**Описание функциональных характеристик, информация для установки и эксплуатации программного обеспечения Модуль программного комплекса Goodfin «Продукт»**

Оглавление

[**Функциональные характеристики** 1](#_Toc57213606)

[**Установка** 1](#_Toc57213607)

[**Требования к рабочему окружению** 1](#_Toc57213608)

[**Эксплуатация системы** 2](#_Toc57213609)

[**Управление продуктами** 2](#_Toc57213610)

[**Пример yaml – файла продукта** 5](#_Toc57213611)

[**Управление тарифами** 17](#_Toc57213612)

[**Пример yaml – файла тарифа** 18](#_Toc57213613)

# **Функциональные характеристики**

**Программное обеспечение Модуль программного комплекса Goodfin «Продукт» (далее по тексту также программное обеспечение/ ПО)**

Назначение: Управление профилем отдельного продукта / отдельной услуги (далее по тексту также продукт), предоставляемой кредитной / финансовой или иной организацией

Область применения: для организаций, использующих для предоставления услуг программный комплекс Goodfin

Функциональные возможности: активация и настройка условий предоставления услуги; управление тарифами; расчет стоимости услуги в соответствии с заданными параметрами.

# **Установка**

ПО распространяется в виде интернет-сервиса, специальные действия по установке ПО на стороне пользователя не требуются

## **Требования к рабочему окружению**

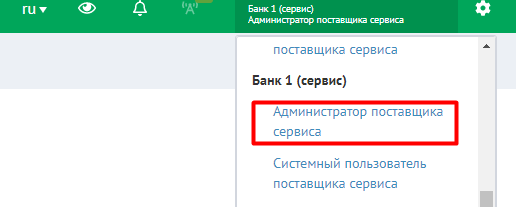
Требуемые программные продукты в составе клиентской части

|  |  |  |
| --- | --- | --- |
| **Наименование** | **Версия** | **Официальный сайт продукта** |
| КриптоПро CSP | Действующие сертифицированные | <http://www.cryptopro.ru/> |
| КриптоПро ЭЦП Browser plug-in | <http://www.cryptopro.ru/> |
| **Браузеры (любой из):** |  |  |
| Firefox | Три последние официальные стабильные версии (вышедшие не позднее года на момент проверки версионности) | <https://www.mozilla.org/> |
| Google Chrome | <https://www.google.ru/chrome/> |

