 후 시가총액, 60거래일간 상승률, PER, ROE, 20일간 기관 순매수량, 20거래일간 외국인 순매수량 중 하나를 입력받아 정렬된 데이터를 시각화한다. 마지막 부분에 ROE로 정렬할 경우, ‘N/A’가 포함되어 있어 정렬하지 못하고 TypeError를 발생시켜 이를 처리하였다.. 또한 1~6이 아닌 다른 값이 입력될 경우도 예외 처리 하였다.





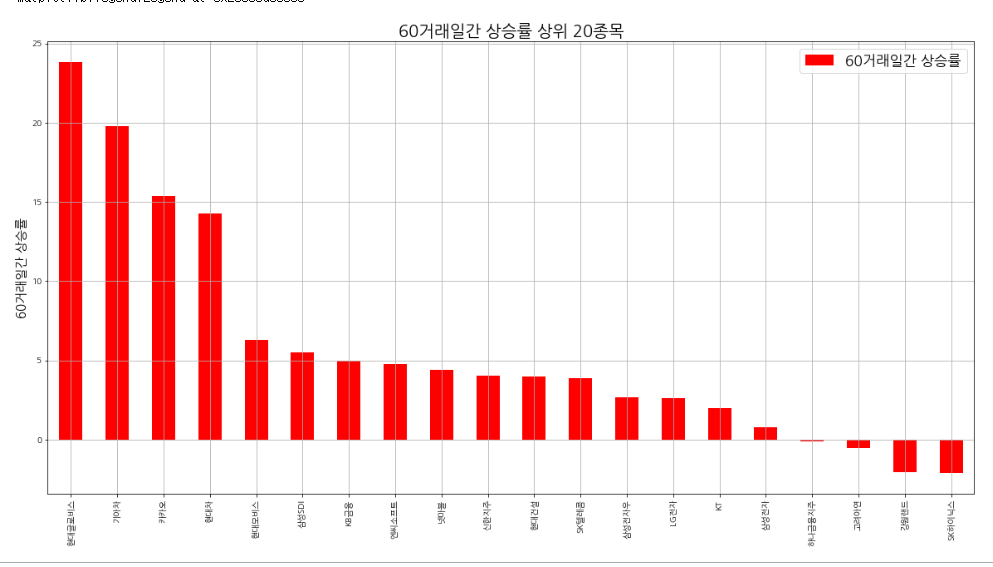


**\*위 그림은 코스피 시장 60일간 기관 순 매수량을 기준으로 정렬한 그림이다.**

**V. 데이터 시각화 : 60일간 상승률 상위 20종목 시각화**

\*최근 상승세를 나타내는 종목을 시가 총액 상위50개의 종목 중에서 보여준다. 해당 셀을 실행하여 상위 20개의 종목 중에서 알고 싶은 종목이 있다면 위의 셀로 돌아가 해당 종목에 대한 정보를 검색하면 될 것이다.





**V-2** : 기간 별 상승률 분석

다음은 60거래일 기준 상승률 20선에 해당하는 종목들을 최근 1~20일간, 21~40일간, 41~60일간으로 나누어 분석하였다. 이 프로그램의 목표는 최종적으로 최근에 상승세인 종목들을 분석하는 것이므로 구간별로 순위를 매겨 따로 그래프를 그리지 않고 60일간 상승률 20선 종목들을 기간별로 분석하였다.



