

한국거래소 상장 주식 데이터 분석

FinanceDataReader

한국 주식가격, 미국주식 가격, 지수, 환율, 암호화폐 가격, 종목 리스팅 등 금융 데이터 수집 라이브러리

```
!pip install -U finance-datareader
```

```
!pip install -U finance-datareader
```

```
!pip install seaborn
```

In [1]:

```
import pandas as pd
import FinanceDataReader as fdr
```

In [2]:

```
#StockListing으로 거래소마다 해당하는 주식 종목의 코드와 각종 정보를 가져와 줍니다.
#KRX, KOSPI, KOSDAQ, KONEX, NYSE, NASDAQ, AMEX, SP500 등
df_krx=fdr.StockListing("KRX")
```

In [3]:

```
df_krx.head()
```

Out[3]:

	Symbol	Market	Name	Sector	Industry	ListingDate	SettleMonth	Representative
0	060310	KOSDAQ	3S	전자부 품 제 조업	반도체 웨이퍼 캐리어	2002-04-23	03월	김세완
1	095570	KOSPI	AJ네 트워 스	산업용 기계 및 장 비 임 대업	렌탈(파 렛트, OA 장비, 건 설장비)	2015-08-21	12월	박대현
2	006840	KOSPI	AK홀 딩스	기타 금융업	지주사업	1999-08-11	12월	채형석, 이석주 (각자 대표이사) http://w
3	054620	KOSDAQ	APS 홀딩 스	기타 금융업	인터넷 트래픽 솔루션	2001-12-04	12월	정기로 http:/
4	265520	KOSDAQ	AP시 스템	특수 목적용 기계 제조업	디스플레 이 제조 장비	2017-04-07	12월	김영주 http

In [4]:

```
#to_csv로 DataFrame을 csv파일로 저장  
df_krx.to_csv("krx.csv", index=False)
```

In [5]:

csv파일을 dataframe으로 읽어보기

pd.read_csv("krx.csv")

Out [5]:

	Symbol	Market	Name	Sector	Industry	ListingDate	SettleMonth	Representative
0	060310	KOSDAQ	3S	전자부 품 제 조업	반도체 웨이퍼 캐리어	2002-04-23	03월	김세완
1	095570	KOSPI	AJ네 트워 스	산업용 기계 및 장 비 임 대업	렌탈(파 렛트, OA 장비, 건 설장비)	2015-08-21	12월	박대현
2	006840	KOSPI	AK홀 딩스	기타 금융업	지주사업	1999-08-11	12월	채형석, 이석주 (각자 대표이사) http
3	054620	KOSDAQ	APS 홀딩 스	기타 금융업	인터넷 트래픽 솔루션	2001-12-04	12월	정기로 h
4	265520	KOSDAQ	AP시 스템	특수 목적용 기계 제조업	디스플레 이 제조 장비	2017-04-07	12월	김영주
...
7890	000545	KOSPI	흥국 화재 우	NaN	NaN	NaN	NaN	NaN
7891	003280	KOSPI	흥아 해운	해상 운송업	외항화물 운송업 (케미컬 탱커)	1976-06-29	12월	이환구
7892	037440	KOSDAQ	희림	건축기 술, 엔 지니어 링 및 관련 기술 서비스 업	설계 및 감리용역	2000-02-03	12월	정영균, 이목운, 허철호 (각자대 표)
7893	419890	KOSPI	히어 로즈 단기 채권 ESG 액티 브	NaN	NaN	NaN	NaN	NaN
7894	238490	KOSDAQ	힘스	특수 목적용 기계 제조업	OLED Mask 인 장기, OLED Mask 검 사기 등	2017-07-20	12월	김주환

7895 rows × 10 columns

In [6]:

```
#데이터분석 pandas, 시각화위해 seaborn
import pandas as pd
import seaborn as sns
```

In [7]:

#주의사항 symbol은 숫자형태지만 앞에 0이 있어 문자임 혹시나 0이 누락되면 숫자로 나올수있으니
df=pd.read_csv("krx.csv",dtype={"Symbol":object})
df.shape

Out[7]:

(7895, 10)

상장회사의 데이터를 통한 여러 데이터 분석

{"회사명":'Name', '종목코드': 'Symbol', '업종': 'Sector', '주요제품': 'Industry', '상장일': 'ListingDate', '결산월': 'SettleMonth', '대표자명': 'Representative', '홈페이지': 'HomePage', '지역': 'Region', }

In [8]:

#데이터를 head를 통해 위에서5개의 데이터 뽑아옴.
df.head()

Out[8]:

	Symbol	Market	Name	Sector	Industry	ListingDate	SettleMonth	Representative
0	060310	KOSDAQ	3S	전자부 품 제 조업	반도체 웨이퍼 캐리어	2002-04-23	03월	김세완
1	095570	KOSPI	AJ네 트워 스	산업용 기계 및 장 비 임 대업	렌탈(파 렛트, OA 장비, 건 설장비)	2015-08-21	12월	박대현
2	006840	KOSPI	AK홀 딩스	기타 금융업	지주사업	1999-08-11	12월	채형석, 이석주 (각자 대표이사) http://w
3	054620	KOSDAQ	APS 홀딩 스	기타 금융업	인터넷 트래픽 솔루션	2001-12-04	12월	정기로 http:/
4	265520	KOSDAQ	AP시 스템	특수 목적용 기계 제조업	디스플레 이 제조 장비	2017-04-07	12월	김영주 http

In [9]:

```
# tail 을 통해 끝에서 5개 데이터 뽑아옴
df.tail()
```

Out[9]:

	Symbol	Market	Name	Sector	Industry	ListingDate	SettleMonth	Representative
7890	000545	KOSPI	흥국화재우	NaN	NaN	NaN	NaN	NaN
7891	003280	KOSPI	흥아해운	해상운송업	외항화물운송업 (케미컬탱커)	1976-06-29	12월	이환구
7892	037440	KOSDAQ	희림	건축기술, 엔지니어링 및 관련 기술 서비스 업	설계 및 감리용역	2000-02-03	12월	정영균, 이목운, 허철호 (각자대 표)
7893	419890	KOSPI	히어로즈단기채권ESG액티브	NaN	NaN	NaN	NaN	NaN
7894	238490	KOSDAQ	힘스	특수목적용기계제조업	OLED Mask 인장기, OLED Mask 검사기 등	2017-07-20	12월	김주환 ht

In [10]:

#sample를 사용하면 랜덤하게 ()안의 갯수만큼 가져옴.

df.sample(5)

Out[10]:

Symbol	Market	Name	Sector	Industry	ListingDate	SettleMonth	Representa
3872	52H939	KOSPI	H939KOSPI200 미래 콜	NaN	NaN	NaN	NaN
7664	009540	KOSPI	한국조선해양	선박, 해 양구조 물, 엔진, 펌프 전 및 보 트 건 조업	1999-08-24	12월	가
7087	57H287	KOSPI	H287KOSPI200 한국 콜	NaN	NaN	NaN	NaN
5240	258610	KOSDAQ	이더블유케이	구조용 금속제 품, 탱 크 및 증기발 생기 제조업	2017-08-22	12월	마이
350	58GQ08	KOSPI	KBGQ08삼성전 자콜	NaN	NaN	NaN	NaN

In [11]:

df.index

Out[11]:

RangeIndex(start=0, stop=7895, step=1)

In [12]:

df.columns

Out[12]:

```
Index(['Symbol', 'Market', 'Name', 'Sector', 'Industry', 'ListingDate',
       'SettleMonth', 'Representative', 'HomePage', 'Region'],
      dtype='object')
```

In [13]:

#값만 보기
df.values

Out[13]:

```
array([['060310', 'KOSDAQ', '3S', ..., '김세완', 'http://www.3sref.com',
       '서울특별시'],
      ['095570', 'KOSPI', 'AJ네트웍스', ..., '박대현',
       'http://www.ajnet.co.kr', '서울특별시'],
      ['006840', 'KOSPI', 'AK홀딩스', ..., '채형석, 이석주(각자 대표이사)',
       'http://www.aekyunggroup.co.kr', '서울특별시'],
      ...,
      ['037440', 'KOSDAQ', '희림', ..., '정영균, 이복운, 허철호 (각자대표)',
       'http://www.heerim.com', '서울특별시'],
      ['419890', 'KOSPI', '히어로즈 단기채권ESG액티브', ..., nan, nan, nan],
      ['238490', 'KOSDAQ', '힘스', ..., '김주환', 'http://www.hims.co.kr',
       '인천광역시']], dtype=object)
```

In [14]:

#이름만 뽑아봄
df["Name"]

Out[14]:

0	3S
1	AJ네트웍스
2	AK홀딩스
3	APS홀딩스
4	AP시스템
...	
7890	흥국화재우
7891	흥아해운
7892	희림
7893	히어로즈 단기채권ESG액티브
7894	힘스

Name: Name, Length: 7895, dtype: object

In [15]:

```
#loc는 위치(locate)를 의미  
#첫번째 행에 있는 종목 뽑아봄  
df.loc[0]
```

Out [15]:

```
Symbol          060310  
Market         KOSDAQ  
Name           3S  
Sector        전자부품 제조업  
Industry      반도체 웨이퍼 캐리어  
ListingDate   2002-04-23  
SettleMonth    03월  
Representative 김세완  
HomePage      http://www.3sref.com (http://www.3sref.com)  
Region        서울특별시  
Name: 0, dtype: object
```

In [16]:

```
# 이번엔 심볼만 데이터를 뽑아봄.  
df["Symbol"]
```

Out [16]:

```
0      060310  
1      095570  
2      006840  
3      054620  
4      265520  
     ...  
7890    000545  
7891    003280  
7892    037440  
7893    419890  
7894    238490  
Name: Symbol, Length: 7895, dtype: object
```

In [17]:

```
# 심볼과 이름만 뽑아봄.
#여러 칼럼을 지정할 때는 리스트 형태로 묶어야함.
# 2차원 행렬은 대괄호가[]2개가 있음
df[['Symbol', "Name"]]
```

Out [17]:

	Symbol	Name
0	060310	3S
1	095570	AJ네트웍스
2	006840	AK홀딩스
3	054620	APS홀딩스
4	265520	AP시스템
...
7890	000545	흥국화재우
7891	003280	흥아해운
7892	037440	희림
7893	419890	히어로즈 단기채권ESG액티브
7894	238490	힘스

7895 rows × 2 columns

In [18]:

```
# 이번엔 여러개의 행을 가져옴 0번째와 1번째
df.loc[[0,1]]
```

Out [18]:

	Symbol	Market	Name	Sector	Industry	ListingDate	SettleMonth	Representative
0	060310	KOSDAQ	3S	전자부 품 제 조업	반도체 웨이퍼 캐리어	2002-04-23	03월	김세완 http://w
1	095570	KOSPI	AJ네 트워 스	산업용 기계 및 장 비 임 대업	렌탈(파 렌트, OA 장비, 건 설장비)	2015-08-21	12월	박대현 http://w

In [19]:

#여러개의 행과 여러개의 컬럼 가져오기

df.loc[[0, 1], ["Name", "Symbol"]]

Out [19]:

	Name	Symbol
0	3S	060310
1	AJ네트웍스	095570

In [20]:

name에서 주식 삼성전자의 데이터만 뽑아내봄.

df[df["Name"] == "삼성전자"]

Out [20]:

	Symbol	Market	Name	Sector	Industry	ListingDate	SettleMonth	Representative
4139	005930	KOSPI	삼성전자	IMT2000 서비스용 동기식 통신 기지국, 및 방 교환국장 송 장 비,데이 비 제 터단말 조업 기,동영 상휴대 폰,핵심 칩,반...	1975-06-11	12월	한종희	http://

1. 지역별 상장회사 데이터 분석

In [21]:

먼저 지역별로 데이터를 뽑아내봄.

df["Region"]

Out [21]:

```

0      서울특별시
1      서울특별시
2      서울특별시
3      경기도
4      경기도
...
7890     ...
7891     서울특별시
7892     서울특별시
7893     NaN
7894     인천광역시
Name: Region, Length: 7895, dtype: object

```

In [22]:

```
# 서울특별시에 위치한 주식회사 뽑아봄.  
df["Region"] == "서울특별시"
```

Out[22]:

```
0      True  
1      True  
2      True  
3     False  
4     False  
...  
7890    False  
7891    True  
7892    True  
7893    False  
7894    False  
Name: Region, Length: 7895, dtype: bool
```

In [23]:

df[df["Region"]=="서울특별시"]

Out[23]:

	Symbol	Market	Name	Sector	Industry	ListingDate	SettleMonth	Representative
0	060310	KOSDAQ	3S	전자부 품 제 조업	반도체 웨이퍼 캐리어	2002-04-23	03월	김세완
1	095570	KOSPI	AJ네 트웍 스	산업용 기계 및 장 비 임 대업	렌탈(파 렛트, OA 장비, 건 설장비)	2015-08-21	12월	박대현
2	006840	KOSPI	AK홀 딩스	기타 금융업	지주사업	1999-08-11	12월	채형석, 이석주 (각자 대표이사) http
5	211270	KOSDAQ	AP위 성	통신 및 방 송 장 비 제 조업	위성통신 단말기	2016-03-04	12월	류장수
50	027410	KOSPI	BGF	기타 금융업	지주회사	2014-05-19	12월	홍정국
...
7884	069260	KOSPI	휴켐 스	기타 화학제 품 제 조업	화합물, 화학제품 제조	2002-10-07	12월	이건호
7887	189980	KOSDAQ	흥국 에프 엔비	비알코 올음료 및 얼 음 제 조업	과일음료 (에이드 베이스, 스무디, 착즙쥬스 등)	2015-08-07	12월	박철범, 오길영
7888	000540	KOSPI	흥국 화재	보험업	손해보험	1974-12-05	12월	권중원
7891	003280	KOSPI	흥아 해운	해상 운송업	외항화물 운송업 (케미컬 탱커)	1976-06-29	12월	이환구
7892	037440	KOSDAQ	희림	건축기 술, 엔 지니어 링 및 관련 기술 서비스 업	설계 및 감리용역	2000-02-03	12월	정영균, 이목운, 허철호 (각자대 표)

1017 rows × 10 columns

In [24]:

```
#일단 위에서 서울특별시에 있는 회사만 뽑아냄
#그 다음 서울특별시안에서 코스피회사만 뽑아내보자
```

```
#우선 코스피 회사만 뽑아봄.
df[df[ "Market "]== "KOSPI " ]
```

Out [24]:

	Symbol	Market	Name	Sector	Industry	ListingDate	SettleMonth	Representative
1	095570	KOSPI	AJ네트웍스	산업용 기계 및 장비 임대업	렌탈(파렛트, OA 장비, 건설장비)	2015-08-21	12월	박대현
2	006840	KOSPI	AK홀딩스	기타 금융업	지주사업	1999-08-11	12월	채형석, 이석주 (각자 대표이사) ht
6	152100	KOSPI	ARIRANG 200	NaN	NaN	NaN	NaN	NaN
7	295820	KOSPI	ARIRANG 200동일 가증	NaN	NaN	NaN	NaN	NaN
8	253150	KOSPI	ARIRANG 200선물 레버리지	NaN	NaN	NaN	NaN	NaN
...
7888	000540	KOSPI	흥국화재	보험업	손해보험	1974-12-05	12월	권중원
7889	000547	KOSPI	흥국화재2 우B	NaN	NaN	NaN	NaN	NaN
7890	000545	KOSPI	흥국화재 우	NaN	NaN	NaN	NaN	NaN
7891	003280	KOSPI	흥아해운	해상 운송업	외항화물 운송업 (케미컬 탱커)	1976-06-29	12월	이환구
7893	419890	KOSPI	하이로즈 단기채권 ESG액티브	NaN	NaN	NaN	NaN	NaN

6195 rows × 10 columns



In [25]:

```
#and >>> &
#or>>>> |  
  
#연산자 우선순위가 있음 그래서 괄호로 둑어줘야함  
(df[ "Region" ]=="서울특별시" ) & (df[ "Market" ]=="KOSPI")
```

Out [25]:

```
0      False
1      True
2      True
3     False
4     False
...
7890    False
7891    True
7892    False
7893    False
7894    False
Length: 7895, dtype: bool
```

In [26]:

```
#한번더 df묶어줌
df[(df[ "Region" ]=="서울특별시") & (df[ "Market" ]=="KOSPI")]
```

Out [26]:

	Symbol	Market	Name	Sector	Industry	ListingDate	SettleMonth	Representative
1	095570	KOSPI	AJ네트웍스	산업용 기계 및 장비 임대업	렌탈(파렛트, OA 장비, 건설장비)	2015-08-21	12월	박대현
2	006840	KOSPI	AK홀딩스	기타 금융업	지주사업	1999-08-11	12월	채형석, 이석주 (각자 대표이사) http://
50	027410	KOSPI	BGF	기타 금융업	지주회사	2014-05-19	12월	홍정국
51	282330	KOSPI	BGF리테일	종합 소매업	체인화 편의점	2017-12-08	12월	이건준
54	001460	KOSPI	BYC	봉제의 복제조업	메리야 스,란제 리 제조, 도매/건축공사/ 부동산 임대, 분양, 공급	1975-06-02	12월	김 대 환
...
7876	079980	KOSPI	휴비스	화학섬유 제조업	합성섬유 (폴리에스테르원사, 원면), 재생섬유, 폴리에스텔원사, 원면, 고상침 제조, 도소매	2012-02-23	12월	신유동
7879	005010	KOSPI	휴스틸	1차 철강 제조업	강관(배관용, 구조용, 유정용) 제조, 도매	1973-06-29	12월	박훈
7884	069260	KOSPI	휴켐스	기타 화학제품 제조업	화합물, 화학제품 제조	2002-10-07	12월	이건호
7888	000540	KOSPI	흥국화재	보험업	손해보험	1974-12-05	12월	권중원 http://
7891	003280	KOSPI	흥아해운	해상 운송업	외항화물 운송업 (케미컬 탱커)	1976-06-29	12월	이환구

422 rows × 10 columns



In [27]:

```
# 이번엔 ListingDate(상장일)을 통해 년도별만 뽑아내고싶어서
# 년도별로 파생변수 새로 생성함

#pd.to_datetime을 통해 데이터의 타입을 변환함.

df["ListingDate"]
```

Out [27]:

```
0      2002-04-23
1      2015-08-21
2      1999-08-11
3      2001-12-04
4      2017-04-07
...
7890      NaN
7891  1976-06-29
7892  2000-02-03
7893      NaN
7894  2017-07-20
Name: ListingDate, Length: 7895, dtype: object
```

In [28]:

```
pd.to_datetime(df["ListingDate"])
```

Out [28]:

```
0      2002-04-23
1      2015-08-21
2      1999-08-11
3      2001-12-04
4      2017-04-07
...
7890      NaT
7891  1976-06-29
7892  2000-02-03
7893      NaT
7894  2017-07-20
Name: ListingDate, Length: 7895, dtype: datetime64[ns]
```

In [29]:

```
df["ListingDate"] = pd.to_datetime(df["ListingDate"])
```

In [30]:

```
df["ListingDate"]
```

Out[30]:

```
0      2002-04-23
1      2015-08-21
2      1999-08-11
3      2001-12-04
4      2017-04-07
...
7890      NaT
7891      1976-06-29
7892      2000-02-03
7893      NaT
7894      2017-07-20
Name: ListingDate, Length: 7895, dtype: datetime64[ns]
```

In [31]:

```
#ListingYear 파생변수 만들기
```

```
df["ListingDate"].dt.year
```

Out[31]:

```
0      2002.0
1      2015.0
2      1999.0
3      2001.0
4      2017.0
...
7890      NaN
7891      1976.0
7892      2000.0
7893      NaN
7894      2017.0
Name: ListingDate, Length: 7895, dtype: float64
```

In [32]:

```
#ListingYear 파생변수 만들기
```

```
#year 변수로 넣어줌
```

```
df["ListingYear"] = df["ListingDate"].dt.year
```

In [33]:

```
df["ListingYear"]
```

Out[33]:

```
0      2002.0
1      2015.0
2      1999.0
3      2001.0
4      2017.0
...
7890      NaN
7891      1976.0
7892      2000.0
7893      NaN
7894      2017.0
Name: ListingYear, Length: 7895, dtype: float64
```