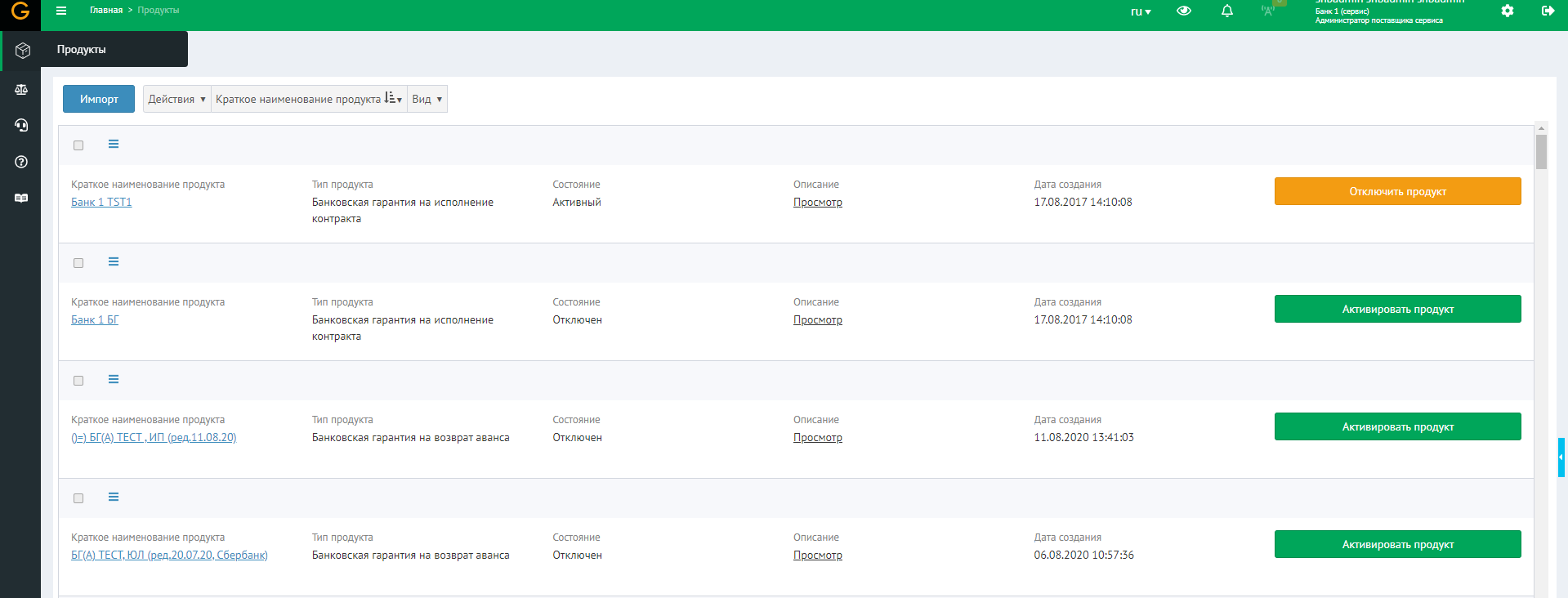
# **Эксплуатация системы**

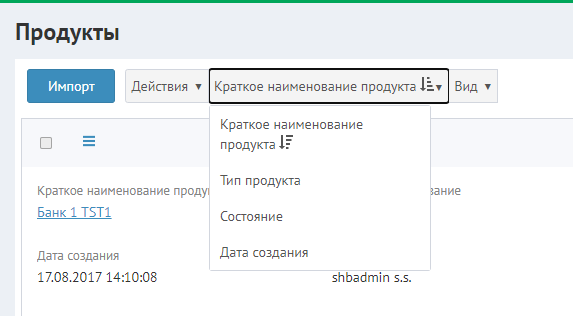
**Управление продуктами**

Управление продуктами происходит в личном кабинете поставщика сервиса, роль Администратор поставщика сервиса.

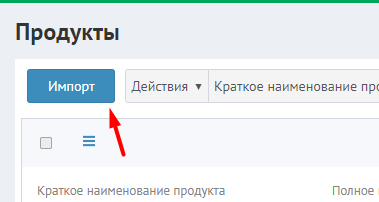


Вид страницы «Продукты» у Администратора поставщика сервиса:

