

٠٢ TSE_fetch

۲۹ مهر ۱۴۰۱

۱ دریافت اطلاعات نمادهای بورس تهران

```
[ ]: import pandas as pd
import pytse_client as tse # https://pypi.org/project/pytse-client/
```

۱.۱ دریافت قیمت نمادها

قیمت‌های ابتدایی، پایانی، بیشترین و کم‌ترین به همراه تعدیل‌شده‌شان را برای ۶ سهم دریافت می‌کنیم. شش سهم: **شپنا، کگل، خاور، کرمان، وبصادر و فارس**

برای اینکه کار کردن با دیتای خروجی راحت‌تر شود، سعی می‌کنیم تا خروجی را به یک multi-index دیتافریم تبدیل کنیم. (همانند خروجی پکیج yfinance هنگامی که دیتای چند نماد را با هم دریافت می‌کنیم)

در گزارش pdf که توسط مبدل لاتک خودکار تولید می‌شود، کاراکترهای فارسی حذف شده‌اند.

```
[2]: tickers = [' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '][
prices_dict = tse.download(symbols=tickers, adjust=True)
prices_dict_reform = {(outerKey, innerKey):
                        values for outerKey, innerDict
                        in prices_dict.items() for innerKey, values
                        in innerDict.iteritems()}
data = pd.DataFrame(prices_dict_reform)
```

دیتای همه ۶ نماد در یک دیتافریم ذخیره شده‌اند و می‌توانیم به این صورت، مثلاً دیتای سهم «خاور» را ببینیم:

در گزارش pdf که توسط مبدل لاتک خودکار تولید می‌شود، کاراکترهای فارسی حذف شده‌اند.

```
[7]: cols = ['open', 'high', 'low', 'close', 'adjClose']
data[' '].set_index('date')[cols].tail()
```

```
[7]:
```

	open	high	low	close	adjClose
date					
2022-10-15	2268.0	2329.0	2241.0	2241.0	2253.0
2022-10-16	2186.0	2200.0	2186.0	2186.0	2186.0
2022-10-17	2121.0	2172.0	2121.0	2150.0	2126.0
2022-10-18	2154.0	2164.0	2063.0	2073.0	2094.0
2022-10-19	2074.0	2107.0	2032.0	2070.0	2054.0

برای نوشتن در خروجی، ابتدا اسم فایل و فولدر مربوطه را انتخاب می‌کنیم. سپس در صورت موجود نبودن فولدر، آن را می‌سازیم. خروجی این مرحله در این مقصد ذخیره می‌شود: ./excel_files/۰۲_TSE_prices.xlsx

```
[4]: import os

dir_name = 'excel_files'
os.makedirs(rf"./{dir_name}", exist_ok=True)
output_file_name = '02_TSE_prices'
path = rf"./{dir_name}/{output_file_name}.xlsx"
```

در یک حلقه for هر کدام از سهم‌ها را در شیت جداگانه‌ای می‌نویسیم.

```
[5]: writer = pd.ExcelWriter(path, engine = 'xlsxwriter')
for col in data.columns.levels[0].tolist():
    data[col].to_excel(writer, sheet_name = col)
writer.save()
```