In [34]:

```
# 위에서 보면 Nan이 나오는데 not a number 약자로 결측치임
#결측치의 타입이 float임
import numpy as np
type(np.nan)
```

Out[34]:

```
float
```

In [35]:

df

Out [35]:

	Symbol	Market	Name	Sector	Industry	ListingDate	SettleMonth	Representative
0	060310	KOSDAQ	3S	전자부 품 제 조업	반도체 웨이퍼 캐리어	2002-04-23	03월	김세완
1	095570	KOSPI	AJ네 트웍 스	산업용 기계 및 장 비 임 대업	렌탈(파 렛트, OA 장비, 건 설장비)	2015-08-21	12월	박대현
2	006840	KOSPI	AK홀 딩스	기타 금융업	지주사업	1999-08-11	12월	채형석, 이석주 (각자 대표이사) http
3	054620	KOSDAQ	APS 홀딩 스	기타 금융업	인터넷 트래픽 솔루션	2001-12-04	12월	정기로 h
4	265520	KOSDAQ	AP시 스템	특수 목적용 기계 제조업	디스플레 이 제조 장비	2017-04-07	12월	김영주
...
7890	000545	KOSPI	흥국 화재 우	NaN	NaN	NaT	NaN	NaN
7891	003280	KOSPI	흥아 해운	해상 운송업	외항화물 운송업 (케미컬 탱커)	1976-06-29	12월	이환구
7892	037440	KOSDAQ	희림	건축기 술, 엔 지니어 링 및 관련 기술 서비스 업	설계 및 감리용역	2000-02-03	12월	정영균, 이목운, 허철호 (각자대 표)
7893	419890	KOSPI	히어 로즈 단기 채권 ESG 액티 브	NaN	NaN	NaT	NaN	NaN
7894	238490	KOSDAQ	힘스	특수 목적용 기계 제조업	OLED Mask 인 장기, OLED Mask 검 사기 등	2017-07-20	12월	김주환

7895 rows × 11 columns

In [36]:

```
df[["ListingYear", "ListingDate"]]
```

Out [36]:

	ListingYear	ListingDate
0	2002.0	2002-04-23
1	2015.0	2015-08-21
2	1999.0	1999-08-11
3	2001.0	2001-12-04
4	2017.0	2017-04-07
...
7890	NaN	NaT
7891	1976.0	1976-06-29
7892	2000.0	2000-02-03
7893	NaN	NaT
7894	2017.0	2017-07-20

7895 rows × 2 columns

In [37]:

```
df[["ListingYear", "ListingDate"]].dtypes
```

Out [37]:

```
ListingYear      float64
ListingDate    datetime64[ns]
dtype: object
```

지역별,년도별로 데이터를 다 뽑아 냈으니

matplotlib 이용해 데이터 시각화

In [38]:

```
# 데이터 시각화 전 폰트 설정
import matplotlib.font_manager as fm
[f.name for f in fm.fontManager.ttflist]
['Adobe Arabic',
'Sitka Small',
'Verdana',
'Microsoft New Tai Lue',
'GyeonggiTitle',
'Playbill',
'Kozuka Gothic Pr6N',
'Trajan Pro',
'Cooper BIK BT',
'Haansoft Batang',
'Wingdings',
'MHunmin',
>Showcard Gothic',

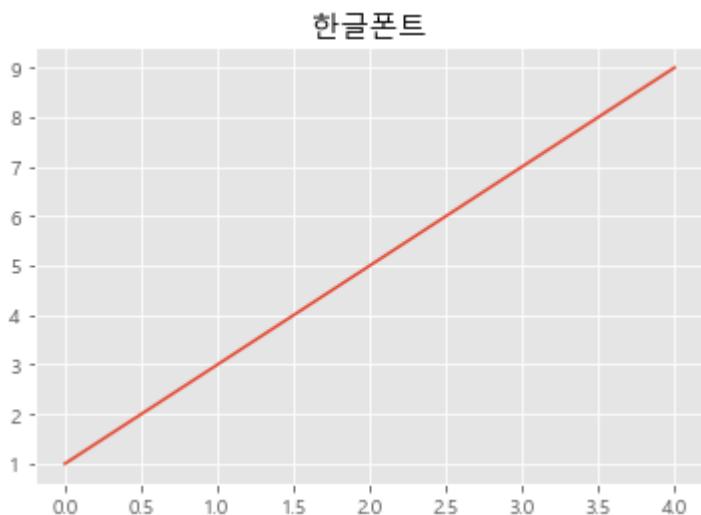
'KBIZmjo B',
'Symbol',
'Bookshelf Symbol 7',
'??????',
'Tw Cen MT Condensed',
'Gloucester MT Extra Condensed',
'??????.ttf']
```

In [39]:

```
import matplotlib.pyplot as plt
font_family=plt.rcParams["font.family"]='Malgun Gothic'
plt.rc("axes",unicode_minus=False)
plt.style.use("ggplot")
pd.Series([1,3,5,7,9]).plot(title="한글폰트")
```

Out [39]:

<AxesSubplot:title={'center':'한글폰트'}>



1. 주식시장 별 상장주식 빈도수 시각화

In [40]:

```
#Market column에 어떤 데이터들이 있는지 unique로 확인 중복을 제거한 값이 나옴
df[ "Market" ].unique()
```

Out [40]:

```
array(['KOSDAQ', 'KOSPI', 'KONEX'], dtype=object)
```

In [41]:

```
# 국내주식 시장은 'KOSDAQ', 'KOSPI', 'KONEX' 가 있음
# value_counts()로 빈도수 구함
df[ "Market" ].value_counts()
```

Out [41]:

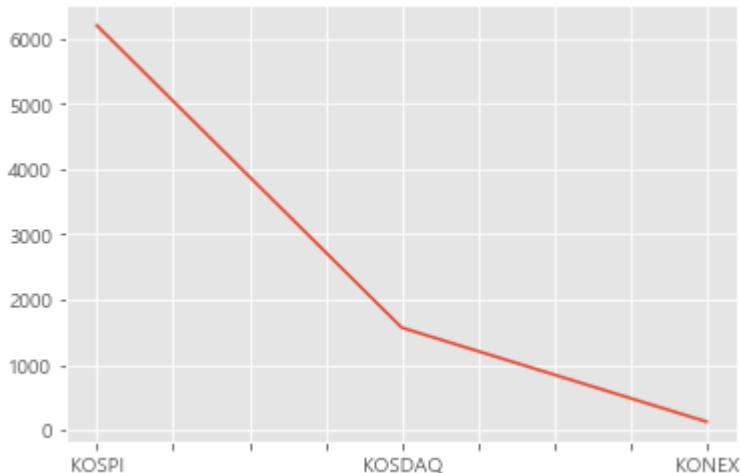
```
KOSPI    6195
KOSDAQ   1570
KONEX    130
Name: Market, dtype: int64
```

In [42]:

```
# value_counts()로 빈도수 구함
df[ "Market" ].value_counts().plot()
```

Out [42]:

```
<AxesSubplot:>
```

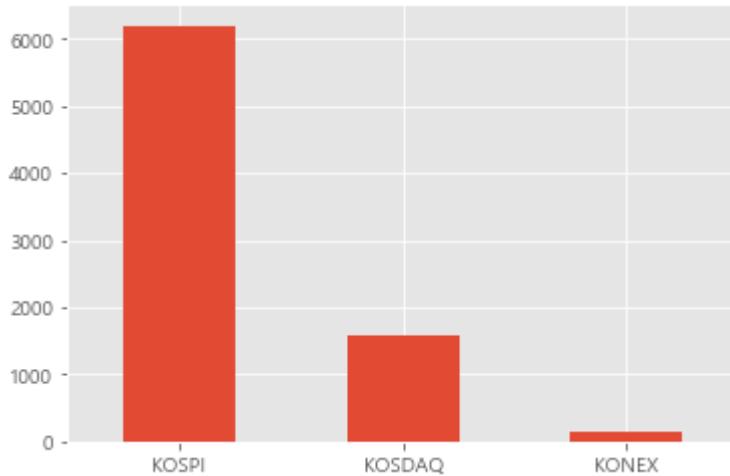


In [43]:

```
# value_counts()로 빈도수 구함  
df[ "Market" ].value_counts().plot.bar(rot=0)
```

Out[43]:

<AxesSubplot:>

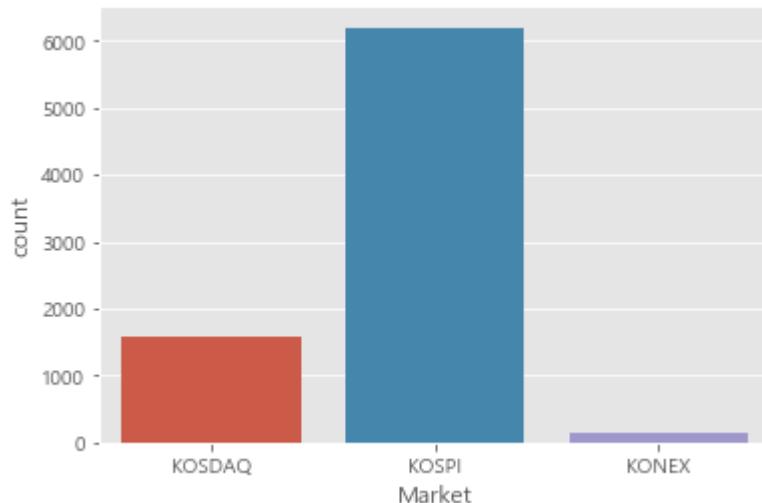


In [44]:

```
#Seaborn countplot 활용해서 빈도수 표현  
import seaborn as sns  
sns.countplot(data=df, x="Market")
```

Out[44]:

<AxesSubplot:xlabel='Market', ylabel='count'>



2. 주식 섹터별 빈도수 구함.

In [45]:

df["Sector"]

Out[45]:

```
0          전자부품 제조업
1          산업용 기계 및 장비 임대업
2          기타 금융업
3          기타 금융업
4          특수 목적용 기계 제조업
...
7890         NaN
7891         해상 운송업
7892  건축기술, 엔지니어링 및 관련 기술 서비스업
7893         NaN
7894         특수 목적용 기계 제조업
Name: Sector, Length: 7895, dtype: object
```

In [46]:

#섹터별 갯수

df["Sector"].value_counts()

Out[46]:

```
소프트웨어 개발 및 공급업      154
특수 목적용 기계 제조업      145
전자부품 제조업            129
자동차 신품 부품 제조업      96
의약품 제조업              95
...
담배 제조업                  1
의복 액세서리 제조업          1
악기 제조업                  1
인형, 장난감 및 오락용품 제조업    1
해체, 선별 및 원료 재생업          1
Name: Sector, Length: 161, dtype: int64
```

In [47]:

```
# 갯수가 많은 순으로 다시 보여주기위해 일단 변수 하나생성
sector_count_top=df[ "Sector" ].value_counts().head(30)
sector_count_top
```

Out[47]:

소프트웨어 개발 및 공급업	154
특수 목적용 기계 제조업	145
전자부품 제조업	129
자동차 신품 부품 제조업	96
의약품 제조업	95
기타 금융업	94
기타 화학제품 제조업	93
금융 지원 서비스업	75
통신 및 방송 장비 제조업	72
반도체 제조업	59
자연과학 및 공학 연구개발업	59
의료용 기기 제조업	57
1차 철강 제조업	57
일반 목적용 기계 제조업	45
플라스틱제품 제조업	45
기초 화학물질 제조업	42
기초 의약물질 및 생물학적 제제 제조업	40
기타 전문 도매업	38
컴퓨터 프로그래밍, 시스템 통합 및 관리업	37
전동기, 발전기 및 전기 변환 · 공급 · 제어 장치 제조업	35
기타 식품 제조업	34
영화, 비디오물, 방송프로그램 제작 및 배급업	32
의료용품 및 기타 의약 관련제품 제조업	30
봉제의복 제조업	28
기계장비 및 관련 물품 도매업	25
기타 금속 가공제품 제조업	24
측정, 시험, 항해, 제어 및 기타 정밀기기 제조업; 광학기기 제외	23
건물 건설업	23
부동산 임대 및 공급업	21
구조용 금속제품, 탱크 및 증기발생기 제조업	20

Name: Sector, dtype: int64

In [48]:

sector_count_top.index

Out[48]:

```
Index(['소프트웨어 개발 및 공급업', '특수 목적용 기계 제조업', '전자부품 제조업',
'자동차 신품 부품 제조업',
'의약품 제조업', '기타 금융업', '기타 화학제품 제조업', '금융 지원 서비스업',
'통신 및 방송 장비 제조업',
'반도체 제조업', '자연과학 및 공학 연구개발업', '의료용 기기 제조업', '1차 철
강 제조업',
'일반 목적용 기계 제조업', '플라스틱제품 제조업', '기초 화학물질 제조업', '기
초 의약물질 및 생물학적 제제 제조업',
'기타 전문 도매업', '컴퓨터 프로그래밍, 시스템 통합 및 관리업',
'전동기, 발전기 및 전기 변환 · 공급 · 제어 장치 제조업', '기타 식품 제조
업',
'영화, 비디오물, 방송프로그램 제작 및 배급업', '의료용품 및 기타 의약 관련제
품 제조업', '봉제의복 제조업',
'기계장비 및 관련 물품 도매업', '기타 금속 가공제품 제조업',
'측정, 시험, 항해, 제어 및 기타 정밀기기 제조업; 광학기기 제외', '건물 건설
업', '부동산 임대 및 공급업',
'구조용 금속제품, 탱크 및 증기발생기 제조업'],
      dtype='object')
```

In [49]:

```
#isin() 함수를 사용하여 특정 조건의 값 추출
df["Sector"].isin(sector_count_top.index)
```

Out[49]:

```
0      True
1     False
2      True
3      True
4      True
...
7890    False
7891    False
7892    False
7893    False
7894    True
Name: Sector, Length: 7895, dtype: bool
```

In [50]:

```
df[df[ "Sector" ].isin(sector_count_top.index)]
```

Out [50]:

	Symbol	Market	Name	Sector	Industry	ListingDate	SettleMonth	Representative
0	060310	KOSDAQ	3S	전자부 품 제 조업	반도체 웨이퍼 캐리어	2002-04-23	03월	김세완
2	006840	KOSPI	AK홀 딩스	기타 금융업	지주사업	1999-08-11	12월	채형석, 이석주 (각자 대표이사) http
3	054620	KOSDAQ	APS 홀딩 스	기타 금융업	인터넷 트래픽 솔루션	2001-12-04	12월	정기로 h
4	265520	KOSDAQ	AP시 스템	특수 목적용 기계 제조업	디스플레 이 제조 장비	2017-04-07	12월	김영주
5	211270	KOSDAQ	AP위 성	통신 및 방 송 장 비 제 조업	위성통신 단말기	2016-03-04	12월	류장수
...
7883	145020	KOSDAQ	휴젤	기초 의약물 질 및 생물학 적 제 제 제 조업	보툴리눔 톡신, 필 러	2015-12-24	12월	손지훈
7884	069260	KOSPI	휴켐 스	기타 화학제 품 제 조업	화합물, 화학제품 제조	2002-10-07	12월	이건호
7885	024060	KOSDAQ	흥구 석유	기타 전문 도매업	휘발유등	1994-12-07	12월	김상우
7886	010240	KOSDAQ	흥국	특수 목적용 기계 제조업	굴삭기 Roller 및 형단조품	2009-05-12	12월	류명준
7894	238490	KOSDAQ	힘스	특수 목적용 기계 제조업	OLED Mask 인 장기, OLED Mask 검 사기 등	2017-07-20	12월	김주환

1727 rows × 11 columns



In [51]:

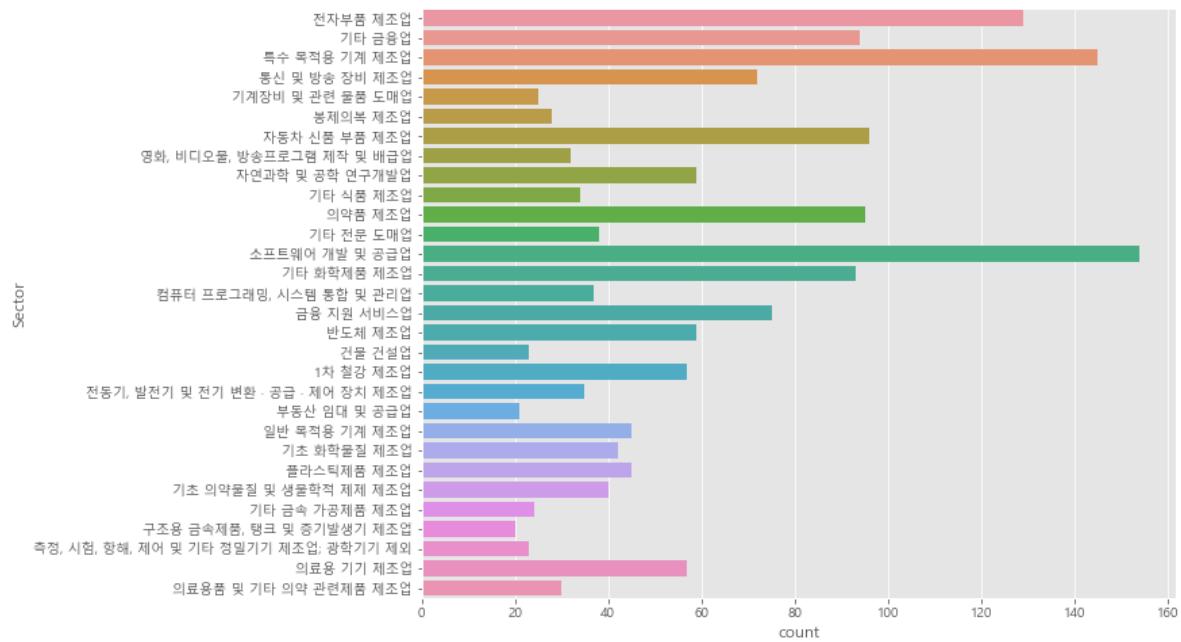
```
df_sector_30=df[df["Sector"].isin(sector_count_top.index)]
```

In [52]:

```
# 섹터별로 많은순으로 30개만 뽑아내봄.
plt.figure(figsize=(10,8))
sns.countplot(data=df_sector_30, y="Sector")
```

Out [52]:

```
<AxesSubplot:xlabel='count', ylabel='Sector'>
```



In [53]:

#색깔 뭐있는지 확인

```
print(plt.colormaps())
```

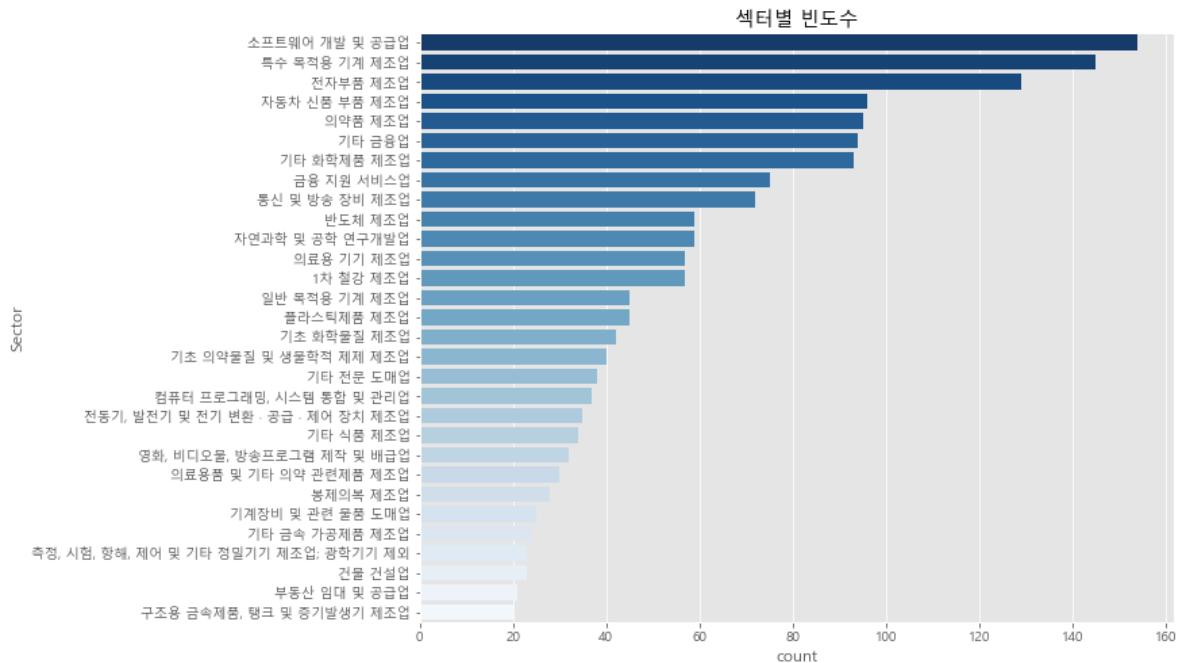
```
['magma', 'inferno', 'plasma', 'viridis', 'cividis', 'twilight', 'twilight_shifted',  
'turbo', 'Blues', 'BrBG', 'BuGn', 'BuPu', 'CMRmap', 'GnBu', 'Greens', 'Greys', 'OrR  
d', 'Oranges', 'PRGn', 'PiYG', 'PuBu', 'PuBuGn', 'PuOr', 'PuRd', 'Purples', 'RdBu',  
'RdGy', 'RdPu', 'RdYlBu', 'RdYlGn', 'Reds', 'Spectral', 'Wistia', 'YlGn', 'YlGnBu',  
'YlOrBr', 'YlOrRd', 'afmhot', 'autumn', 'binary', 'bone', 'brg', 'bwr', 'cool', 'coo  
lwarm', 'copper', 'cubehelix', 'flag', 'gist_earth', 'gist_gray', 'gist_heat', 'gist  
_ncar', 'gist_rainbow', 'gist_stern', 'gist_yarg', 'gnuplot', 'gnuplot2', 'gray', 'h  
ot', 'hsv', 'jet', 'nipy_spectral', 'ocean', 'pink', 'prism', 'rainbow', 'seismic',  
'spring', 'summer', 'terrain', 'winter', 'Accent', 'Dark2', 'Paired', 'Pastel1', 'Pa  
stel2', 'Set1', 'Set2', 'Set3', 'tab10', 'tab20', 'tab20b', 'tab20c', 'magma_r', 'in  
ferno_r', 'plasma_r', 'viridis_r', 'cividis_r', 'twilight_r', 'twilight_shifted_r',  
'turbo_r', 'Blues_r', 'BrBG_r', 'BuGn_r', 'BuPu_r', 'CMRmap_r', 'GnBu_r', 'Greens_  
r', 'Greys_r', 'OrRd_r', 'Oranges_r', 'PRGn_r', 'PiYG_r', 'PuBu_r', 'PuBuGn_r', 'Pu0  
r_r', 'PuRd_r', 'Purples_r', 'RdBu_r', 'RdGy_r', 'RdPu_r', 'RdYlBu_r', 'RdYlGn_r',  
'Reds_r', 'Spectral_r', 'Wistia_r', 'YlGn_r', 'YlGnBu_r', 'YlOrBr_r', 'YlOrRd_r', 'a  
fmhot_r', 'autumn_r', 'binary_r', 'bone_r', 'brg_r', 'bwr_r', 'cool_r', 'coolwarm_  
r', 'copper_r', 'cubehelix_r', 'flag_r', 'gist_earth_r', 'gist_gray_r', 'gist_heat_  
r', 'gist_ncar_r', 'gist_rainbow_r', 'gist_stern_r', 'gist_yarg_r', 'gnuplot_r', 'gn  
uplot2_r', 'gray_r', 'hot_r', 'hsv_r', 'jet_r', 'nipy_spectral_r', 'ocean_r', 'pink_  
r', 'prism_r', 'rainbow_r', 'seismic_r', 'spring_r', 'summer_r', 'terrain_r', 'winte  
r_r', 'Accent_r', 'Dark2_r', 'Paired_r', 'Pastel1_r', 'Pastel2_r', 'Set1_r', 'Set2_  
r', 'Set3_r', 'tab10_r', 'tab20_r', 'tab20b_r', 'tab20c_r', 'rocket', 'rocket_r', 'm  
ako', 'mako_r', 'icefire', 'icefire_r', 'vlag', 'vlag_r', 'flare', 'flare_r', 'cres  
t', 'crest_r']
```

In [54]:

```
# 보기좋게 빈도수가 많은순으로 정렬
#빈도수가 가장 많은 데이터를 색인하기
#빈도수 색깔을 blues로했는데 색깔 진한거를 많은순으로 바꾸고 싶으면 색깔뒤에 _r 붙여주면 뒤바뀜
# 타이틀 지정
plt.figure(figsize=(10,8))
sns.countplot(data=df_sector_30, y="Sector", palette="Blues_r", order=sector_count_top.index).set_tit
```

Out [54]:

Text(0.5, 1.0, '섹터별 빈도수')



3. 주식 산업별 빈도수 데이터분석

In [55]:

df

Out [55]:

	Symbol	Market	Name	Sector	Industry	ListingDate	SettleMonth	Representative
0	060310	KOSDAQ	3S	전자부 품 제 조업	반도체 웨이퍼 캐리어	2002-04-23	03월	김세완
1	095570	KOSPI	AJ네 트웍 스	산업용 기계 및 장 비 임 대업	렌탈(파 렛트, OA 장비, 건 설장비)	2015-08-21	12월	박대현
2	006840	KOSPI	AK홀 딩스	기타 금융업	지주사업	1999-08-11	12월	채형석, 이석주 (각자 대표이사) http
3	054620	KOSDAQ	APS 홀딩 스	기타 금융업	인터넷 트래픽 솔루션	2001-12-04	12월	정기로 h
4	265520	KOSDAQ	AP시 스템	특수 목적용 기계 제조업	디스플레 이 제조 장비	2017-04-07	12월	김영주
...
7890	000545	KOSPI	흥국 화재 우	NaN	NaN	NaT	NaN	NaN
7891	003280	KOSPI	흥아 해운	해상 운송업	외항화물 운송업 (케미컬 탱커)	1976-06-29	12월	이환구
7892	037440	KOSDAQ	희림	건축기 술, 엔 지니어 링 및 관련 기술 서비스 업	설계 및 감리용역	2000-02-03	12월	정영균, 이목운, 허철호 (각자대 표)
7893	419890	KOSPI	히어 로즈 단기 채권 ESG 액티 브	NaN	NaN	NaT	NaN	NaN
7894	238490	KOSDAQ	힘스	특수 목적용 기계 제조업	OLED Mask 인 장기, OLED Mask 검 사기 등	2017-07-20	12월	김주환

7895 rows × 11 columns

In [56]:

df["Industry"]

Out [56]:

```

0           반도체 웨이퍼 캐리어
1           렌탈(파렛트, OA장비, 건설장비)
2           지주사업
3           인터넷 트래픽 솔루션
4           디스플레이 제조 장비
...
7890          NaN
7891  외항화물운송업(케미컬탱커)
7892          설계 및 감리용역
7893          NaN
7894  OLED Mask 인장기, OLED Mask 검사기 등
Name: Industry, Length: 7895, dtype: object

```

In [57]:

df["Industry"].value_counts().head(30)

Out [57]:

기업인수합병	28
지주회사	20
지주사업	9
모바일게임	7
화장품	7
기업인수 및 합병	6
금융 지원 서비스	5
기업 인수 및 합병	5
-	4
부동산투자	4
합병	4
셋톱박스	3
금융 지원 서비스업	3
벤처캐피탈	3
금융지원 서비스	3
도급공사	3
자동차부품	3
의약품	3
건강기능식품	3
금융지주회사	3
증권업	3
화장품 용기	2
백화점	2
엔진부품	2
레미콘	2
비금융지주회사	2
카지노	2
원료의약품	2
이미지센서 패키징	2
의료기기	2

Name: Industry, dtype: int64

In [58]:

```
industry_count_top=df["Industry"].value_counts().head(30)  
industry_count_top.index
```

Out[58]:

```
Index(['기업인수합병', '지주회사', '지주사업', '모바일게임', '화장품', '기업인수 및  
합병', '금융 지원 서비스',  
       '기업 인수 및 합병', '-', '부동산투자', '합병', '셋톱박스', '금융 지원 서비스  
업', '벤처캐피탈',  
       '금융지원 서비스', '도급공사', '자동차부품', '의약품', '건강기능식품', '금융  
지주회사', '증권업', '화장품 용기',  
       '백화점', '엔진부품', '레미콘', '비금융지주회사', '카지노', '원료의약품', '이  
미지센서 패키징', '의료기기'],  
      dtype='object')
```

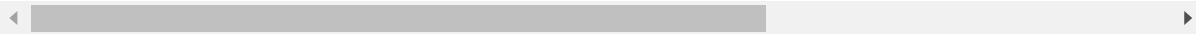
In [59]:

```
df[df["Industry"].isin(industry_count_top.index)]
```

Out [59]:

	Symbol	Market	Name	Sector	Industry	ListingDate	SettleMonth	Representative
2	006840	KOSPI	AK홀딩스	기타 금융업	지주사업	1999-08-11	12월	채형석, 이석주 (각자 대표이사)
50	027410	KOSPI	BGF	기타 금융업	지주회사	2014-05-19	12월	홍정국
53	138930	KOSPI	BNK 금융지주	기타 금융업	금융지주 회사	2011-03-30	12월	김지완
56	013720	KOSDAQ	CBI	자동차 신품 부품 제조업	엔진부품	1993-12-29	12월	오경원
57	001040	KOSPI	CJ	기타 금융업	지주회사	1973-06-29	12월	손경식, 김홍기
...
7849	126640	KOSDAQ	화신 정공	자동차 신품 부품 제조업	자동차부 품	2010-08-31	12월	정서진
7859	298040	KOSPI	효성 중공업	전동 기, 발 전기 및 전 기 변 환 · 공 급 · 제 어 장 치 제 조업	-	2018-07-13	12월	요코타 타케시 http://
7862	298000	KOSPI	효성 화학	기초 화학물 질 제 조업	-	2018-07-13	12월	이건종
7868	353190	KOSDAQ	휴럼	기타 식품 제조업	금융지원 서비스	2020-06-17	12월	김진석
7871	115160	KOSDAQ	휴맥스	영상 및 음 향기기 제조업	셋톱박스	2009-11-16	12월	김태훈

147 rows × 11 columns

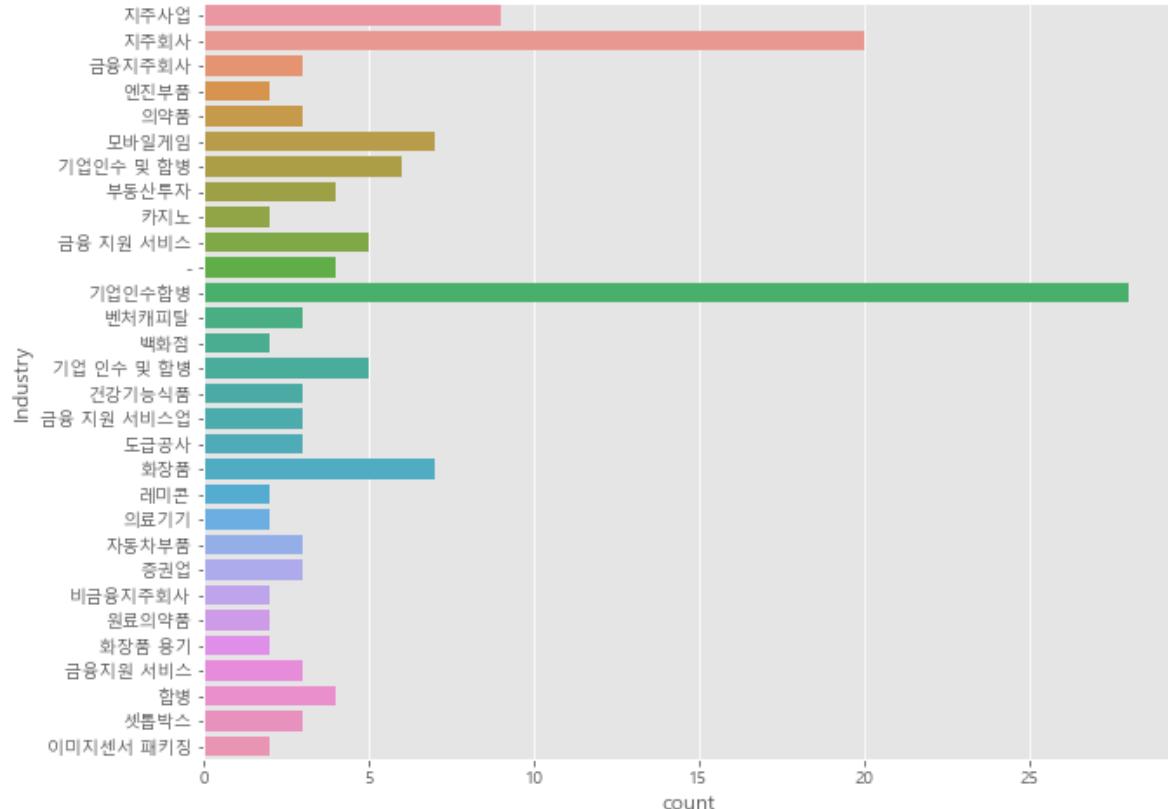


In [60]:

```
#seaborn  
plt.figure(figsize=(10,8))  
  
sns.countplot(data=df[df["Industry"].isin(industry_count_top.index)],y="Industry")
```

Out[60]:

<AxesSubplot:xlabel='count', ylabel='Industry'>

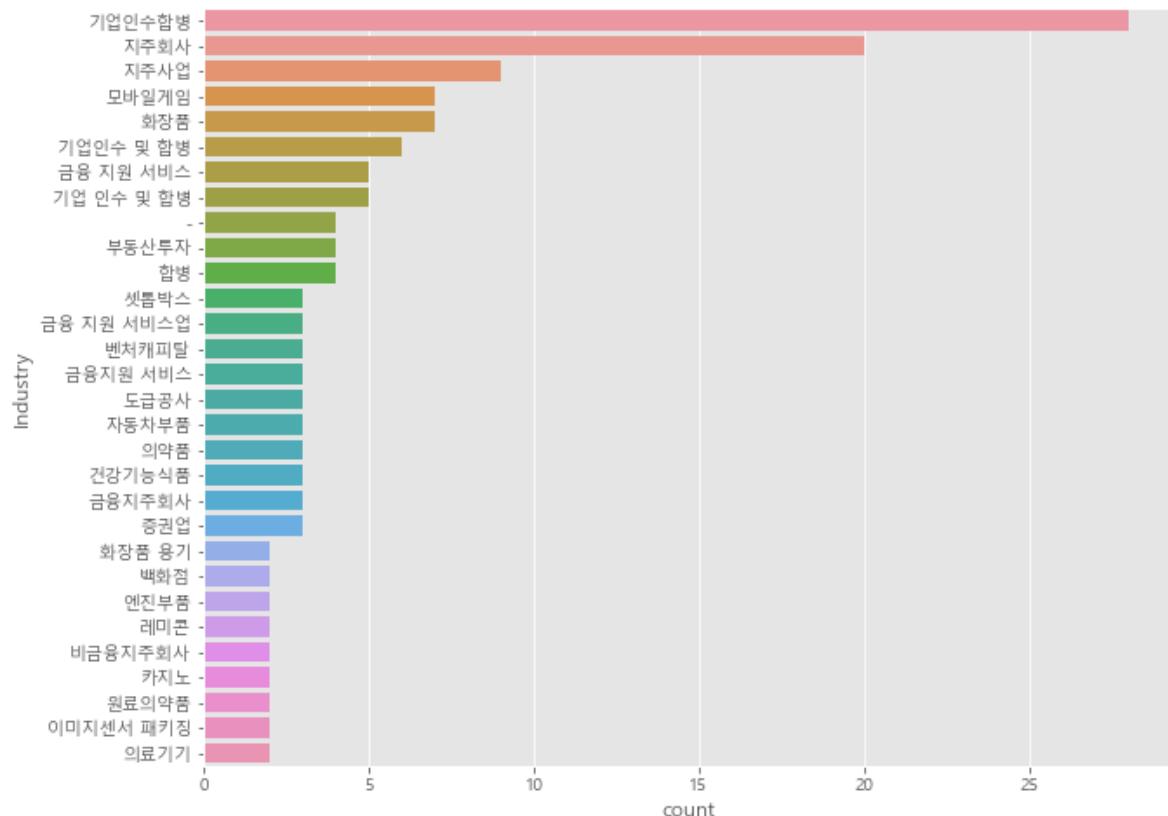


In [61]:

```
#seaborn  
plt.figure(figsize=(10,8))  
  
sns.countplot(data=df[df["Industry"].isin(industry_count_top.index)],y="Industry",order=industry_cou
```

Out[61]:

<AxesSubplot:xlabel='count', ylabel='Industry'>



4. 주식 지역별 데이터분석

In [62]:

df["Region"]

Out[62]:

```
0      서울특별시
1      서울특별시
2      서울특별시
3      경기도
4      경기도
...
7890     NaN
7891     서울특별시
7892     서울특별시
7893     NaN
7894     인천광역시
Name: Region, Length: 7895, dtype: object
```

In [63]:

df["Region"].value_counts().head(30)

Out[63]:

서울특별시	1017
경기도	709
충청남도	98
경상남도	90
인천광역시	87
충청북도	85
부산광역시	77
경상북도	64
대구광역시	53
대전광역시	52
전라북도	29
강원도	27
울산광역시	26
전라남도	22
광주광역시	22
세종특별자치시	10
제주특별자치도	9
홍콩	8
미국	6
케이맨 제도	5
일본	3
싱가포르	1

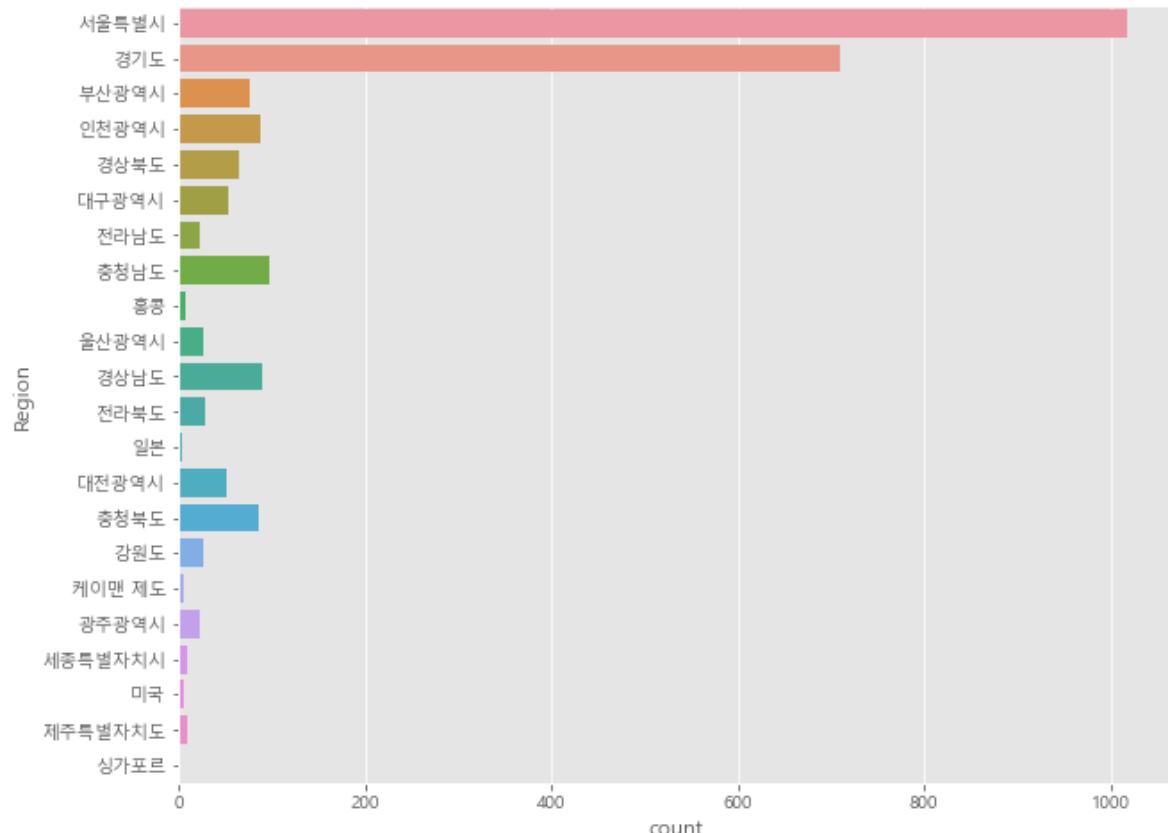
Name: Region, dtype: int64

In [64]:

```
#Region의 빈도수를 seaborn으로 표시  
plt.figure(figsize=(10,8))  
  
sns.countplot(data=df, y="Region")
```