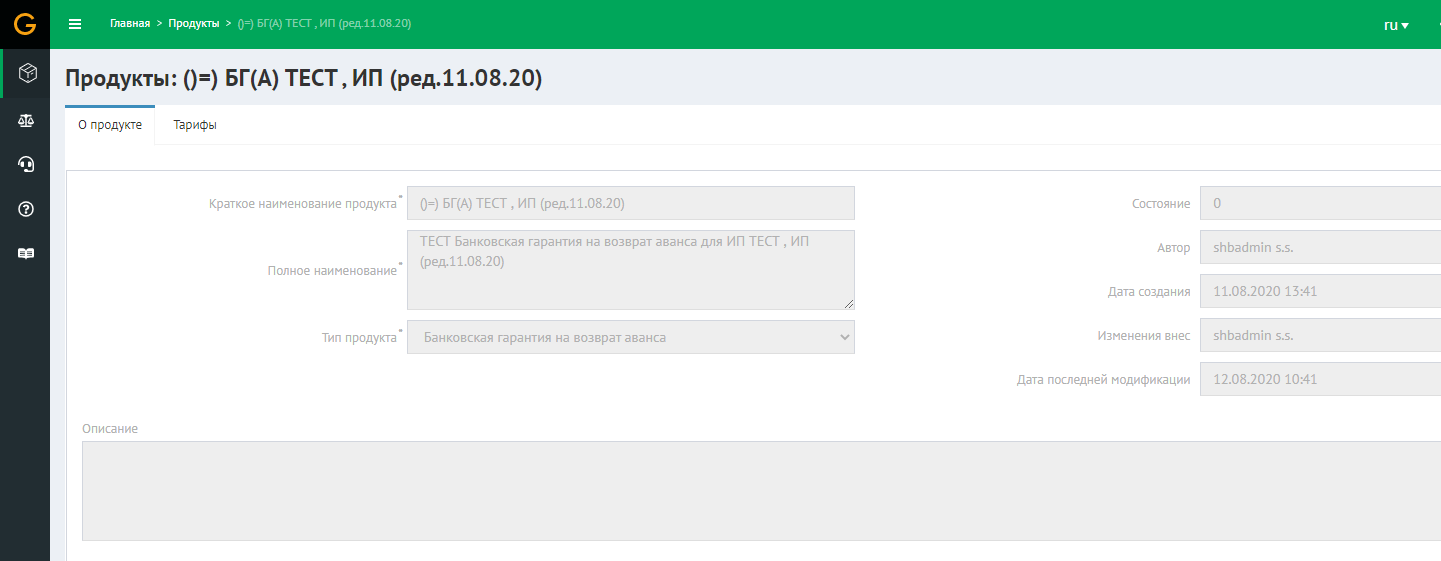


Продукты можно отсортировать по наименованию, типу, состоянию и дате создания: 

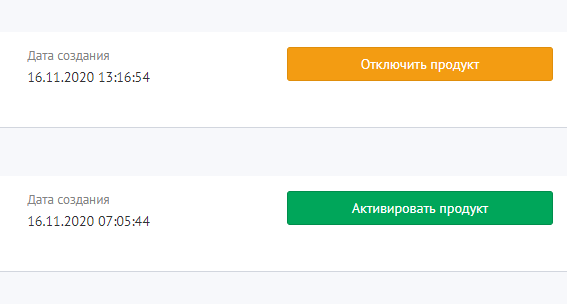
Создание нового продукта происходит путем импорта конфигурационного файла продукта (yaml). Пример yaml файла указан в соответствующем разделе. Продукты загружаются в систему через кнопку Импорт.



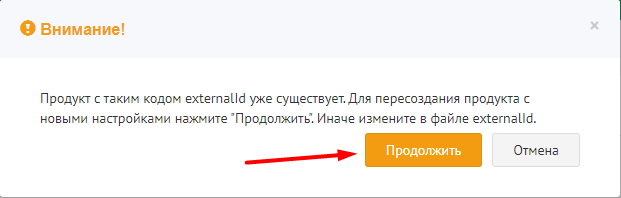
Посмотреть продукт, можно нажав на него, откроются данные для проверки, которые будут отображаться пользователю. Такие как наименование, описание, тип продукта.

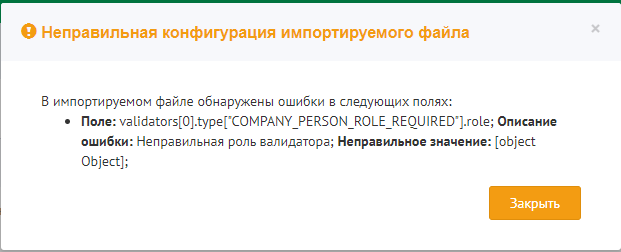


Для включения продукта, чтобы он был доступен для пользователей, требуется нажать кнопку «Активировать продукт». Для отключения – «Отключить продукт».



