

## **Инструкция по установке Q.FloweR**

Exported on Jul 08, 2019

**Запуск сервиса как standalone Java приложения.**

Для запуска сервиса в этом режиме выполните:

```
java -jar flower.jar
```

**Параметры «Q.FloweR»****Параметры модуля Q.AuthE**

Параметры сервиса задаются одним из способов:

- переменными окружения
- файлом конфигурации yml
- параметрами командной строки, например,

```
--spring.datasource.driverClassName=org.postgresql.Driver
```

Ниже приведен список переменных окружения используемых для задания параметров сервиса, в скобках указаны наименования параметров для командной строки и файла конфигурации yml.

**Общие параметры**

SERVICE\_PORT (server.port) – IP-порт сервиса, значение по умолчанию: 8080

**Параметры service discovery**

EUREKA\_CLIENT\_ENABLED (eureka.client.enabled) – признак использования сервиса, значение по умолчанию: true

EUREKA\_URL (eureka.client.serviceUrl.defaultZone) – адрес подключения к сервису

EUREKA\_PREFEREDIP\_ENABLED (eureka.instance.preferIpAddress) – признак использования IP-адреса вместо имени хоста при регистрации сервиса, значение по умолчанию: true

**Параметры провайдеров аутентификации и авторизации**

OAUTH2\_CLIENT\_ID (security.oauth2.client.clientId) - ID клиента для авторизации по протоколу OAuth 2.0

OAUTH2\_CLIENT\_SECRET(security.oauth2.client.clientSecret) - секрет клиента для авторизации по протоколу OAuth 2.0

AUTH\_SERVER\_URL - URL сервера авторизации

**Для провайдера аутентификации FLEXTERA:**

USER (flowcontroller.user) - логин пользователя, под которым будет осуществляться выполнение пейджфло и серверных процессов

PASSWORD (flowcontroller.password) - пароль пользователя, под которым будет осуществляться выполнение пейджфло и серверных процессов

**Проверка правильности установки:**

После завершения установки и запуска Q.FloweR любым из предложенных способов можно выполнить простую проверку его работоспособности запросив информацию о его текущей версии.

Сделать это можно командой `curl http://${HOST}:${PORT}/flow/actuator` ,

где `{HOST}` - имя хоста или IP-адрес и `{PORT}` - порт

Пример вызова:

```
curl http://localhost:8080/flow/actuator
```

Ожидаемое состояние: должен вернуться статус 200 и ответ:

```
{ "_links": { "self": {  
  "href": "http://localhost:8080/flow/actuator",  
  "templated": false }, "health": {  
  
  "href": "http://localhost:8080/flow/actuator/health",  
  "templated": false  
  
  }, "info": {  
  
  "href": "http://localhost:8080/flow/actuator/info",  
  "templated": false  
  
  } } }
```