Out [64]:

<AxesSubplot:xlabel='count', ylabel='Region'>



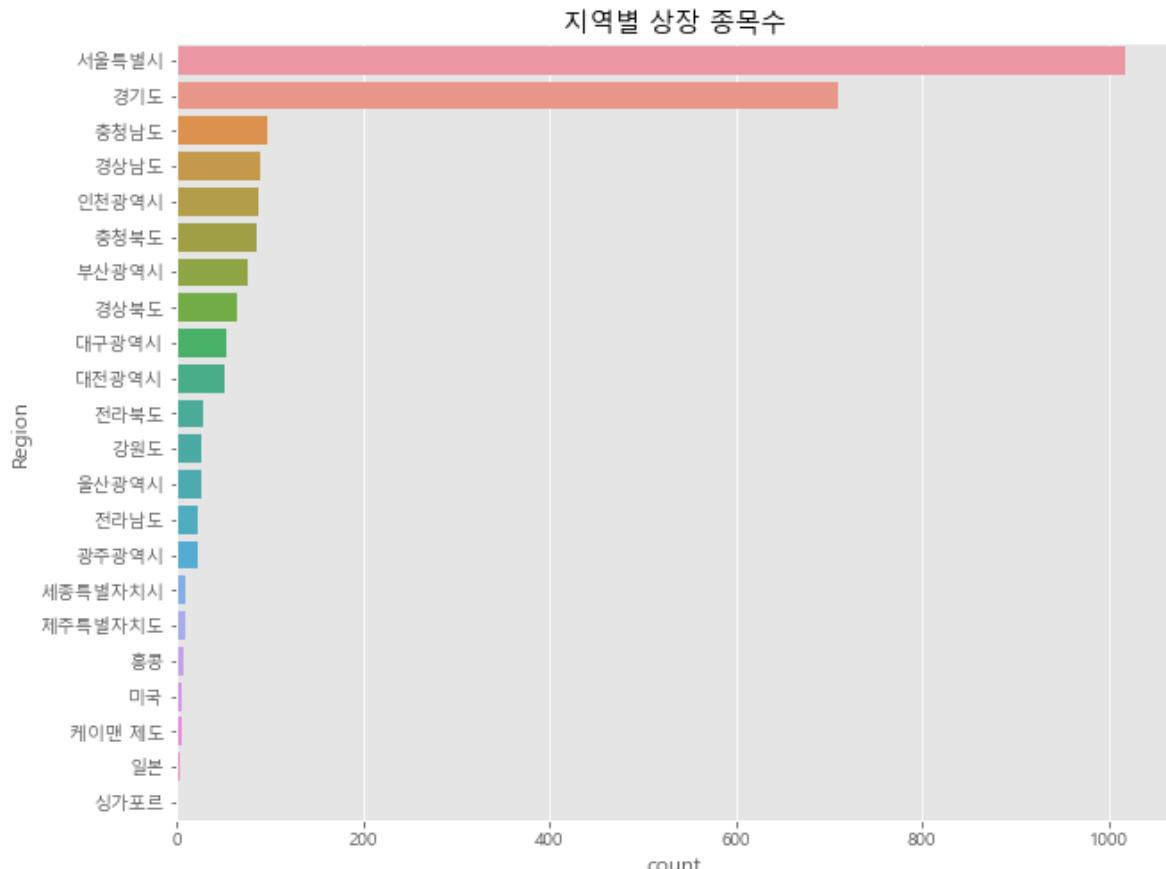
In [65]:

```
plt.figure(figsize=(10,8))

sns.countplot(data=df, y="Region",
               order=df["Region"].value_counts().index).set_title("지역별 상장 종목수")
```

Out [65]:

Text(0.5, 1.0, '지역별 상장 종목수')



위에서 시장, 섹터, 산업, 지역 별로 빈도수를 구했으니 두개 이상의 빈도수 구하기

1. 시장+지역별로 구해봄.

In [66]:

```
# Market과 Region으로 빈도수 구하기
pd.crosstab(df[ "Market" ],df[ "Region" ])
```

Out [66]:

Region	강원도	경기도	경상남도	경상북도	광주광역시	대구광역시	대전광역시	미국	부산광역시	서울특별시	울산광역시	인천광역시	일본	전라남도	전라북도	제주특별자치도	충청북도	충청남도	케이맨제도	...
Market																					
KONEX	6	33	1	5	4	1	7	0	2	48	...	0	4	0	3	1	1	4	9	0	
KOSDAQ	17	526	47	40	9	31	38	6	40	547	...	9	59	3	12	16	2	68	61	4	
KOSPI	4	150	42	19	9	21	7	0	35	422	...	17	24	0	7	12	6	26	15	1	

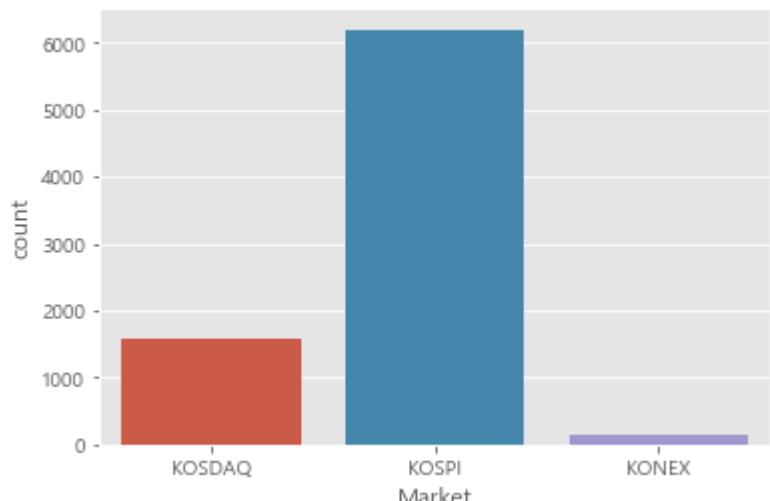
3 rows × 22 columns

In [67]:

```
sns.countplot(data=df, x="Market")
```

Out [67]:

<AxesSubplot:xlabel='Market', ylabel='count'>

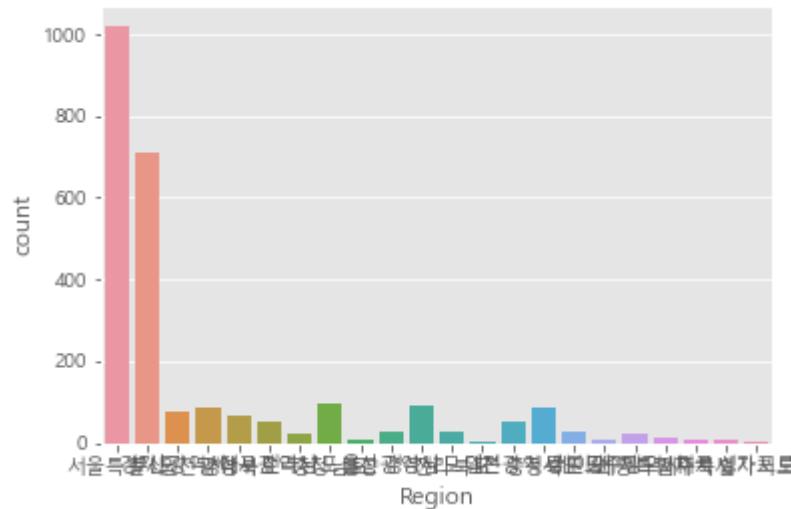


In [68]:

```
sns.countplot(data=df, x="Region")
```

Out[68]:

```
<AxesSubplot:xlabel='Region', ylabel='count'>
```

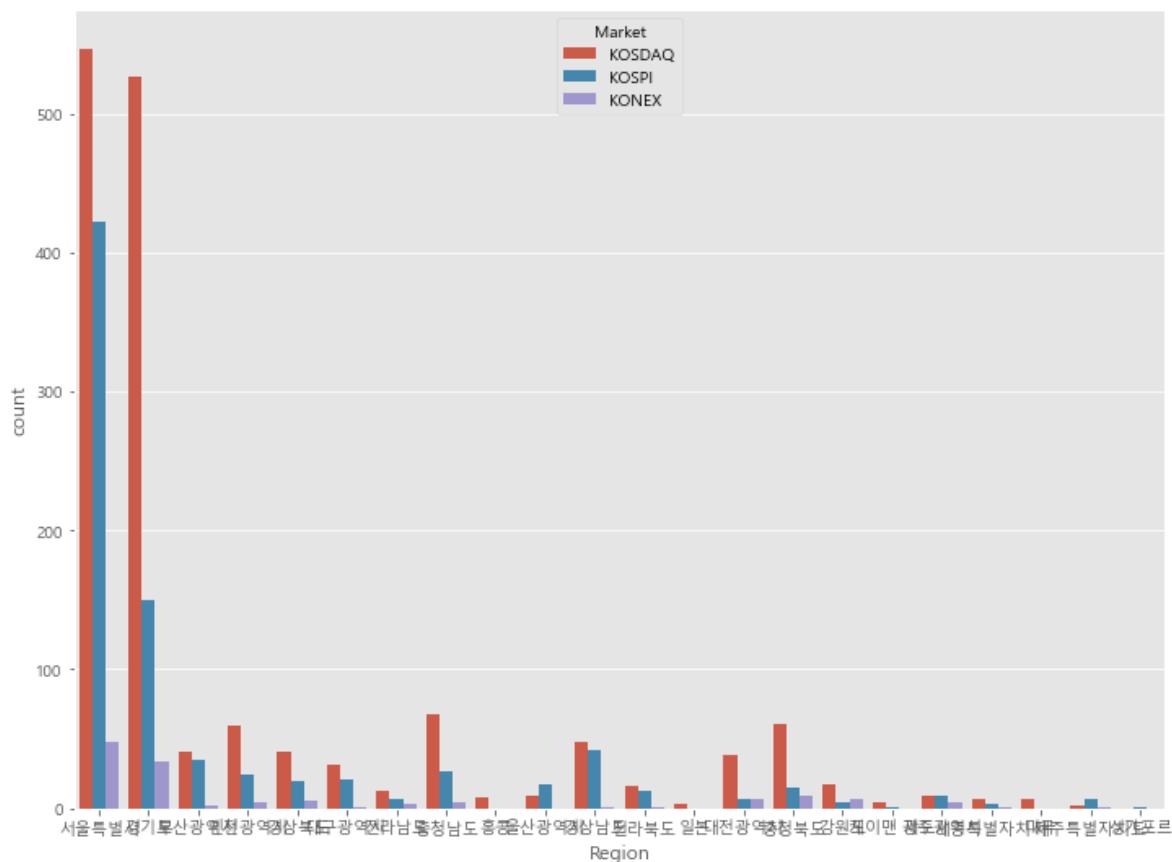


In [69]:

```
# hue는 범주형 데이터에서 색깔별로 시각화 해줌  
plt.figure(figsize=(12,9))
```

```
sns.countplot(data=df, x="Region",hue="Market")
```

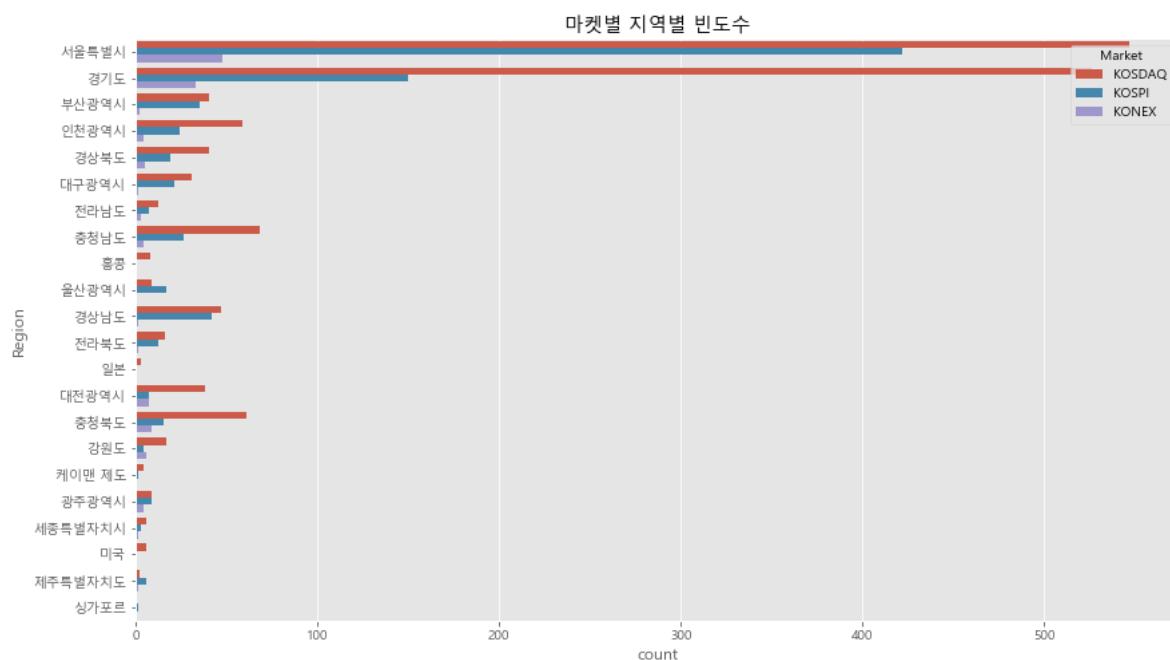
Out[69]:



In [70]:

```
# hue는 범주형 데이터에서 색깔별로 시각화 해줌
plt.figure(figsize=(14,8))
# plt.xticks(rotation=90) #글씨를 90도로 나오게 해줌
#x축을 y축으로 바꾸면 더 날음
#로그 안나오게 하려면 변수에 넣어줌

#countplot(data=dataframe,y="y축", hue=데이터 column)
#y는 countplot를 어떤 부분을 count할지 정해줌
# data는 사용할 data를 정의
#hue는 정한 데이터를 다시 hue기준으로 더 나누어서 count 해줌.
g=sns.countplot(data=df, y="Region",hue="Market").set_title("마켓별 지역별 빈도수")
```



2. 시장+연도별 구해봄.

In [71]:

```
pd.crosstab(df[ "Market" ],df[ "ListingYear" ])
```

Out[71]:

Market	1956.0	1962.0	1964.0	1966.0	1968.0	1969.0	1970.0	1971.0	1972.0	1973.0	...
KONEX	0	0	0	0	0	0	0	0	0	0	...
KOSDAQ	0	0	0	0	0	0	0	0	0	0	...
KOSPI	5	1	1	1	6	5	4	1	3	22	...

3 rows × 58 columns

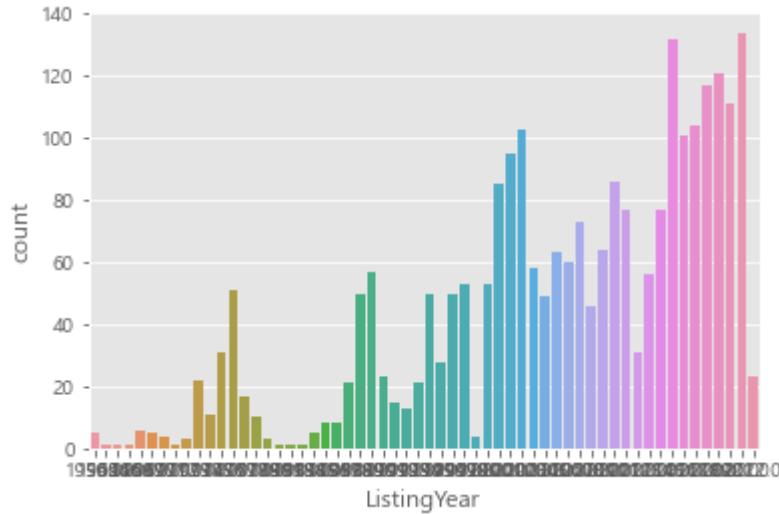
In [72]:

#연도가 많아서 잘안보임

sns.countplot(data=df, x="ListingYear")

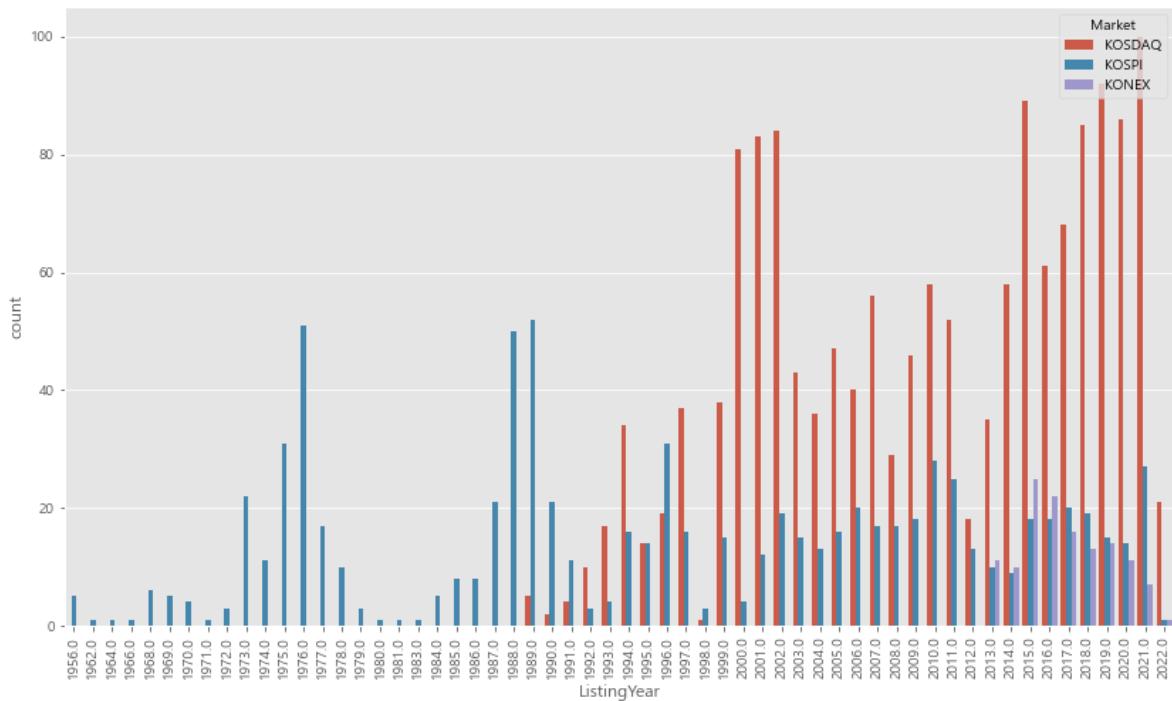
Out[72]:

<AxesSubplot:xlabel='ListingYear', ylabel='count'>



In [73]:

```
#hue를 사용해 다른 변수를 함께 표현. hue는 색상을 의미
# 차트 사이즈 키워주고
# 년도를 세로로 해서 잘보이게 정렬
#x축은 년도 hue를 통해 시장을 더 쪼개서 보여줌
plt.figure(figsize=(14,8))
plt.xticks(rotation=90)
g=sns.countplot(data=df, x="ListingYear",hue="Market")
```



3. 응용 분석 (2018년에 상장한, 서울특별시에 소재한 코스피 종목 찾아봄.)