Для внесения правок в продукт требуется изменить yaml файл и перезагрузить его в систему, не добавляя новый продукт, а обновляя старый. Обновление происходит через кнопку импорт и согласие об обновлении. Обновлять можно только НЕ активные продукты, должна гореть кнопка «Активировать продукт».

  
Если в yaml файле допущена ошибка, будет выведено уведомление. Требуется исправить ошибку и загрузить файл вновь.



**В настройке yaml – файла продукта требуется настроить разделы финансовых показателей, правило применимости настройки, валидаторы (особенные поля обязательные для заполнения вне заявки для конкретного продукта), обязательные поля в заявке, обязательные документы к прикреплению и т.д.**

**Пример yaml – файла продукта**

**#настройки конкретного продукта**

**#элемент product обязателен и один**

**product:**

**#общие сведения о продукте**

**#name varchar(500) NOT NULL**

**name: "Банковская гарантия на исполнение для ЮЛ банка Открытие (ред.29.05.19)"**

**#shortname varchar(50) NOT NULL**

**shortName: "БГ на исполнение для ЮЛ (ред.29.05.19)"**

**#producttype varchar(50) NOT NULL**

**#мнемокод типа продукта, выбирается из справочника product\_types**

**#тип продукта должен совпадать с префиксам в перечене обязательных продуктовых полей**

**#например, "BG" - Банковская гарантия на исполнение контракта, "PARTICIPATION\_BG" - Банковская гарантия на участие, "TENDER\_LOAN" - Тендерный займ**

**productType: "BG"**

**#description varchar(1000) NULL**

**#Краткое описание паспорта продукта для вывода пользователям, чтобы пользователи могли понять ограничения по продукту**

**description: "[b]Законы:[/b] 44, 223, 615-ПП/185.[br][b]Макс. сумма БГ:[/b] 150 млн.р.[br][b]Maкс. срок действия БГ:[/b] 1095 дн.[br][b]Срок существования клиента:[/b] от 3 мес.-до 500 т.р.; от 6 мес.-до 1 м.р.; от 9 мес.-от 1 м.р. до 7 м.р.; от 12 мес.-от 7 м.р. до 50 м.р.; от 24 мес.-от 50 м.р. до 150 м.р.[br][b]Maкс. лимит на клиента:[/b] 150 м.р.[br][b]ОПФ клиента:[/b] АО, ООО, ИП.[br][b]Стоп регионы (по принципалу, бенефициару и месту выполнения контракта) :[/b] Респ. Адыгея; Карачаево-Черкесская Респ.; Кабардино-Балкарская Респ.; Респ. Северная Осетия – Алания; Респ. Ингушетия; Чеченская Респ.; Респ. Дагестан; Респ. Крым, г. Севастополь (допускается как место выполнения работ).[br][b]Предметы в стопе:[/b] автодорожное строительство «с нуля»; строительство жилых домов, высотных домов, коммерческой недвижимости, производственных объектов; приобретение жилья в рамках 214 ФЗ от 31.12.2004; строительство объектов для нужд МВД РФ, ФСБ РФ, Министерства обороны РФ, Администрации Президента РФ.[br]"**

**# workterm integer NULL**

**#Минимальное количество рабочих дней, в течение которых сервис готов предоставить продукт или услугу в законченном виде**

**workTerm: 1**

**#externalid uuid NOT NULL**

**#задается согласно требованиям к guid, например, использовать https://www.guidgenerator.com для генерации**

**#d1acb28c-9736-46b7-84c7-f739289773f3 - это код реальный продукта БГ для ЮЛ от банка Открытие**

**#если externalId совпадет с существующим продуктом в БД, то будет выполнена попытка перезаписи настроек продукта**

**externalId: "d1acb28c-9736-46b7-84c7-f739289773f3"**

**#правило применимости продукта, данные клиента и сделки должны удовлетворять этому правилу, чтобы получить предложение по продукту**

