***Проект для анализа и визуализации данных об акциях***

**Общий обзор**  
Этот проект предназначен для загрузки исторических данных об акциях и их визуализации. Он использует библиотеку yfinance для получения данных и matplotlib для создания графиков. Пользователи могут выбирать различные тикеры и временные периоды для анализа, а также просматривать движение цен и скользящие средние на графике.  
  
**Структура и модули проекта**

**1. main.py:**  
- Является точкой входа в программу.  
  
- Запрашивает у пользователя тикер акции, временной период и порог анализа колебаний цены; загружает данные, обрабатывает их и выводит результаты на консоль, в виде графика и сохраненных файлов.  
  
**2. data\_download.py:**  
- Отвечает за загрузку данных об акциях.  
  
- Содержит функции:

- извлечения данных об акциях из интернета,

- расчёта скользящего среднего,

- расчета среднего значения цены за период,

- оценки степени колебаний цены и сравнения с заданным

порогом,

- сохранение файла данных в формате .csv.  
  
  
**3. data\_plotting.py:**  
- Отвечает за визуализацию данных.  
  
- Содержит функции для создания и сохранения графиков цен закрытия и скользящих средних.

**Описание функций**

**1. main.py:**  
- main(): Основная функция, управляющая процессом загрузки, обработки данных и их визуализации. Запрашивает у пользователя ввод данных, вызывает функции загрузки и обработки данных, а затем передаёт результаты на визуализацию.  
  
**2. data\_download.py:**  
- fetch\_stock\_data(ticker, period): Получает исторические данные об акциях для указанного тикера и временного периода. Возвращает DataFrame с данными.  
  
- add\_moving\_average(data, window\_size): Добавляет в DataFrame колонку со скользящим средним, рассчитанным на основе цен закрытия.  
  
- calculate\_and\_display\_average\_price(data):

"""

function calculate average price of "Closed" data.

Result is printed out in console.

"""

- notify\_if\_strong\_fluctuations(data, threshold):

"""

function get DataFrame(data), calculate minimum and maximum

price, and compare ratio of minimum and maximum with

given threshold. Result is printed out in console.

"""

- export\_data\_to\_csv(data, filename):

"""

function get DataFrame(data) and save

data as filename file in .csv format

"""

**3. data\_plotting.py:**  
- create\_and\_save\_plot(data, ticker, period, filename): Создаёт график, отображающий цены закрытия и скользящие средние. Предоставляет возможность сохранения графика в файл. Параметр filename опционален; если он не указан, имя файла генерируется автоматически.

**Пошаговое использование**  
1. Запустите main.py.  
  
2. Введите интересующий вас тикер акции (например, 'AAPL' для Apple Inc).

3. Введите желаемый временной период для анализа (например, '1mo' для данных за один месяц).

4. Введите порог колебаний средней цены акции в %.  
  
  
5. Программа

- обработает введённые данные,

- загрузит соответствующие данные об акциях,

- сохранит данные в .csv файле,

- рассчитает скользящее среднее,

- построит и сохранит график,

- выведет данные расчетов средней цены,

- выведет данные о минимальной и максимальной цене и результаты анализа относительно порога.