In [74]:

df[df["Region"]=="서울특별시"]

Out[74]:

	Symbol	Market	Name	Sector	Industry	ListingDate	SettleMonth	Representative
0	060310	KOSDAQ	3S	전자부 품 제 조업	반도체 웨이퍼 캐리어	2002-04-23	03월	김세완
1	095570	KOSPI	AJ네 트웍 스	산업용 기계 및 장 비 임 대업	렌탈(파 렛트, OA 장비, 건 설장비)	2015-08-21	12월	박대현
2	006840	KOSPI	AK홀 딩스	기타 금융업	지주사업	1999-08-11	12월	채형석, 이석주 (각자 대표이사) http
5	211270	KOSDAQ	AP위 성	통신 및 방 송 장 비 제 조업	위성통신 단말기	2016-03-04	12월	류장수
50	027410	KOSPI	BGF	기타 금융업	지주회사	2014-05-19	12월	홍정국
...
7884	069260	KOSPI	휴켐 스	기타 화학제 품 제 조업	화합물, 화학제품 제조	2002-10-07	12월	이건호
7887	189980	KOSDAQ	흥국 에프 엔비	비알코 올음료 및 얼 음 제 조업	과일음료 (에이드 베이스, 스무디, 착즙쥬스 등)	2015-08-07	12월	박철범, 오길영
7888	000540	KOSPI	흥국 화재	보험업	손해보험	1974-12-05	12월	권중원
7891	003280	KOSPI	흥아 해운	해상 운송업	외항화물 운송업 (케미컬 탱커)	1976-06-29	12월	이환구
7892	037440	KOSDAQ	희림	건축기 술, 엔 지니어 링 및 관련 기술 서비스 업	설계 및 감리용역	2000-02-03	12월	정영균, 이목운, 허철호 (각자대 표)

1017 rows × 11 columns

In [75]:

```
#년도별 요약  
df[ "ListingYear" ].describe()
```

Out[75]:

```
count    2500.000000  
mean     2005.676000  
std      12.913657  
min      1956.000000  
25%     1999.000000  
50%     2008.000000  
75%     2016.000000  
max      2022.000000  
Name: ListingYear, dtype: float64
```

In [76]:

df[(df["ListingYear"] == 2018)]

Out[76]:

	Symbol	Market	Name	Sector	Industry	ListingDate	SettleMonth	Representative
97	245620	KOSDAQ	EDGC	자연과학 및 공학 연구개발업	유전체 분석 진단 서비스	2018-06-26	12월	각자대표이사 신상철, 이민섭
159	294870	KOSPI	HDC 현대 산업 개발	건물 건설업	외주주택, 자체 공사, 일반건축, 토목 등	2018-06-12	12월	권순호, 정경구, 하원기 (각자 대표이사)
184	950170	KOSDAQ	JTC	종합 소매업	식품류, 생활용품류, 화장품류 (사후면세업)	2018-04-06	02월	구철모
1801	255220	KOSDAQ	SG	기타 비금속 광물제품 제조업	아스콘, 레미콘	2018-01-26	12월	박창호
1835	285130	KOSPI	SK케미칼	합성고무 및 플라스틱 물질 제조업	PETG 수지, PET 수지, 바이오디젤, 백신	2018-01-05	12월	김철, 전광현
7859	298040	KOSPI	효성 중공업	전동기, 발전기 및 전기변환·공급·제어 장치 제조업	-	2018-07-13	12월	요코타 타케시
7860	298050	KOSPI	효성 첨단 소재	화학섬유 제조업	PET타이어코드	2018-07-13	12월	황정모
7861	298020	KOSPI	효성 티앤씨	화학섬유 제조업	섬유, 산업자재, 화학, 종공업, 건설, 무역, 금융 등	2018-07-13	12월	김용섭

	Symbol	Market	Name	Sector	Industry	ListingDate	SettleMonth	Representative
7862	298000	KOSPI	효성 화학	기초 화학물 질 제 조업	-	2018-07-13	12월	이건종
7866	290270	KOSDAQ	휴네 시온	소프트 웨어 개발 및 공 급업	망간자료 전송(i- oneNet)	2018-08-02	12월	정동섭

117 rows × 11 columns

In [77]:

#서울에 있는 회사 찾기

#2019년에 상장한 서울특별시에 소재한 코스피 종목 찾아봄. 없으면 코스닥

df[(df["ListingYear"]==2018) & (df["Region"]=="서울특별시") & (df["Market"]=="KOSPI")]

Out[77]:

	Symbol	Market	Name	Sector	Industry	ListingDate	SettleMonth	Representative
159	294870	KOSPI	HDC 현대 산업 개발	건설업	외주주 택, 자체 공사, 일 반건축, 토목 등	2018-06-12	12월	권순호, 정경구, 하원기 (각자 대 표이사)
2551	286940	KOSPI	롯데 정보 통신	컴퓨터 프로그 래밍, 시스템 통합 및 관 리업	SM, SI	2018-07-27	12월	노준형
4265	306200	KOSPI	세아 제강	1차 철 강 제 조업	강관	2018-10-05	12월	이휘령, 김석일
4682	293940	KOSPI	신한 알파 리츠	부동산 임대 및 공 급업	비거주 부동산 임대 서 비스업	2018-08-08	03월	이준구
4765	267850	KOSPI	아시 아나 IDT	컴퓨터 프로그 래밍, 시스템 통합 및 관 리업	IT 서비 스, IT 솔 루션	2018-11-23	12월	서근식
4830	018250	KOSPI	애경 산업	기타 화학제 품 제 조업	Age 20's, 루 나, 트리 오, 2080 치약, 순 샘, 제트 등	2018-03-22	12월	채동석, 임재영
5117	010400	KOSPI	우진 아이 엔에 스	건물설 비 설 치 공 사업	반도체/ 디스플레 이 코팅 덕트	2018-09-14	12월	홍경모
5246	088260	KOSPI	이리 스코 크랩	부동산 임대 및 공 급업	부동산투 자회사	2018-06-27	06월	허승재
5728	091810	KOSPI	티웨 이항 공	항공 여객 운송업	저가 항 공운송	2018-08-01	12월	정홍근

	Symbol	Market	Name	Sector	Industry	ListingDate	SettleMonth	Representative
7732	300720	KOSPI	한일 시멘트	시멘트, 석회, 플라스터 및 그 제품 제조업	시멘트, 레미콘, 레미탈	2018-08-06	12월	전근식
7859	298040	KOSPI	효성 중공업	전동 기, 발전기 및 전기변환·공급·제어장치 제조업	-	2018-07-13	12월	요코타 타케시 http://
7860	298050	KOSPI	효성 첨단 소재	화학섬유 제조업	PET타이어코드	2018-07-13	12월	황정모 http://
7861	298020	KOSPI	효성 티앤씨	화학섬유 제조업	섬유, 산업자재, 화학, 종공업, 건설, 무역, 금융 등	2018-07-13	12월	김용섭
7862	298000	KOSPI	효성 화학	기초화학물질 제조업	-	2018-07-13	12월	이건종

개별 종목 시세 수집 및 시각화

FinanceDataReader 라이브러리를 통해 종목별 일별시세를 바로 불러 올수 있음

In [78]:

```
# 삼성전자의 주가 확인

# datareader('종목코드', '시작일', '종료일')
# 일별시세 확인
df=fdr.DataReader('005930', "2017", "2021")
df.head()
```

Out[78]:

	Open	High	Low	Close	Volume	Change
Date						
2017-01-02	35980	36240	35880	36100	93012	0.001665
2017-01-03	36280	36620	36020	36480	147153	0.010526
2017-01-04	36500	36520	36100	36160	159435	-0.008772
2017-01-05	36060	36060	35540	35560	219349	-0.016593
2017-01-06	36180	36440	36040	36200	177619	0.017998

In [79]:

```
#종가
df["Close"]
```

Out[79]:

```
Date
2017-01-02    36100
2017-01-03    36480
2017-01-04    36160
2017-01-05    35560
2017-01-06    36200
...
2020-12-23    73900
2020-12-24    77800
2020-12-28    78700
2020-12-29    78300
2020-12-30    81000
Name: Close, Length: 981, dtype: int64
```

In [80]:

```
df["Close"].plot()
```

Out [80]:

```
<AxesSubplot:xlabel='Date'>
```



In [81]:

```
# csv파일에 저장되어있는 데이터와 라이브러리를 통한 종목별 시세를 같이사용해서 종목별 시세 뽑아봄.
df_krx=pd.read_csv("krx.csv")
df_krx.shape
```

Out [81]:

```
(7895, 10)
```

In [82]:

```
# 우선 종목에 대한 종목코드를 알아야 일별 시세를 불러올수 있음
df_krx[["Symbol", "Name"]].head()
```

Out [82]:

	Symbol	Name
0	060310	3S
1	095570	AJ네트웍스
2	006840	AK홀딩스
3	054620	APS홀딩스
4	265520	AP시스템

In [83]:

df_krx["Name"]

Out [83] :

```

0          3S
1      AJ네트웍스
2      AK홀딩스
3      APS홀딩스
4      AP시스템
...
7890    흥국화재우
7891    흥아해운
7892    희림
7893  하이로즈 단기채권ESG액티브
7894        힙스
Name: Name, Length: 7895, dtype: object

```

In [84]:

df_krx["Name"] == "삼성전자"

Out [84] :

```

0    False
1    False
2    False
3    False
4    False
...
7890   False
7891   False
7892   False
7893   False
7894   False
Name: Name, Length: 7895, dtype: bool

```

In [85]:

df_krx[df_krx["Name"] == "삼성전자"]

Out [85] :

	Symbol	Market	Name	Sector	Industry	ListingDate	SettleMonth	Representative
4139	005930	KOSPI	삼성전자	통신 및 방송 장비 제조업	IMT2000 서비스용 동기식 기지국, 교환국장 비, 데이 터단말 기, 동영 상휴대 폰, 핵심 칩, 반...	1975-06-11	12월	한종희 http://

In [86]:

```
df_krx.loc[df_krx["Name"] == "삼성전자", "Symbol"]
```

Out [86]:

```
4139    005930
Name: Symbol, dtype: object
```

In [87]:

```
#우리는 시리즈 형태를 원하는게 아니고 여기서 종목코드만 뽑아 내고싶음
df_krx.loc[df_krx["Name"] == "삼성전자", "Symbol"].tolist()
```

Out [87]:

```
['005930']
```

In [88]:

```
#위에처럼 리스트 형태가 아니라 0번째 인덱스값 가져옴 string형태로
df_krx.loc[df_krx["Name"] == "삼성전자", "Symbol"].tolist()[0]
```

Out [88]:

```
'005930'
```

In [89]:

```
# 종목명을 item_name변수로 바꿔서 넣어줌
item_name="삼성전자"
df_krx.loc[df_krx["Name"] == item_name, "Symbol"].tolist()[0]
```

Out [89]:

```
'005930'
```

1. 종목명으로 종목코드를 찾을 수 있게 함

In [90]:

```
# item_code_by_item_name 함수씀
def item_code_by_item_name(item_name):
    return item_name
```

In [91]:

```
item_code_by_item_name(item_name)
```

Out [91]:

```
'삼성전자'
```

In [92]:

```
def item_code_by_item_name(item_name):
    item_code=df_krx.loc[df_krx["Name"] == item_name,"Symbol"].tolist()[0]
    return item_code
item_code_by_item_name(item_name)
```

Out [92]:

'005930'

In [93]:

item_code_by_item_name('하이브')

Out [93]:

'352820'

In [94]:

item_code_by_item_name('LG에너지솔루션')

Out [94]:

'373220'

In [95]:

#종목코드 대신에 종목명만 넣으면 바로 시세를 알수있게 바꿈.
 fdr.DataReader(item_code_by_item_name("삼성전자"))

Out [95]:

	Open	High	Low	Close	Volume	Change
Date						
1998-01-23	1234	1263	1200	1262	616720	NaN
1998-01-24	1288	1361	1270	1362	467950	0.079239
1998-01-26	1377	1468	1377	1461	1244670	0.072687
1998-01-30	1577	1577	1561	1578	700090	0.080082
1998-01-31	1702	1702	1577	1616	871390	0.024081
...
2022-03-03	72300	73100	72200	72900	13232638	0.016736
2022-03-04	72700	72700	71200	71500	13409634	-0.019204
2022-03-07	70000	70600	69900	70100	18617138	-0.019580
2022-03-08	68800	70000	68700	69500	15828269	-0.008559
2022-03-10	70800	71200	70600	70700	10972325	0.017266

6000 rows × 6 columns

2. 종목명으로 일별시세까지 같이 불러오기

In [96]:

```
def find_item_list(item_name,year=2021):  
  
    item_code=item_code_by_item_name(item_name)  
    if item_code:  
        df_day=fdr.DataReader(item_code,str(year))  
        return df_day  
    else:  
        return False
```

In [97]:

```
find_item_list("삼성전자",2021)
```

Out[97]:

	Open	High	Low	Close	Volume	Change
Date						
2021-01-04	81000	84400	80200	83000	38655276	0.024691
2021-01-05	81600	83900	81600	83900	35335669	0.010843
2021-01-06	83300	84500	82100	82200	42089013	-0.020262
2021-01-07	82800	84200	82700	82900	32644642	0.008516
2021-01-08	83300	90000	83000	88800	59013307	0.071170
...
2022-03-03	72300	73100	72200	72900	13232638	0.016736
2022-03-04	72700	72700	71200	71500	13409634	-0.019204
2022-03-07	70000	70600	69900	70100	18617138	-0.019580
2022-03-08	68800	70000	68700	69500	15828269	-0.008559
2022-03-10	70800	71200	70600	70700	10972325	0.017266

292 rows × 6 columns

In [98]:

```
stock_daily=find_item_list("삼성전자",2021)
stock_daily
```

Out[98]:

	Open	High	Low	Close	Volume	Change
Date						
2021-01-04	81000	84400	80200	83000	38655276	0.024691
2021-01-05	81600	83900	81600	83900	35335669	0.010843
2021-01-06	83300	84500	82100	82200	42089013	-0.020262
2021-01-07	82800	84200	82700	82900	32644642	0.008516
2021-01-08	83300	90000	83000	88800	59013307	0.071170
...
2022-03-03	72300	73100	72200	72900	13232638	0.016736
2022-03-04	72700	72700	71200	71500	13409634	-0.019204
2022-03-07	70000	70600	69900	70100	18617138	-0.019580
2022-03-08	68800	70000	68700	69500	15828269	-0.008559
2022-03-10	70800	71200	70600	70700	10972325	0.017266

292 rows × 6 columns

In [99]:

```
stock_daily["Close"]
```

Out[99]:

Date	Close
2021-01-04	83000
2021-01-05	83900
2021-01-06	82200
2021-01-07	82900
2021-01-08	88800
...	
2022-03-03	72900
2022-03-04	71500
2022-03-07	70100
2022-03-08	69500
2022-03-10	70700

Name: Close, Length: 292, dtype: int64

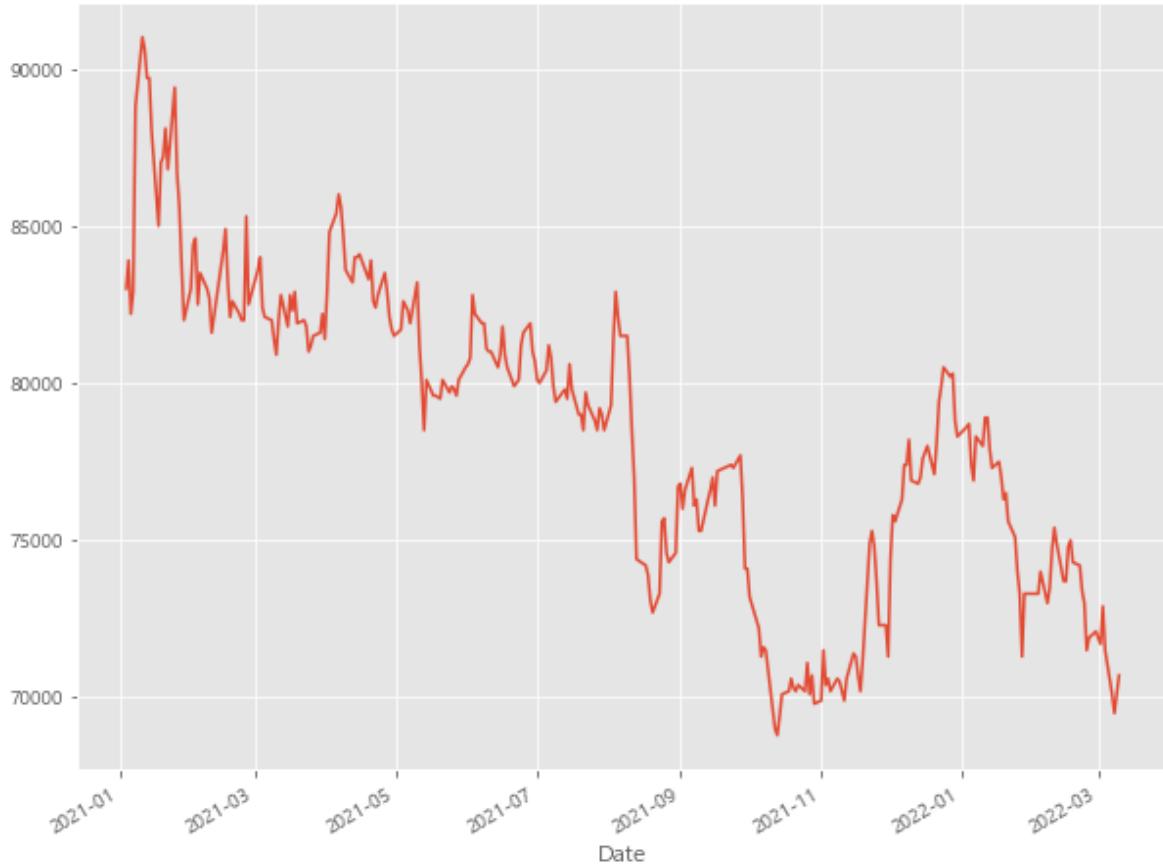
3. 일별시세를 시각화 해보기

In [100]:

```
#삼성전자 시세 시각화  
#종가만 시각화  
plt.figure(figsize=(10,8))  
stock_daily["Close"].plot()
```

Out[100]:

<AxesSubplot:xlabel='Date'>

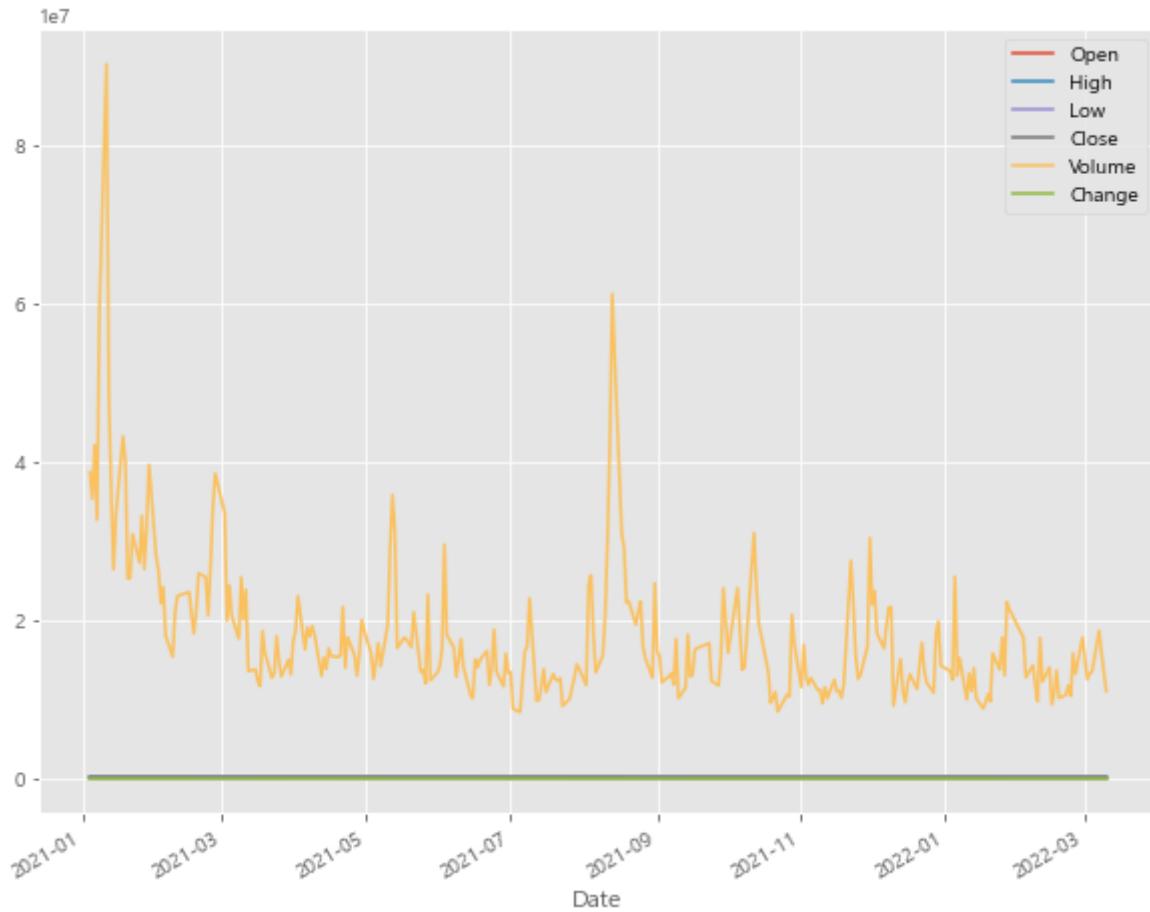


In [101]:

stock_daily.plot(figsize=(10,8))