**#правило обязательно и одно на продукт**

**productApplicabilityRule:**

**#name varchar(150) NOT NULL**

**name: "БГ на исполнение ЮЛ банка Открытие. Доступно: по ФЗ 223, 44, 185/615 ПП; для коммер. и некомм. головных компаний"**

**#type varchar(50) NOT NULL, по умолчанию значение = "script"**

**type: "script"**

**#script character varying(4000) NOT NULL**

**#Скрипт правила в виде функции isAvailableForDealApplicationData(dealApplicationData), возвращающей TRUE или FALSE.**

**script: "function isAvailableForDealApplicationData(dealApplicationData) {**

**var orgTypes = ['COM', 'UNCOM'];**

**var companyClass = java.lang.Class.forName('com.keyintegrity.shb.company.query.dto.CompanyDto');**

**var clientCompany = dealApplicationData.fetchFields[companyClass][dealApplicationData.result.client.id];**

**if(clientCompany == null || clientCompany.orgType == null || clientCompany.legalType == null) {**

**return null;**

**}**

**var clientOrgTypeId = clientCompany.orgType.id;**

**var clientLegalTypeId = clientCompany.legalType.id;**

**return orgTypes.indexOf(clientOrgTypeId) !== -1 && clientLegalTypeId == 'ORGANIZATION';**

**}"**

**#-------------------------------------------------------------------------------------------------------------**

**#требования к карточке компании**

**#данный раздел должен существовать в настройках, если тип продукта предоставляется компаниям, иначе будет раздел individualFields**

