**Создание сервиса продуктов**

**api:**

* GET /products – отдает все продукты которые поддерживает сервис
* GET /products/{id}/rules – отдает правила применимости продукта к заемщику
* POST /products/{id}/rules – добавляет правило к продукту
* DELETE /products/{id}/rules/{id} – удаляет правило у продукта
* POST /products/apply – отдает продукты применимые к переданой информации о заемщике

**информация о заемщике состоит из:**

{

salary: 30000, #зарплата

claim: 20000, #желаемая сумма кредита

is\_debtor: true, #есть-ли текущая задолженость

}

**Пример взаимодействия с сервисом:**

Допустим у нас есть три продукта:

1. Кредит до 200000 под 6% на 3 года

выдается если зп заемщика больше 50000 и отсутствует текущая задолженность

2. Кредит на любую сумму под 15%

выдается если у заемщика отсутствует задолженность

3. Кредит до 1000000 под 12% на 5 лет

выдается если у заемщика ЗП больше 25000

Тогда для заемщика с информацией как в примере выше будет подходящим продукт номер 3

Если информация о заемщике будет как:

{

salary: 60000, #зарплата

claim: 300000, #желаемая сумма кредита

is\_debtor: false, #есть-ли текущая задолженность

}

то подходящими будут продукты 2 и 3

Пожелания к сервису:

* Сервис должен работать с данными в формате json
* Формат создания правил надо придумать самостоятельно
* Реализовать данный сервис с помощью Spring Boot
* Хранить продукты в БД на выбор (можно in-memory)
* В идеале структура БД должна создаваться с помощью Liquibase
* При старте сервиса должно быть предопределено несколько продуктов и правил к ним. В идеале тоже лучше сделать заполнение стандартными данными через Liquibase
* Должны быть модульные автотесты
* У сущностей должны быть кроме всего прочего такие служебные поля: Дата создания, Дата последнего изменения, Признак удаленной или активной сущности (Удаленные сущности помечаются флагом и продолжают храниться в бд, но в работе не используются)
* Конфигурационные параметры нужно хранить в файле конфигов (application.properties или application.yml)

Бонусное задание: Реализовать пополнение/изменение списка банковских продуктов с помощью очереди RabbitMQ. Тоже в формате json.