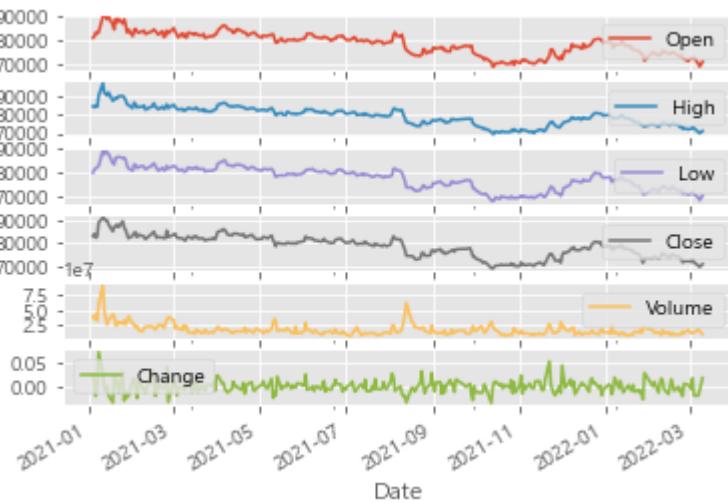
Out[101]:

<AxesSubplot:xlabel='Date'>



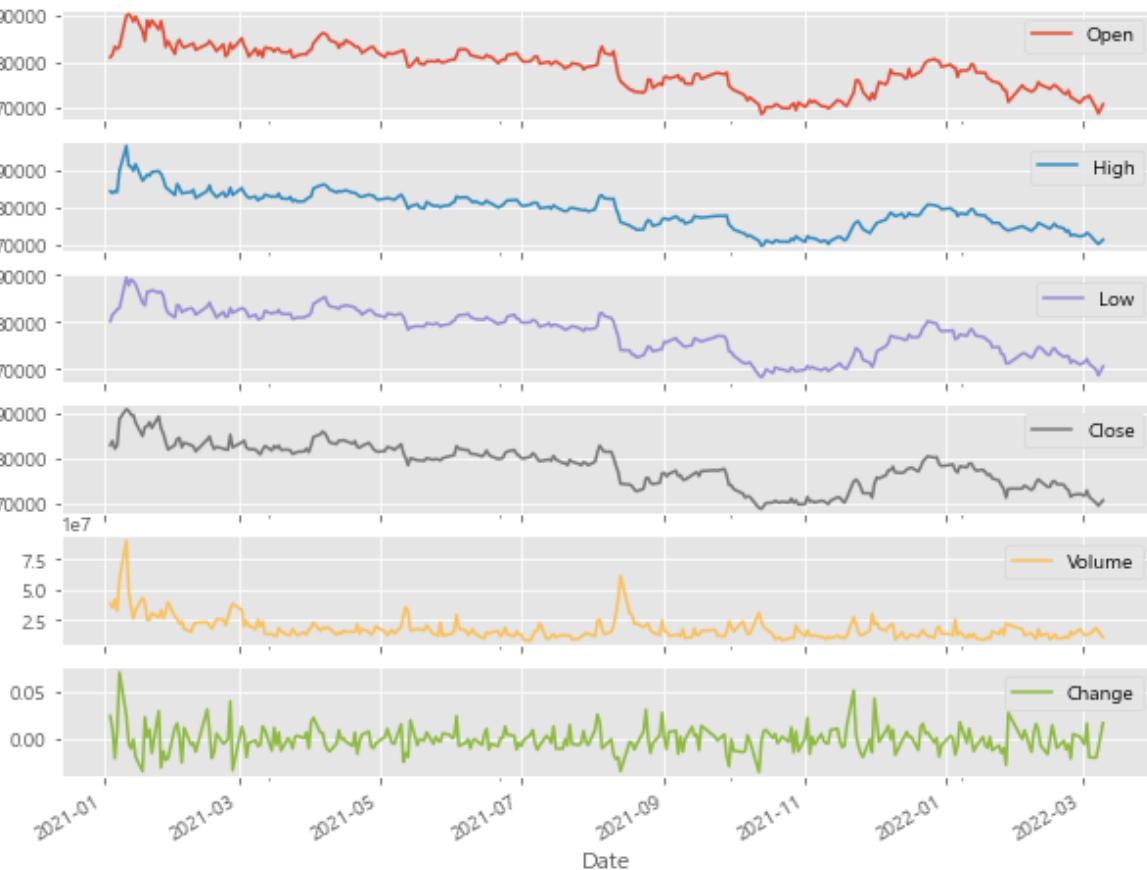
In [102]:

```
#수집한 종목의 선그래프를 pandas로 그림  
# 스케일이 다르기 때문에 subplots를 사용해 그려봄.  
g=stock_daily.plot(subplots=True)
```



In [103]:

```
g=stock_daily.plot(subplots=True,figsize=(10,8))
```

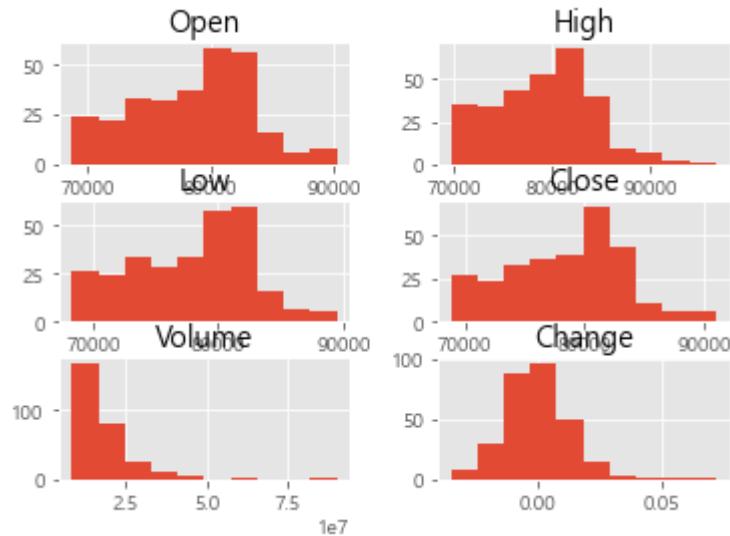


In [104]:

```
#히스토그램도 그려봄  
#삼성전자 시세 히스토그램  
stock_daily.hist()
```

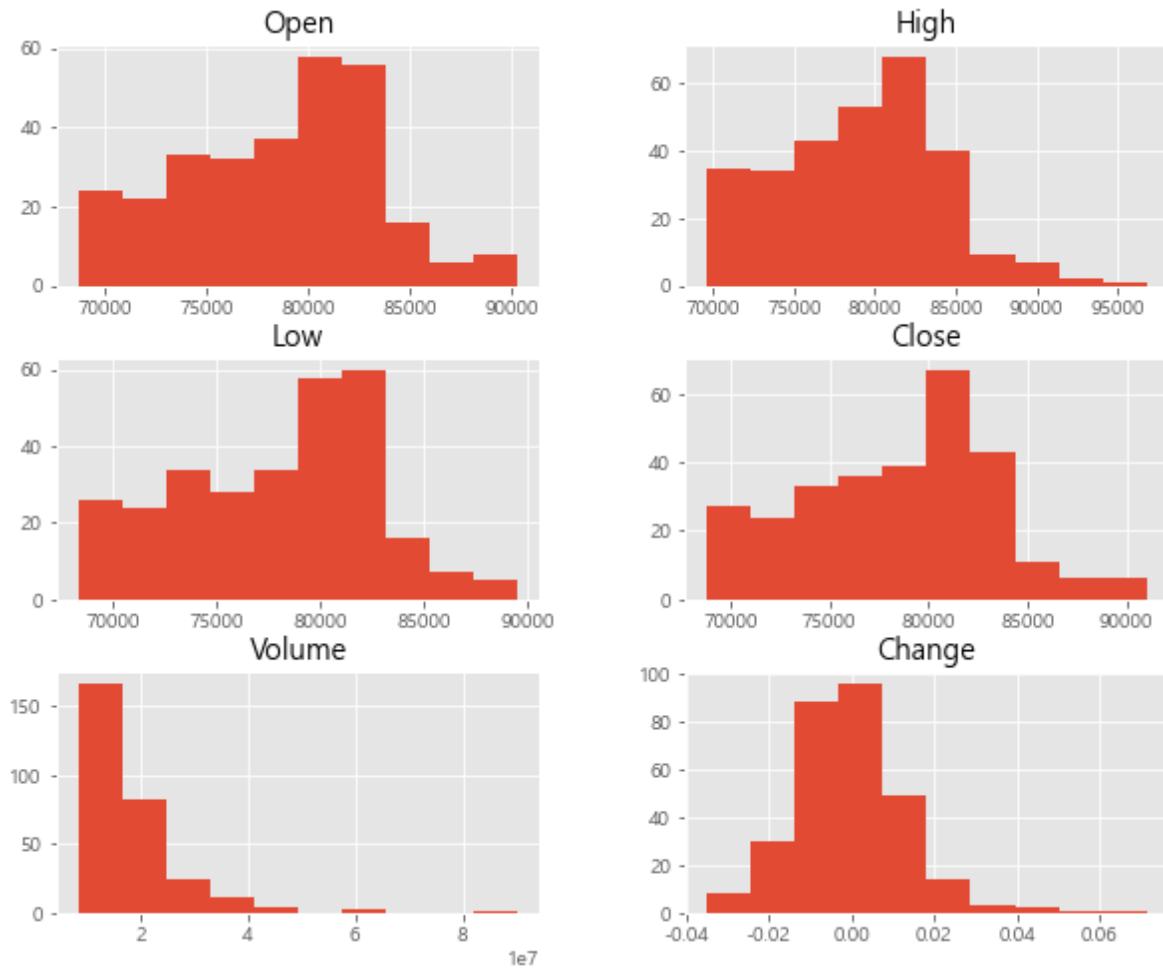
Out[104]:

```
array([[<AxesSubplot:title={'center':'Open'}>,  
       <AxesSubplot:title={'center':'High'}>],  
      [<AxesSubplot:title={'center':'Low'}>,  
       <AxesSubplot:title={'center':'Close'}>],  
      [<AxesSubplot:title={'center':'Volume'}>,  
       <AxesSubplot:title={'center':'Change'}>]], dtype=object)
```



In [105]:

```
h=stock_daily.hist(figsize=(10,8))
```



In [106]:

#막대 더쪼갤

h=stock_daily.hist(figsize=(10,8),bins=20)

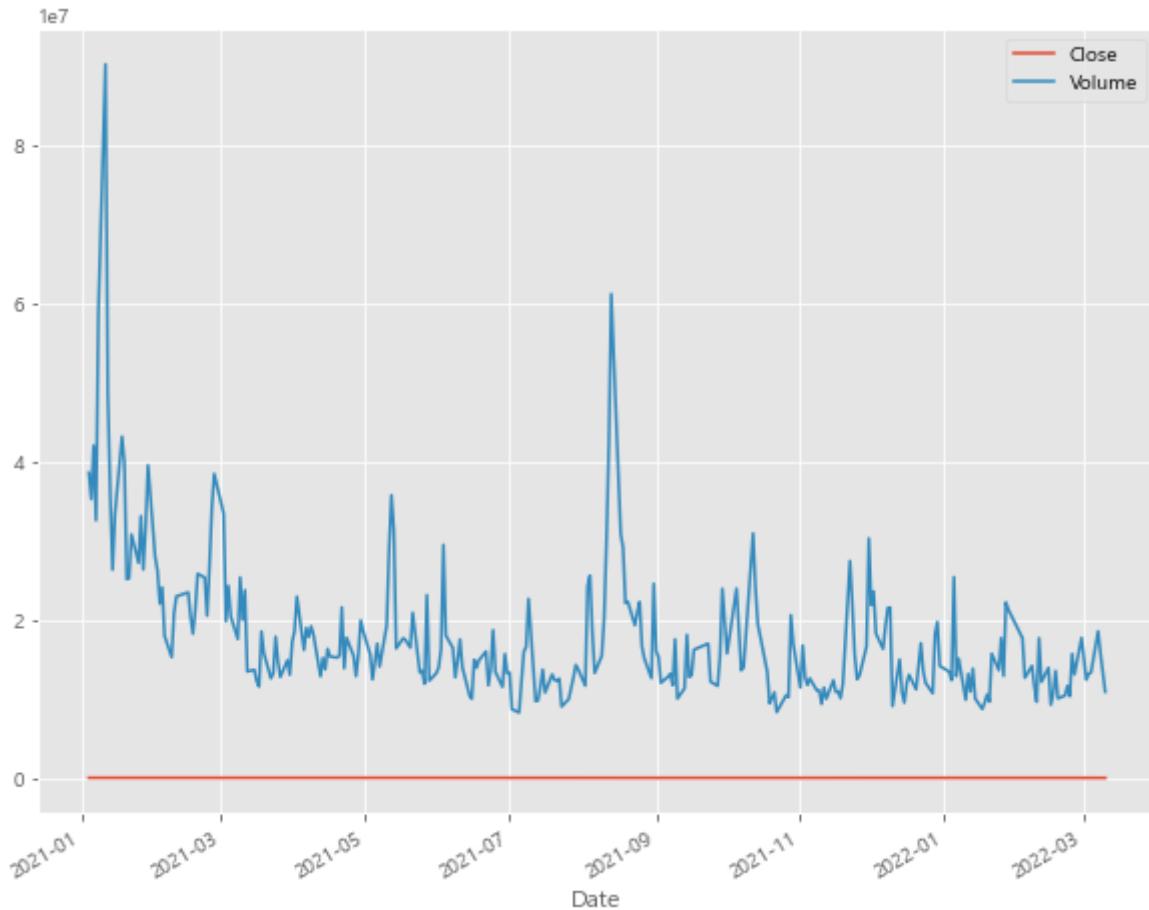


In [107]:

```
#이번엔 종가와 거래량만 가져와서 그려보기  
stock_daily[["Close", "Volume"]].plot(figsize=(10,8))
```

Out[107]:

<AxesSubplot:xlabel='Date'>

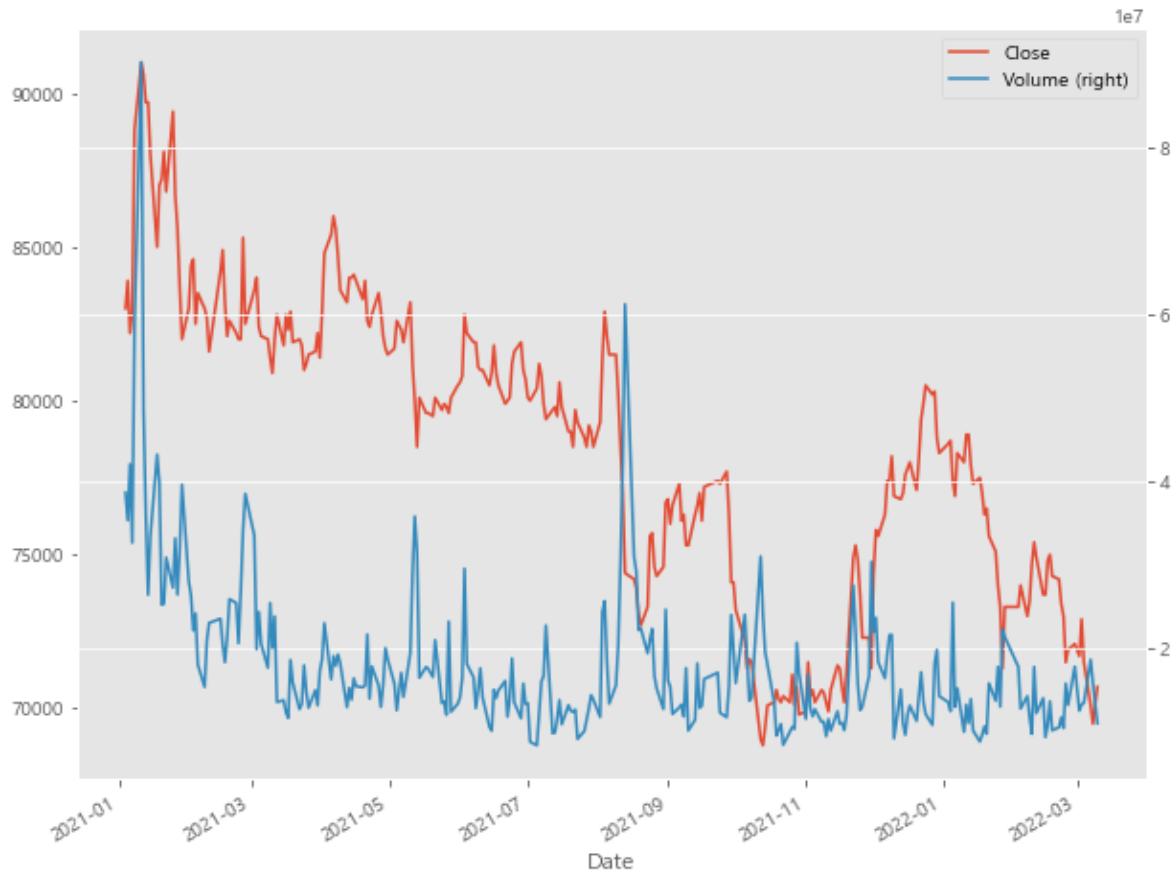


In [108]:

```
# secondary_y 옵션을 통한 종가와 거래량에 대한 2축 그라프  
stock_daily[["Close", "Volume"]].plot(secondary_y="Volume", figsize=(10,8))
```

Out [108]:

<AxesSubplot:xlabel='Date'>



이제 22년 투자할만한 종목 뽑아보기

In [109]:

```
# krx거래소에서 주식데이터의 정보가 들어있는 csv파일 저장 후 불러옴.
import pandas as pd
df1=pd.read_csv("per.csv",encoding='cp949')
df1
```

Out [109]:

	종목코드	종목명	종가	대비	등락률	EPS	PER	선행EPS	선행PER	BPS	PBR	주당배당금	배당수익률
0	060310	AJ	3105	-55	-1.74	16.0	194.06	NaN	NaN	792.0	3.92	0	0.00
1	095570	네트워크	5560	20	0.36	NaN	NaN	NaN	NaN	6089.0	0.91	210	3.78
2	006840	AK	20300	-350	-1.69	NaN	NaN	NaN	NaN	50471.0	0.40	400	1.97
3	054620	APS	11850	-200	-1.66	NaN	NaN	NaN	NaN	8135.0	1.46	0	0.00
4	265520	AK	24000	250	1.05	1736.0	13.82	3190.0	7.52	9041.0	2.65	120	0.50
...
2439	000547	흥국화재2우B	28100	-700	-2.43	NaN	NaN	NaN	NaN	NaN	NaN	0	0.00
2440	000545	흥국화재우	7850	-210	-2.61	NaN	NaN	NaN	NaN	NaN	NaN	0	0.00
2441	003280	흥아해운	3225	-145	-4.30	NaN	NaN	NaN	NaN	NaN	NaN	0	0.00
2442	037440	희림	8510	190	2.28	389.0	21.88	NaN	NaN	4034.0	2.11	150	1.76
2443	238490	힘스	9460	-80	-0.84	1522.0	6.22	NaN	NaN	6872.0	1.38	150	1.59

2444 rows × 13 columns



In [110]:

df1.shape

Out[110]:

(2444, 13)

per(주가수익비율)로 저평가되어있는 주식 뽑아보기

per란 한국말로 주가수익비율을 뜻합니다. 간단하게 말하면

기업의 순이익에 비해 주식 가격이 높게(고평가) 또는 낮게(저평가)를 판단

per가 낮을수록 해당 기업의 가치가 저평가 되어있다고 볼수 있음.