**#при анализе yaml считать, если поле в списке required не найдено, то оно необязательно**

**companyFields:**

**required:**

**- "bankingDetails"**

**- "bankingDetails.bankName"**

**- "bankingDetails.bik"**

**- "bankingDetails.paymentAccount"**

**- "baseOkved"**

**- "factAddress"**

**- "factAddress.addressType"**

**- "factAddress.city"**

**- "factAddress.house"**

**- "factAddress.isOwned"**

**- "factAddress.postCode"**

**- "factAddress.region"**

**- "factAddress.street"**

**- "fullName"**

**- "inn"**

**- "kpp"**

**- "legalAddress"**

**- "legalAddress.addressType"**

**- "legalAddress.city"**

**- "legalAddress.house"**

**- "legalAddress.isOwned"**

**- "legalAddress.postCode"**

**- "legalAddress.region"**

**- "legalAddress.street"**

**- "legalFounders.address"**

**- "legalFounders.inn"**

**- "legalFounders.kpp"**

**- "legalFounders.name"**

**- "legalFounders.ogrn"**

**- "legalFounders.percent"**

**- "licenses.activity"**

**- "licenses.date"**

**- "licenses.listActivities"**

**- "licenses.number"**

**- "licenses.who"**

**- "ogrn"**

**- "okopf"**

**- "okpo"**

**- "orgType"**

**- "persons"**

**- "persons.birthday"**

**- "persons.birthdayPlace"**

**- "persons.docDate"**

**- "persons.docNumber"**

**- "persons.docSeries"**

**- "persons.docType"**

**- "persons.docWho"**

**- "persons.firstName"**

**- "persons.inn"**

**- "persons.lastName"**

**- "persons.mainOksm"**

**- "persons.middleName"**

**- "persons.regAddress"**

**- "regDate"**

**- "regOrg"**

**- "regPlace"**

**- "shortName"**

**- "taxSystem"**

**#-------------------------------------------------------------------------------------------------------------**

**#требования к продуктовым полям**

**#набор продуктовых полей будет отличаться в зависимости от типа продукта**

**#при анализе yaml считать, если поле в списке required не найдено, то оно необязательно**

**#при описании продуктовых полей префикс типа продукта оставляем, чтобы данные совпадали со справочными**

**productFields:**

**required:**

**- "BG:productDealState.actualContractPrice"**

**- "BG:productDealState.amount"**

**- "BG:productDealState.concludedContracts"**

**- "BG:productDealState.customer"**

**- "BG:productDealState.customer.inn"**

**- "BG:productDealState.customer.kpp"**

**- "BG:productDealState.customer.name"**

**- "BG:productDealState.customer.ogrn"**

**- "BG:productDealState.customer.region"**

**- "BG:productDealState.deadlineForGrant"**

**- "BG:productDealState.endDate"**

**- "BG:productDealState.hasLoss"**

**- "BG:productDealState.isIndisputableWriteOff"**

**- "BG:productDealState.isPrepayment"**

**- "BG:productDealState.isQuickDecision"**

**- "BG:productDealState.isWarrantyPeriod"**

**- "BG:productDealState.loanCommitments"**

**- "BG:productDealState.startDate"**

**- "BG:productDealState.tender"**

**- "BG:productDealState.tender.enforceAmount"**

**- "BG:productDealState.tender.federalLaw"**

**- "BG:productDealState.tender.initialContractPrice"**

**- "BG:productDealState.tender.isMultiCustomer"**

**- "BG:productDealState.tender.isMultiLot"**

**- "BG:productDealState.tender.lotName"**

**- "BG:productDealState.tender.lotNumber"**

**- "BG:productDealState.tender.name"**

**- "BG:productDealState.tender.placementType"**

**- "BG:productDealState.tender.publishDateTime"**

**- "BG:productDealState.tender.registryNumber"**

**- "BG:productDealState.tender.url"**

**#-------------------------------------------------------------------------------------------------------------**

**#требования к фин. показателям**

**finFields:**

**#описываются требования к каждой фин. форме, на текущий момент поддерживается два набора отчетности по две формы в каждом**

