

Тестовое Backend Developer

У Twitter есть специальный API, подписаться на "бесконечный" поток твитов по заданным заранее фильтрам. Ваша задача написать клиент, который будет:

- Загружать набор фильтров из конфигурационного файла
- Применять актуальные фильтры к настройкам потока
- Подписываться на поток твитов и обрабатывать его

Обработка твитов подразумевает под собой подсчет метрик по тексту твитов и дальнейший экспорт этих метрик с помощью специальной библиотеки `io.prometheus:simpleclient`. Выставленные запущенным приложением метрики можно посмотреть выполнив команду

```
$ curl localhost:8080/metrics
```

Либо открыв <http://localhost:8080/metrics> в браузере.

Вместе с заданием предлагается заготовка проекта, в которой частично реализованы необходимые механизмы, чтобы было понятно с чего начинать. Так в стартовом проекте уже реализовано:

- Минимальный набор библиотек для реализации задания
- Загрузка конфигурационного файла с правилами для Twitter API
- Получение Api Key и Secret Key через `System.getProperty()`
- Пример создания метрик для Prometheus и описание всех необходимых метрик
- Настроен экспорт метрик с помощью `io.prometheus:simpleclient_httpserver`.

NOTE Все классы в рамках проекта можно менять по своему усмотрению.

NOTE Для выполнения задания **потребуется** зарегистрировать свое приложение в Twitter.com и пользоваться собственными Api Key и Secret Key.

Требования

Готовое решение, должно быть упаковано в ZIP архив, содержащий директорию с именем `java-developer-test-[первая_буква_имени+фамилия]`. Пример: `java-developer-test-aretrov`. Папка должна представлять собой Git репозиторий с историей коммитов. Проверяться будет на Java 11.

К решению должны прилагаться следующие файлы

- Файл с условием тестового задания - этот файл.

- Файл [Readme.md](#), содержащий инструкции по запуску. Дополнительную информацию относительно сборки и запуска.

Функциональные требования

- Реализован интерфейс `TwitterStreamConnector`, согласно требованиям описанным в JavaDoc этого интерфейса
- Реализован метод обновления метрик `PrometheusTweetAggregator#process`, согласно требованиям описанным в JavaDoc этого класса и метода соответственно.
- Реализован метод `TextUtils::wordCount` и проходят тесты на его реализацию (см. класс `TextUtilsTest`)

Полезные ссылки

- [Filtered stream. Quick Start](#) — основная документация по работе с Twitter Api, требуемым в задаче.
- [POST oauth2/token](#) — получение Bearer Token для выполнения запросов к API
- [client_java#instrumenting](#) — коротко о том, как экспортируются метрики с помощью