In [111]:

df1[["종목명", "PER"]]

Out[111]:

	종목명	PER
0	3S	194.06
1	AJ네트웍스	NaN
2	AK홀딩스	NaN
3	APS홀딩스	NaN
4	AP시스템	13.82
...
2439	흥국화재2우B	NaN
2440	흥국화재우	NaN
2441	흥아해운	NaN
2442	희림	21.88
2443	힘스	6.22

2444 rows × 2 columns

per주가수익비율이 1보다 작은 종목뽑아봄.

1이라고 나오는건 예를들어 주가가 1주에 10만원인데
1년에 순이익을 1주당 10만원 내면 1이다.

굉장히 주가대비 종목이 저평가 되어있다 볼수있다.

저평가가 되었다는말은 주식가격이 종목 순이익에 비해 싸다는뜻

In [112]:

```
df1[df1["PER"]<1]
```

Out[112]:

		종목코드	종목명	종가	대비	등락률	EPS	PER	선행EPS	선행PER	BPS	PBR	주당배당금	배당수익률
1000	037350	성도이엔지		5160	20	0.39	6074.0	0.85	NaN	NaN	14797.0	0.35	200	3.88
1071	036830	솔브레인홀딩스		27950	-100	-0.36	106557.0	0.26	NaN	NaN	49397.0	0.57	0	0.00
2334	034810	해성산업		11950	0	0.00	25207.0	0.47	NaN	NaN	32729.0	0.37	175	1.46

현실적으로 1밑으로는 너무 없다 보니 3정도로 보고 뽑아봄

In [113]:

```
per=df1[df1["PER"]<3]
per
```

Out[113]:

주식 번호	종목코드	종목명	종가	대비	등락률	EPS	PER	선행EPS	선행PER	BPS	PBR	주당 배당금
26	023460	CNH	2105	-45	-2.09	824.0	2.55	NaN	NaN	3446.0	0.61	5
36	016610	DB금융투자	6080	-30	-0.49	2327.0	2.61	NaN	NaN	18855.0	0.32	30
126	092230	KPX홀딩스	60100	-100	-0.17	34397.0	1.75	NaN	NaN	183257.0	0.33	310
190	040610	SG&G	2195	-55	-2.44	787.0	2.79	NaN	NaN	8110.0	0.27	
194	016250	SGC이테크건설	56800	-900	-1.56	43859.0	1.30	20841.0	2.73	54475.0	1.04	150
209	006120	SK디스커버리	39000	0	0.00	13096.0	2.98	NaN	NaN	95458.0	0.41	100
392	137940	넥스트아이	1195	-10	-0.83	570.0	2.10	NaN	NaN	1528.0	0.78	1
440	008060	대덕	7200	-70	-0.96	3409.0	2.11	NaN	NaN	17300.0	0.42	30
696	016100	리더스코스메틱	2295	-20	-0.86	1313.0	1.75	NaN	NaN	3725.0	0.62	
710	038290	마크로젠	25650	150	0.59	9002.0	2.85	NaN	NaN	18637.0	1.38	30
767	007120	미래아이엔지	398	-3	-0.75	153.0	2.60	NaN	NaN	691.0	0.58	
932	009770	삼정펄프	27550	-650	-2.30	10743.0	2.56	NaN	NaN	84390.0	0.33	100
991	035890	서희건설	1655	-10	-0.60	632.0	2.62	NaN	NaN	2260.0	0.73	4
997	136490	선진	10500	-100	-0.94	3516.0	2.99	NaN	NaN	15558.0	0.67	10
1000	037350	성도이엔지	5160	20	0.39	6074.0	0.85	NaN	NaN	14797.0	0.35	20
1031	024830	세원물산	6280	0	0.00	2199.0	2.86	NaN	NaN	24775.0	0.25	2

종목코드	종목명	종가	대비	등락률	EPS	PER	선행EPS	선행PER	주당	
									BPS	PBR
1071	036830 솔브레인홀딩스	27950	-100	-0.36	106557.0	0.26	NaN	NaN	49397.0	0.57
1125	001720 신영증권	62400	-400	-0.64	22418.0	2.78	NaN	NaN	81242.0	0.77 400
1136	002870 신풍제지	1870	-45	-2.35	1189.0	1.57	NaN	NaN	2884.0	0.65 5
1534	005820 원림	20900	-800	-3.69	7819.0	2.67	NaN	NaN	45998.0	0.45 50
1537	032940 원의	4385	-15	-0.34	4368.0	1.00	NaN	NaN	10289.0	0.43
1594	024800 유성티엔에스	2990	-25	-0.83	2123.0	1.41	NaN	NaN	9044.0	0.33 1
1595	000700 유수홀딩스	5520	-60	-1.08	3202.0	1.72	NaN	NaN	11744.0	0.47 50
1650	078020 이베스트투자증권	7220	-90	-1.23	2413.0	2.99	NaN	NaN	11051.0	0.65 55
1676	353810 이지바이오	4885	-295	-5.69	2476.0	1.97	NaN	NaN	996.0	4.90 7
1679	035810 이지홀딩스	3885	-165	-4.07	1634.0	2.38	NaN	NaN	8129.0	0.48 5
1796	229000 젠큐릭스	13400	-300	-2.19	5605.0	2.39	NaN	NaN	8342.0	1.61
1873	052300 초록뱀컴퍼니	1010	-5	-0.49	370.0	2.73	NaN	NaN	1985.0	0.51
1908	083550 케이엠	8040	-80	-0.99	3174.0	2.53	NaN	NaN	10271.0	0.78 10
1960	002020 코오롱	28650	-250	-0.87	10889.0	2.63	NaN	NaN	52610.0	0.54 50
2022	009410 태영건설	9980	-10	-0.10	9514.0	1.05	3463.0	2.88	15974.0	0.62 32
2062	363280 티와이홀딩스	27500	250	0.92	10193.0	2.70	2870.0	9.58	28162.0	0.98
2283	004150 한솔홀딩스	3210	-30	-0.93	1695.0	1.89	NaN	NaN	11972.0	0.27 12
2287	004960 한신공영	17200	-200	-1.15	9155.0	1.88	NaN	NaN	52869.0	0.33 40
2334	034810 해성산업	11950	0	0.00	25207.0	0.47	NaN	NaN	32729.0	0.37 17

종목코드	종목명	종가	대비	등락률	EPS	PER	선행EPS	선행PER	BPS	PBR	주당
											배당
2420	090710	휴림 로봇	962	-3	-0.31	473.0	2.03	NaN	NaN	789.0	1.22
2426	079980	휴비스	6960	-120	-1.69	2497.0	2.79	NaN	NaN	13659.0	0.51

per가 3보다 낮은 종목명만 뽑아봄.

In [114]:

```
stock_list=per[["종목명", "종목코드"]].values
stock_list
```

Out [114]:

```
array([['CNH ', '023460'],
       ['DB금융투자 ', '016610'],
       ['KPX홀딩스 ', '092230'],
       ['SG&G ', '040610'],
       ['SGC이테크건설 ', '016250'],
       ['SK디스커버리 ', '006120'],
       ['넥스트아이 ', '137940'],
       ['대덕 ', '008060'],
       ['리더스코스메틱 ', '016100'],
       ['마크로젠 ', '038290'],
       ['미래아이엔지 ', '007120'],
       ['삼정펄프 ', '009770'],
       ['서희건설 ', '035890'],
       ['선진 ', '136490'],
       ['성도이엔지 ', '037350'],
       ['세원물산 ', '024830'],
       ['솔브레인홀딩스 ', '036830'],
       ['신영증권 ', '001720'],
       ['신풍제지 ', '002870'],
       ['원림 ', '005820'],
       ['원익 ', '032940'],
       ['유성티엔에스 ', '024800'],
       ['유수홀딩스 ', '000700'],
       ['이베스트투자증권 ', '078020'],
       ['이지바이오 ', '353810'],
       ['이지홀딩스 ', '035810'],
       ['젠큐릭스 ', '229000'],
       ['초록뱀컴퍼니 ', '052300'],
       ['케이엠 ', '083550'],
       ['코오롱 ', '002020'],
       ['태영건설 ', '009410'],
       ['티와이홀딩스 ', '363280'],
       ['한솔홀딩스 ', '004150'],
       ['한신공영 ', '004960'],
       ['해성산업 ', '034810'],
       ['휴림로봇 ', '090710'],
       ['휴비스 ', '079980']], dtype=object)
```

위에서 뽑아본 per가 3보다 낮은 종목들의 종가(Close) 가져오기

In [115]:

```
import pandas as pd

df_list=[fdr.DataReader(code,'2021-02-26')['Close']for name, code in stock_list]
len(df_list)
```

Out[115]:

37

In [116]:

#pd.concat()로 합치기

```
df1=pd.concat(df_list, axis=1)
df1.columns=[name for name, code in stock_list]
df1
```

Out[116]:

	CNH	DB 금융 투자	KPX 홀딩 스	SG&G	SGC 이테 크건 설	SK디 스커 버리	넥스 트아 이	대덕	리더 스코 스메 틱	마크 로zen	...	초록 뱀컴 퍼니	케이 엠	:
Date														
2021-02-26	2585	5730	57500	2450	39446	65400	1315	6700	3380	33650	...	1515	10234	22
2021-03-02	2620	5730	57500	2450	39524	65300	1300	6710	3405	33300	...	1560	10282	22
2021-03-03	2635	5900	57200	2545	39679	63000	1270	6900	3445	32750	...	1590	10378	22
2021-03-04	2610	5850	56400	2630	39524	60300	1245	6830	3565	31550	...	1615	10282	22
2021-03-05	2580	5800	55900	2465	39446	59800	1255	6760	3565	32850	...	1585	10282	21
...
2022-03-03	2160	6280	61100	2285	58800	40850	1240	7540	2455	26400	...	1040	8160	29
2022-03-04	2165	6270	60600	2325	58900	39950	1250	7460	2435	26200	...	1030	8170	29
2022-03-07	2150	6110	60200	2250	57700	39000	1205	7270	2315	25500	...	1015	8120	28
2022-03-08	2065	6020	59900	2200	56600	38850	1245	7170	2265	25400	...	994	8100	28
2022-03-10	2135	6270	60100	2550	57900	39250	1255	7340	2325	26300	...	1020	8170	28

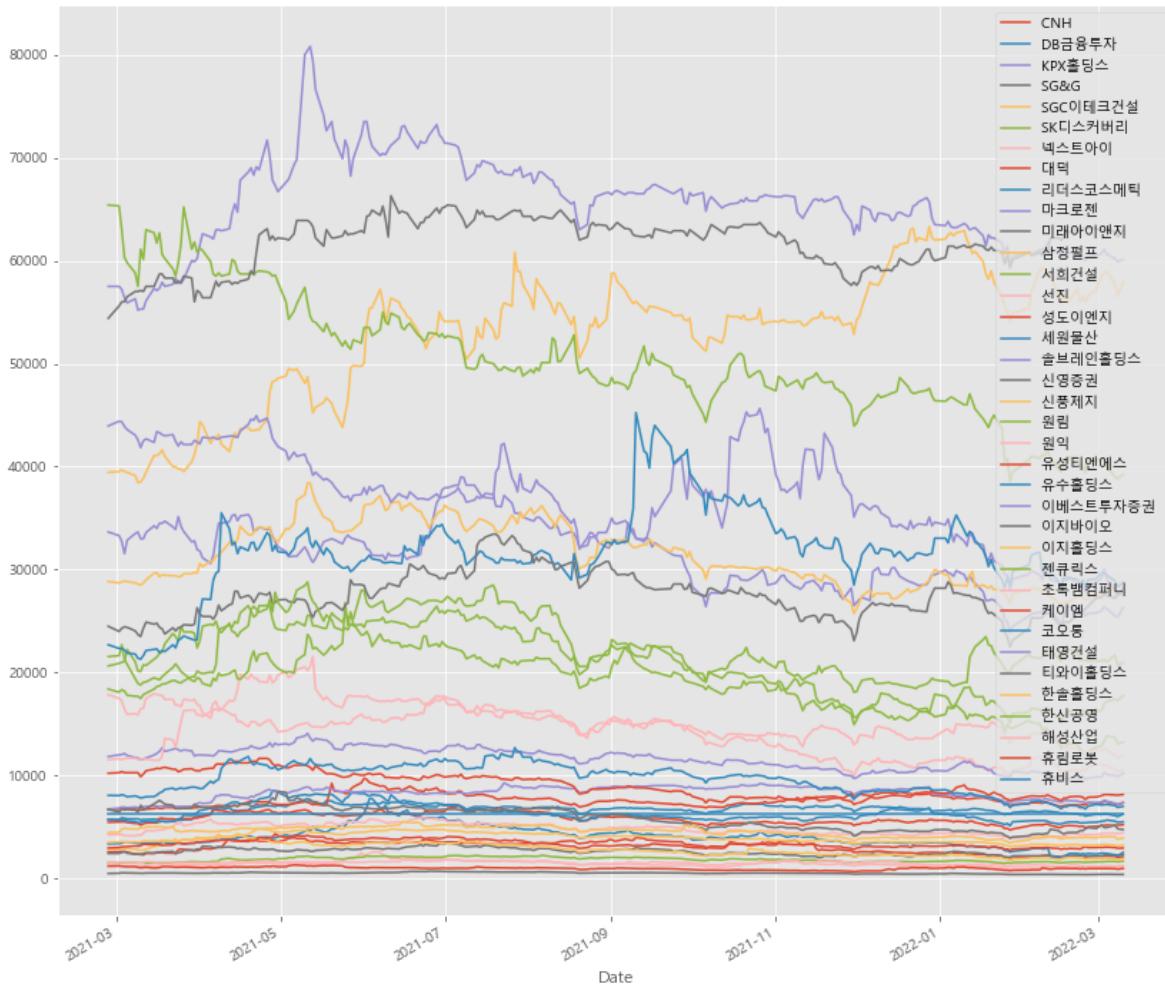
255 rows × 37 columns

In [117]:

df1.plot(figsize=(14, 13))

Out[117]:

<AxesSubplot:xlabel='Date'>



종목이 너무 많으니 조건을 하나 더 추가

PBR 주당 순자산 비율을 뜻함.

간단하게 말하자면 주가가 순자산(자본금과 자본잉여금,

이익잉여금의 합계)에 비해 1주당 몇 배로 거래되고 있는지
측정

pbr이 낮으면 주식의 재무구조가 안정적이라, 기업의 수익률과 성장력
이 높다고 할수있음. 그래서 pbr이 낮고+ per가 낮은 종목으로 더 추려
봄

In [118]:

```
# krx거래소에서 주식데이터의 정보가 들어있는 csv파일 저장 후 불러옴.
import pandas as pd
df3=pd.read_csv("per.csv",encoding='cp949')
df3
```

Out[118]:

	종목코드	종목명	종가	대비	등락률	EPS	PER	선행EPS	선행PER	BPS	PBR	주당배당금	배당수익률
0	060310	3S	3105	-55	-1.74	16.0	194.06	NaN	NaN	792.0	3.92	0	0.00
1	095570	AJ 네트워크	5560	20	0.36	NaN	NaN	NaN	NaN	6089.0	0.91	210	3.78
2	006840	AK 헐딩스	20300	-350	-1.69	NaN	NaN	NaN	NaN	50471.0	0.40	400	1.97
3	054620	APS 헐딩스	11850	-200	-1.66	NaN	NaN	NaN	NaN	8135.0	1.46	0	0.00
4	265520	AP 시스템	24000	250	1.05	1736.0	13.82	3190.0	7.52	9041.0	2.65	120	0.50
...
2439	000547	흥국화재 2우B	28100	-700	-2.43	NaN	NaN	NaN	NaN	NaN	NaN	0	0.00
2440	000545	흥국화재 우	7850	-210	-2.61	NaN	NaN	NaN	NaN	NaN	NaN	0	0.00
2441	003280	흥아해운	3225	-145	-4.30	NaN	NaN	NaN	NaN	NaN	NaN	0	0.00
2442	037440	희림	8510	190	2.28	389.0	21.88	NaN	NaN	4034.0	2.11	150	1.76
2443	238490	힘스	9460	-80	-0.84	1522.0	6.22	NaN	NaN	6872.0	1.38	150	1.59

2444 rows × 13 columns



In [119]:

df3[df3["PBR"]<0.5]

Out[119]:

	종목코드	종목명	종가	대비	등락률	EPS	PER	선행EPS	선행PER	BPS	PBR	주당배수
2	006840	AK홀딩스	20300	-350	-1.69	NaN	NaN	NaN	NaN	50471.0	0.40	400 1.9
6	027410	BGF	5220	10	0.19	285.0	18.32	150.0	34.76	15781.0	0.33	110 2.1
9	138930	BNK금융지주	7360	-110	-1.47	1498.0	4.91	2695.0	2.73	26781.0	0.27	320 4.3
31	000590	CS홀딩스	68100	-1100	-1.59	6657.0	10.23	NaN	NaN	210057.0	0.32	500 0.7
36	016610	DB금융투자	6080	-30	-0.49	2327.0	2.61	NaN	NaN	18855.0	0.32	300 4.9
...
2404	000850	화천기공	32100	-300	-0.93	2053.0	15.64	NaN	NaN	134620.0	0.24	750 2.3
2422	115160	휴맥스	4125	5	0.12	NaN	NaN	NaN	NaN	10255.0	0.40	0 0.0
2423	028080	휴맥스홀딩스	4100	-95	-2.26	NaN	NaN	NaN	NaN	10309.0	0.40	0 0.0
2429	005010	휴스틸	17850	-1450	-7.51	1134.0	15.74	NaN	NaN	54380.0	0.33	200 1.1
2438	000540	흥국화재	3730	0	0.00	268.0	13.92	NaN	NaN	11107.0	0.34	0 0.0

178 rows × 13 columns



In [120]:

```
per=df3[(df3["PER"]<3) & (df3["PBR"]<0.5)]
per
```

Out[120]:

	종목코드	종목명	종가	대비	등락률	EPS	PER	선행EPS	선행PER	BPS	PBR	주당배당금	배당수익률
36	016610	DB금융투자	6080	-30	-0.49	2327.0	2.61	NaN	NaN	18855.0	0.32	300	4.93
126	092230	KPX홀딩스	60100	-100	-0.17	34397.0	1.75	NaN	NaN	183257.0	0.33	3100	5.16
190	040610	SG&G	2195	-55	-2.44	787.0	2.79	NaN	NaN	8110.0	0.27	0	0.00
209	006120	SK디스커버리	39000	0	0.00	13096.0	2.98	NaN	NaN	95458.0	0.41	1000	2.56
440	008060	대덕	7200	-70	-0.96	3409.0	2.11	NaN	NaN	17300.0	0.42	300	4.17
932	009770	삼정펄프	27550	-650	-2.30	10743.0	2.56	NaN	NaN	84390.0	0.33	1000	3.63
1000	037350	성도이엔지	5160	20	0.39	6074.0	0.85	NaN	NaN	14797.0	0.35	200	3.88
1031	024830	세원물산	6280	0	0.00	2199.0	2.86	NaN	NaN	24775.0	0.25	25	0.40
1534	005820	원림	20900	-800	-3.69	7819.0	2.67	NaN	NaN	45998.0	0.45	500	2.39
1537	032940	원의	4385	-15	-0.34	4368.0	1.00	NaN	NaN	10289.0	0.43	0	0.00
1594	024800	유성티엔에스	2990	-25	-0.83	2123.0	1.41	NaN	NaN	9044.0	0.33	15	0.50
1595	000700	유수홀딩스	5520	-60	-1.08	3202.0	1.72	NaN	NaN	11744.0	0.47	500	9.06
1679	035810	이지홀딩스	3885	-165	-4.07	1634.0	2.38	NaN	NaN	8129.0	0.48	50	1.29
2283	004150	한솔홀딩스	3210	-30	-0.93	1695.0	1.89	NaN	NaN	11972.0	0.27	120	3.74
2287	004960	한신공영	17200	-200	-1.15	9155.0	1.88	NaN	NaN	52869.0	0.33	400	2.33
2334	034810	해성산업	11950	0	0.00	25207.0	0.47	NaN	NaN	32729.0	0.37	175	1.46