**#набор отчетности КНД 0710098 с формами ОКУД 0710001 и ОКУД 0710002**

**#набор отчетности КНД 0710099 с формами ОКУД 0710001 и ОКУД 0710002**

**forms:**

**- #form and indicators**

**#в formSet подставляем значение из табл. finance\_forms\_set.kndcode**

**#в form подставляем значение из табл. finance\_forms.okud**

**formSet: "0710098"**

**form: "0710001"**

**periodTypes:**

**- "currentReportingPeriod"**

**- "previousYear"**

**- "yearPrecedingPrevious"**

**indicators:**

**required:**

**- "noMaterial"**

**- "material"**

**- "1210"**

**- "finAssets"**

**- "1250"**

**- "capital"**

**- "1410"**

**- "longTermDuties"**

**- "1510"**

**- "1520"**

**- "shortTermDuties"**

**- "1600"**

**- "1700"**

**- #form and indicators**

**#в formSet подставляем значение из табл. finance\_forms\_set.kndcode**

**#в form подставляем значение из табл. finance\_forms.okud**

**formSet: "0710098"**

**form: "0710002"**

**periodTypes:**

**- "currentReportingPeriod"**

**- "previousYear"**

**- "yearPrecedingPrevious"**

**indicators:**

**required:**

**- "2110"**

**- "expenses"**

**- "otherIncomes"**

**- "2330"**

**- "2350"**

**- "2400"**

**- "incomeTaxes"**

**- #form and indicators**

**#в formSet подставляем значение из табл. finance\_forms\_set.kndcode**

**#в form подставляем значение из табл. finance\_forms.okud**

**formSet: "0710099"**

**form: "0710001"**

**periodTypes:**

**- "currentReportingPeriod"**

**- "previousYear"**

**- "yearPrecedingPrevious"**

**indicators:**

**required:**

**- "1100"**

**- "1110"**

**- "1120"**

**- "1130"**

**- "1140"**

**- "1150"**

**- "1160"**

**- "1170"**

**- "1180"**

**- "1190"**

**- "1200"**

**- "1210"**

**- "1220"**

**- "1230"**

**- "1240"**

**- "1250"**

**- "1260"**

**- "1300"**

**- "1310"**

**- "1320"**

**- "1340"**

**- "1350"**

**- "1360"**

**- "1370"**

**- "1400"**

**- "1410"**

**- "1420"**

**- "1430"**

**- "1450"**

**- "1500"**

**- "1510"**

**- "1520"**

**- "1530"**

**- "1540"**

**- "1550"**

**- "1600"**

**- "1700"**

**- #form and indicators**

**#в formSet подставляем значение из табл. finance\_forms\_set.kndcode**

**#в form подставляем значение из табл. finance\_forms.okud**

**formSet: "0710099"**

**form: "0710002"**

**periodTypes:**

**- "currentReportingPeriod"**

**- "previousYear"**

**- "yearPrecedingPrevious"**

**indicators:**

**required:**

**- "2100"**

**- "2110"**

**- "2120"**

**- "2200"**

**- "2210"**

**- "2220"**

**- "2300"**

**- "2310"**

**- "2320"**

**- "2330"**

**- "2340"**

**- "2350"**

**- "2400"**

**- "2410"**

**- "2421"**

**- "2430"**

**- "2450"**

**- "2460"**

**- "2500"**

**- "2510"**

**- "2520"**

**- "2900"**

**- "2910"**

**#-------------------------------------------------------------------------------------------------------------**

**#валидаторы по табл. products\_validators и validators**

**validators:**

**#каждый блок в массиве - это разобранный products\_validators.validatorjson text NOT NULL**

**#обязательно в каждом блоке массива элемент type с наименованием функции валидатора**

**#прочие элементы в блоке массива после type играют роль входных параметров для функции валидатора, параметров может не быть, а может быть несколько с разными наименованиями**

**- #validator**

**type: "COMPANY\_PERSON\_ROLE\_REQUIRED"**

**role: "BOOKER"**

**- #validator**

**type: "COMPANY\_PERSON\_ROLE\_REQUIRED"**

**role: "DIRECTOR"**

**- #validator**

**type: "COMPANY\_FOUNDER\_REQUIRED"**

**#-------------------------------------------------------------------------------------------------------------**

**#пакеты документов**

**#public.product\_docs\_packages**

**#в списке столько Блоков "- docsPackage:" сколько пакетов для продукта**

**docsPackages:**

**- #docsPackage 1**

**#name varchar(500) NOT NULL**

**#Указать полное наименование пакета документов, включая наименование продукта и сервиса для полноты понимания**

**name: "Пакет БГ(И), Банк Открытие: ВСЕ клиенты, 1 млн < сумма БГ на исполнение < = 50 млн"**

**#shortname varchar(50) NOT NULL**

**#Указать краткое наименование пакета документов, чтобы по нему понять условия применимости пакета документов**

**shortName: "ЮЛ, 1 млн < сумма < = 50 млн"**

**#description varchar(1000) NULL**

**description: NULL**

**#данные по табл. public.packages\_docs\_types**

**#списком перечисляются мнемокоды типов обязательных документов в пакете**

**docsTypes:**

**- "ANALYTICAL\_BALANCE\_LAST\_REPORTING\_YEAR"**

**- "LEASE\_AGREEMENT\_OWNERSHIP"**

**- "FINANCE\_REPORT\_LAST\_QUARTER"**

**#правило применимости пакета документов**

**applicabilityRule:**

**#name varchar(150) NOT NULL**

**#наименование правила применимости**

**name: "сумма БГ на исполнение больше 1 млн И меньше или равна 50 млн"**

**#type varchar(50) NOT NULL, по умолчанию значение = "script"**

**type: "script"**

**#script character varying(4000) NOT NULL**

**#Скрипт правила в виде функции isAvailableForDealApplicationData(dealApplicationData), возвращающей TRUE или FALSE.**

**script: " function isAvailableForDealApplicationData(dealApplicationData) { if (dealApplicationData.result.productDealState.amount == null) { return null; } return dealApplicationData.result.productDealState.amount > 1000000 && dealApplicationData.result.productDealState.amount <= 50000000; } "**

