



# Работа с внешними финансовыми данными



# Задача

---



**Загрузка финансовых данных из  
внешних источников**

# Выгрузка внешних данных

---



1. Переходим на сайт [quote.ru](https://quote.ru)
2. Выбираем интересующий [наш тикер](#)
3. Загружаем данные
4. Переносим в папку с проектом

```
import pandas_datareader.data as web
import pandas as pd
import datetime as dt
import matplotlib.pyplot as plt
```

# Pandas Dataframes



Index

	Time	Low	High	Open	Last	DayVol	DayVal	Deals	
0	0	2017-02-2	8811	9220	8827	9190	179430	1623488512	16722
1	1	2017-02-2	8993	9649	9151	9133	172263	1577417289	13278
2	2	2017-03-0	9027	9293	9053	9150	65407	600085107	6747
.	3	2017-03-0	9130	9381	9214	9250	106834	987810025	11508
.	4	2017-03-0	9152	9310	9211	9175	63469	585202638	5587
.	5	2017-03-0	9150	9300	9150	9250	61228	566125921	5855
	6	2017-03-0	9128	9271	9251	9233	80690	742879577	7843
	7	2017-03-0	9044	9280	9230	9061	245125	2253231699	20282
	8	2017-03-10	9100	9250	9100	9155	97717	895671917	8885
	9	2017-03-13	9101	9270	9155	9235	106811	982190135	14359
	10	2017-03-14	9171	9379	9250	9270	96098	888607029	12184

# Pandas Dataframes

---



## 1. Загрузка данных:

```
web.DataReader(tickers, dataSource, startDate, endDate)
```

### **DataSource:**

1. Google Finance
2. [Morningstar](#)
3. [FermaFrench](#)
4. [FRED](#)
5. [World Bank](#)
6. [Eurostat](#)
7. [Stooq](#)
8. [Moex](#)

```
from pandas_datareader import wb  
wb.download(indicator='NY.GDP.PCAP.KD', country=['RU'], start=2005, end=2008)
```

# Pandas Dataframes

---



1. Загрузка данных:

```
Import quandl  
mydata = quandl.get("BANKRUSSIA/KEYECIND")
```

# Задача

---



**Работа с табличными данными в  
памяти**

# Лямбды:



```
def calculateTheDiff(row):  
    return row['Value']/row['Prev'] - 1  
  
inflation['pctChange0'] = inflation.apply(lambda row: calculateTheDiff(row), axis=1)
```

Для каждого элемента в inflation.

Тот же самый синтаксис что и:

```
for row in inflation:  
    inflation[row]['pctChange0'] = calculateTheDiff(row)
```



# Лямбды:

---



`inflation['pctChange1'] = inflation.apply(lambda row: (row['Value']/row['Prev'] - 1), axis=1)`



## Статистическая обработка финансовых данных

# Задача

---



**Вычисление основных метрик**

# Задача

---



**Графическое изображение данных**



**Интеграция сложных  
математических вычислений в  
эксель, через нативные функции**