In [121]:

```
stock_list=per[["종목명", "종목코드"]].values  
stock_list
```

Out[121]:

```
array([['DB금융투자 ', '016610'],  
       ['KPX홀딩스 ', '092230'],  
       ['SG&G ', '040610'],  
       ['SK디스커버리 ', '006120'],  
       ['대덕 ', '008060'],  
       ['삼정펄프 ', '009770'],  
       ['성도이엔지 ', '037350'],  
       ['세원물산 ', '024830'],  
       ['원림 ', '005820'],  
       ['원익 ', '032940'],  
       ['유성티엔에스 ', '024800'],  
       ['유수홀딩스 ', '000700'],  
       ['이자홀딩스 ', '035810'],  
       ['한솔홀딩스 ', '004150'],  
       ['한신공영 ', '004960'],  
       ['해성산업 ', '034810]], dtype=object)
```

In [122]:

```
import pandas as pd  
  
df_list=[fdr.DataReader(code,'2019')[['Close']]for name, code in stock_list]  
len(df_list)
```

Out[122]:

16

In [123]:

#pd.concat()로 합치기

```
df3=pd.concat(df_list, axis=1)
df3.columns=[name for name, code in stock_list]
df3
```

Out[123]:

	DB 금융 투자	KPX 홀딩 스	SG&G	SK디 스커 버리	대덕	삼정 펄프	성도 이엔 지	세원 물산	원림	원익	유성 티엔 에스	유수 홀딩 스	이지 홀딩 스	...
Date														...
2019-01-02	4320	61400	2330	25600	8933	33400	4910	5210	17047	4120	2505	6370	5846	5
2019-01-03	4300	60600	2380	25000	9195	33350	4810	5180	17047	3970	2480	6300	5826	5
2019-01-04	4360	60600	2495	24900	9275	33350	4870	5290	17047	3880	2530	7000	5866	4
2019-01-07	4485	61000	2345	24650	9386	33450	4870	5290	17430	4185	2615	7110	6116	4
2019-01-08	4460	61000	2285	24850	9305	33050	4845	5250	17430	4350	2650	7080	6056	4
...
2022-03-03	6280	61100	2285	40850	7540	27650	5170	6280	21250	4395	3055	5600	4060	3
2022-03-04	6270	60600	2325	39950	7460	27800	5200	6280	21050	4400	3070	5680	4105	3
2022-03-07	6110	60200	2250	39000	7270	28200	5140	6280	21700	4400	3015	5580	4050	3
2022-03-08	6020	59900	2200	38850	7170	27450	5190	6280	20750	4370	2945	5460	3830	3
2022-03-10	6270	60100	2550	39250	7340	27650	5220	6280	20950	4430	2965	5510	3930	3

786 rows × 16 columns



종목간의 가격차이가 커서 서로 비교어려워서, `secondary_y`옵션을

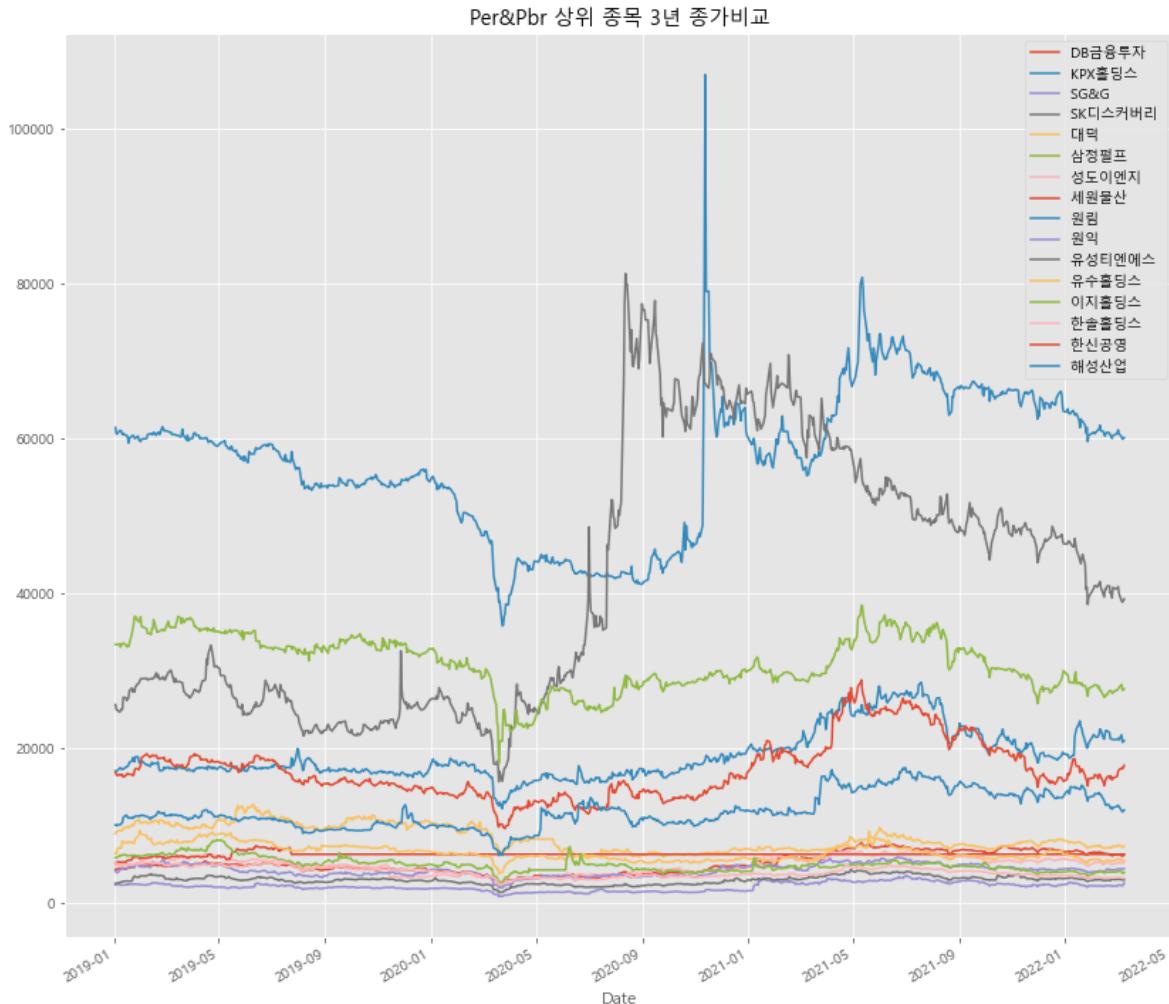
활용하여 서로 다른 영역을 비교하기 용이

In [124]:

```
df3.plot(figsize=(14, 13))
plt.title('Per&Pbr 상위 종목 3년 종가비교')
```

Out[124]:

Text(0.5, 1.0, 'Per&Pbr 상위 종목 3년 종가비교')



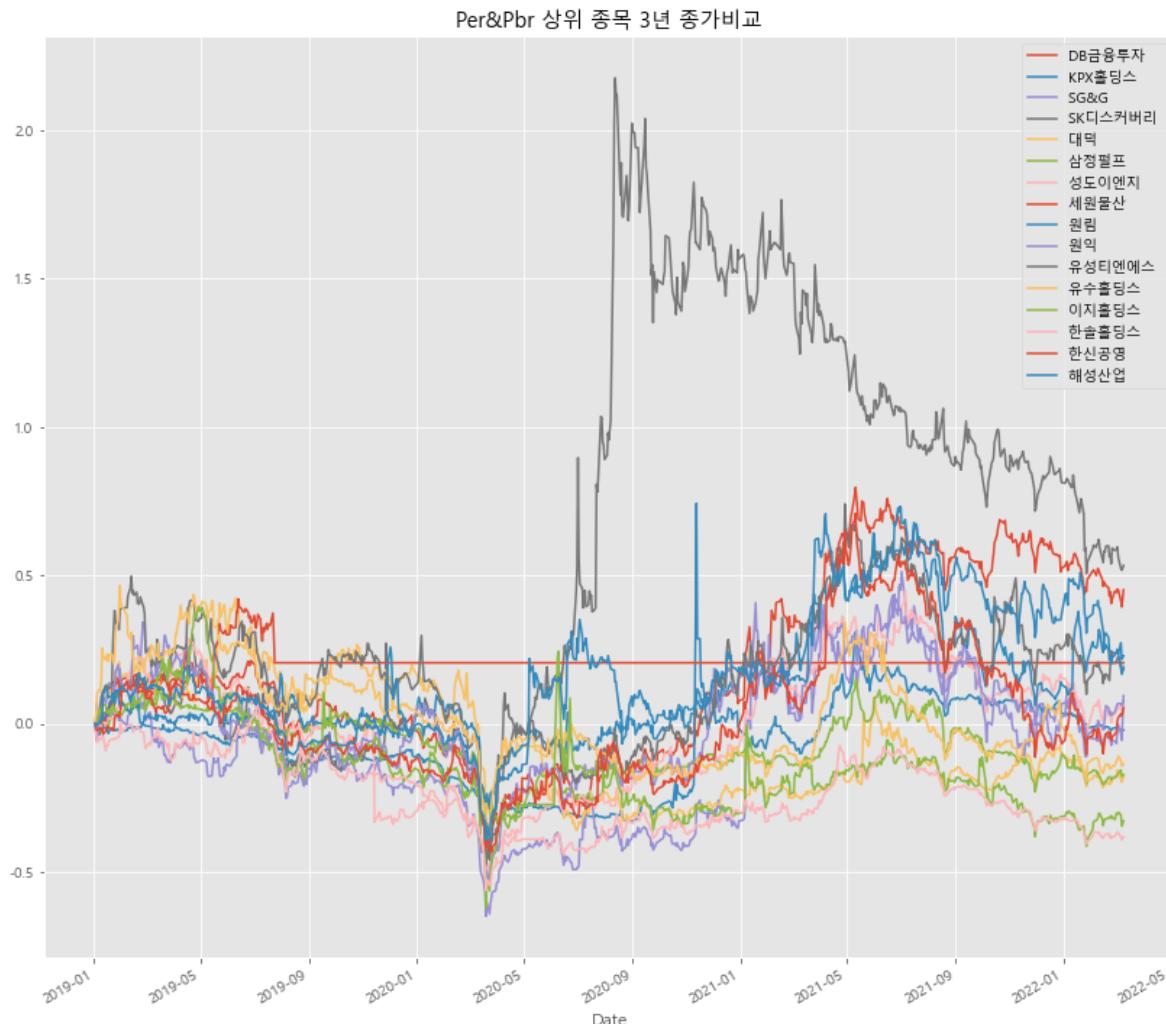
종목들이 서로 다른 행보를 가지는 거처럼 보이지만 시작점이 다르기 때문에
 시간에 따른 변화를 보기 쉽지 않기 때문에
 시작점을 0으로 동일하기 맞추고 시간에 따른 수익률을 보기 위해
 첫번째 행(row)으로 모든 행(row)을 나누고 1.0을 빼줌

In [125]:

```
df_plot=df3/df3.iloc[0] - 1.0  
df_plot.plot(figsize=(14,13))  
plt.title('Per&Pbr 상위 종목 3년 종가비교')
```

Out [125]:

Text(0.5, 1.0, 'Per&Pbr 상위 종목 3년 종가비교')

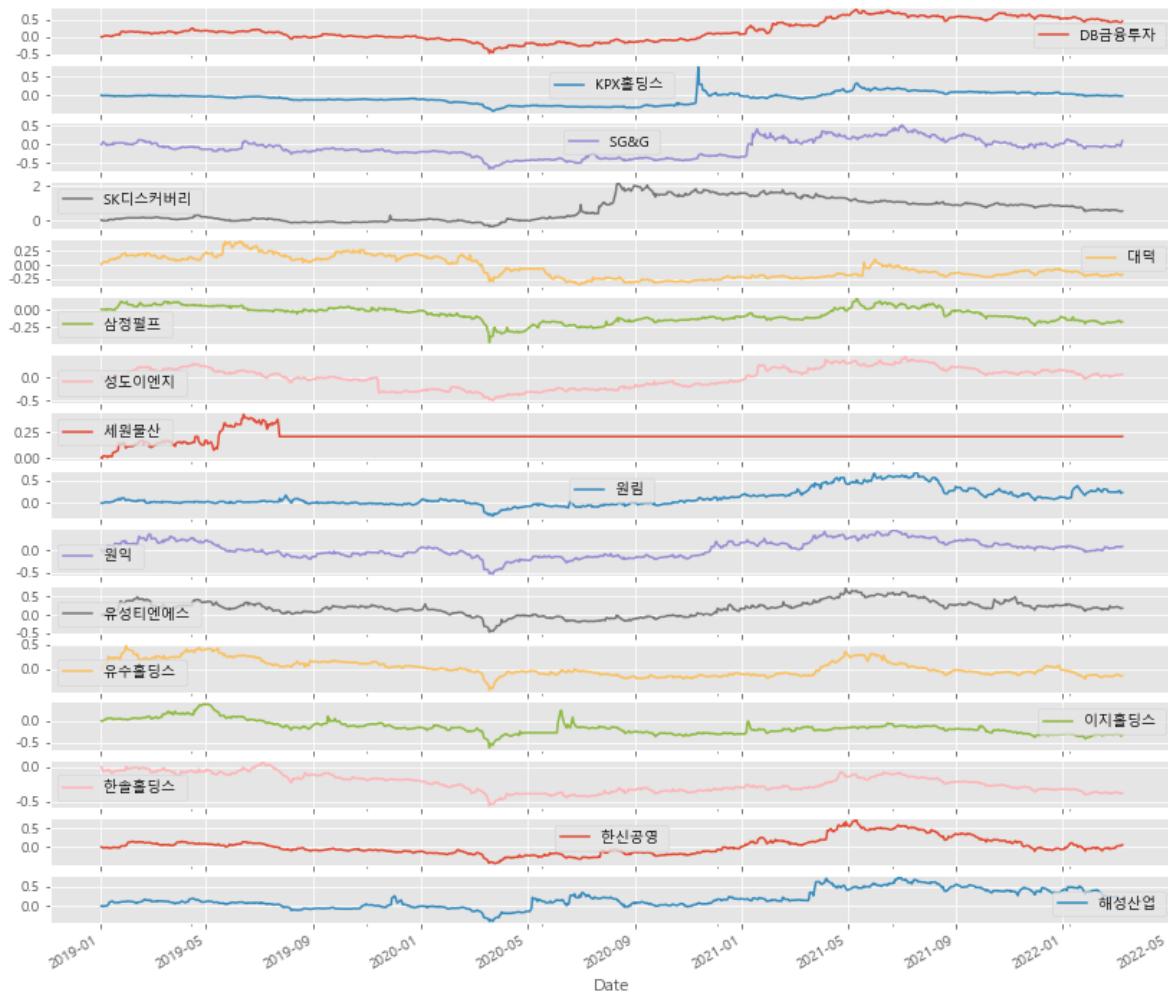


In [126]:

```
df3=df_plot.plot(subplots=True, figsize=(14, 13))
df3
```

Out[126]:

```
array([<AxesSubplot:xlabel='Date'>, <AxesSubplot:xlabel='Date'>,
       <AxesSubplot:xlabel='Date'>, <AxesSubplot:xlabel='Date'>],
      dtype=object)
```



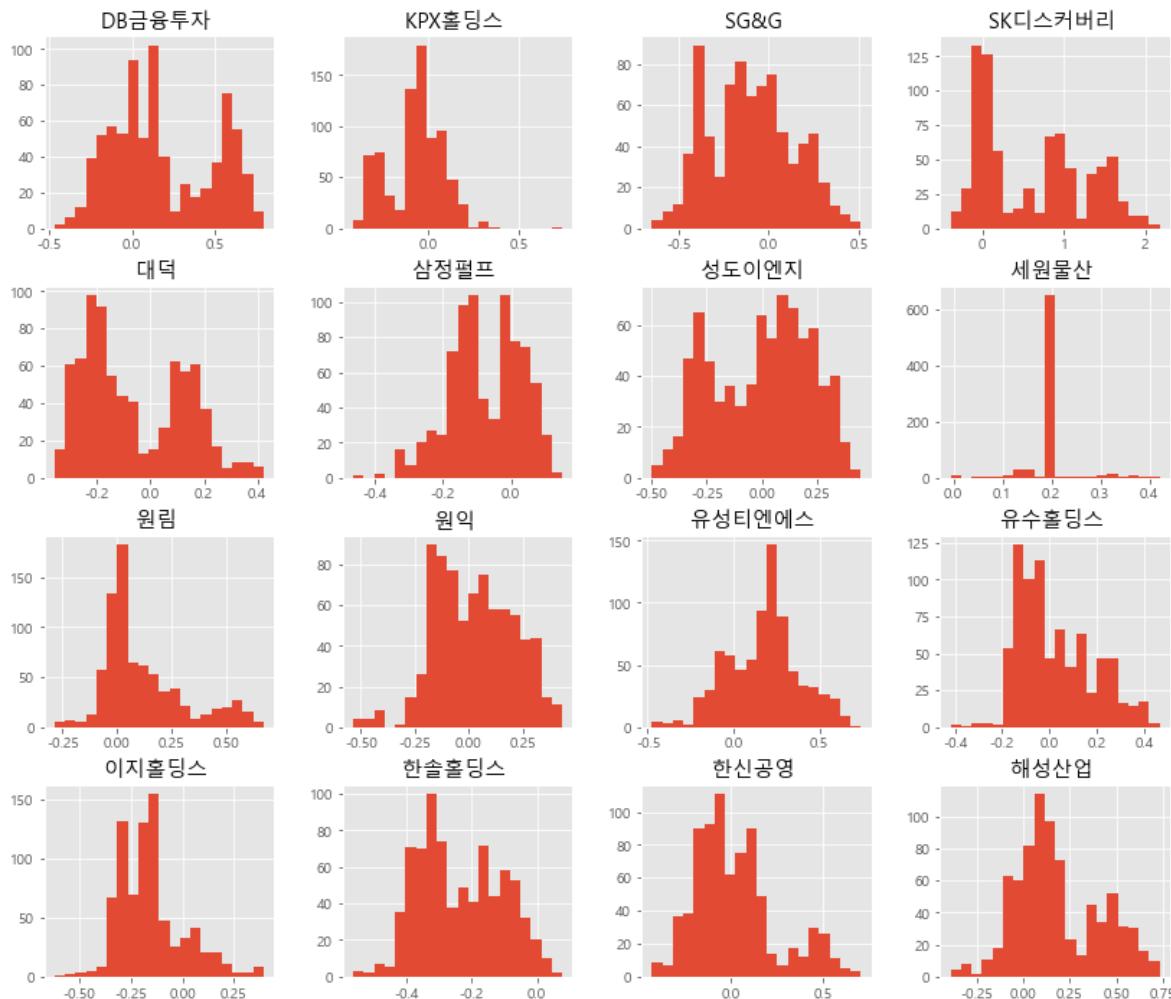
In [127]:

전 종목 종가 빈도수

```
import matplotlib.pyplot as plt
df_plot.hist(figsize=(14, 12), bins=20)
```

Out[127]:

```
array([[<AxesSubplot:title={'center': 'DB금융투자'}>,
       <AxesSubplot:title={'center': 'KPX홀딩스'}>,
       <AxesSubplot:title={'center': 'SG&G'}>,
       <AxesSubplot:title={'center': 'SK디스커버리'}>],
      [<AxesSubplot:title={'center': '대덕'}>,
       <AxesSubplot:title={'center': '삼정펄프'}>,
       <AxesSubplot:title={'center': '성도이엔지'}>,
       <AxesSubplot:title={'center': '세원물산'}>],
      [<AxesSubplot:title={'center': '원림'}>,
       <AxesSubplot:title={'center': '원익'}>,
       <AxesSubplot:title={'center': '유성티엔에스'}>,
       <AxesSubplot:title={'center': '유수홀딩스'}>],
      [<AxesSubplot:title={'center': '이지홀딩스'}>,
       <AxesSubplot:title={'center': '한솔홀딩스'}>,
       <AxesSubplot:title={'center': '한신공영'}>,
       <AxesSubplot:title={'center': '해성산업'}>]], dtype=object)
```

Type *Markdown* and *LaTeX*: α^2