**- #docsPackage 2**

**#name varchar(500) NOT NULL**

**#Указать полное наименование пакета документов, включая наименование продукта и сервиса для полноты понимания**

**name: "Пакет БГ(И), Банк Открытие: ВСЕ клиенты, сумма БГ на исполнение > 50 млн"**

**#shortname varchar(50) NOT NULL**

**#Указать краткое наименование пакета документов, чтобы по нему понять условия применимости пакета документов**

**shortName: "ЮЛ, сумма > 50 млн"**

**#description varchar(1000) NULL**

**description: NULL**

**#данные по табл. public.packages\_docs\_types**

**#списком перечисляются мнемокоды типов обязательных документов в пакете**

**docsTypes:**

**- "FINANCE\_REPORT\_LAST\_QUARTER"**

**- "ANALYTICAL\_BALANCE\_LAST\_REPORTING\_YEAR"**

**- "ANNUAL\_ACCOUNTING\_FINANCIAL\_STATEMENTS"**

**- "LEASE\_AGREEMENT\_OWNERSHIP"**

**#правило применимости пакета документов**

**applicabilityRule:**

**#name varchar(150) NOT NULL**

**#наименование правила применимости**

**name: "сумма БГ на исполнение больше 50 млн"**

**#type varchar(50) NOT NULL, по умолчанию значение = "script"**

**type: "script"**

**#script character varying(4000) NOT NULL**

**#Скрипт правила в виде функции isAvailableForDealApplicationData(dealApplicationData), возвращающей TRUE или FALSE.**

**script: " function isAvailableForDealApplicationData(dealApplicationData) { if (dealApplicationData.result.productDealState.amount == null) { return null; } return dealApplicationData.result.productDealState.amount > 50000000; } "**

**- #docsPackage 3**

**#name varchar(500) NOT NULL**

**#Указать полное наименование пакета документов, включая наименование продукта и сервиса для полноты понимания**

**name: "Пакет БГ(И), Банк Открытие: ЮЛ на ОСНО, сумма БГ на исполнение > 10 млн"**

**#shortname varchar(50) NOT NULL**

**#Указать краткое наименование пакета документов, чтобы по нему понять условия применимости пакета документов**

**shortName: "ЮЛ на ОСНО, сумма > 10 млн"**

**#description varchar(1000) NULL**

**description: NULL**

**#данные по табл. public.packages\_docs\_types**

**#списком перечисляются мнемокоды типов обязательных документов в пакете**

**docsTypes:**

**- "TAX\_DECL\_VALUE\_ADDED\_TAX\_LAST\_TAX\_PERIOD"**

**- "TAX\_DECL\_ORGANIZATION\_INCOME\_TAX\_LAST\_TAX\_PERIOD"**

**#правило применимости пакета документов**

**applicabilityRule:**

**#name varchar(150) NOT NULL**

**#наименование правила применимости**

**name: "ЮЛ с налогообложением = ОСНО И сумма БГ на исполнение больше 10 млн"**

**#type varchar(50) NOT NULL, по умолчанию значение = "script"**

**type: "script"**

**#script character varying(4000) NOT NULL**

**#Скрипт правила в виде функции isAvailableForDealApplicationData(dealApplicationData), возвращающей TRUE или FALSE.**

**script: " function isAvailableForDealApplicationData(dealApplicationData) { var companyClass = java.lang.Class.forName('com.keyintegrity.shb.company.query.dto.CompanyDto'); var clientCompany = dealApplicationData.fetchFields[companyClass][dealApplicationData.result.client.id]; if (clientCompany == null) { return null; } var taxSystemClass = java.lang.Class.forName('com.keyintegrity.shb.company.query.dto.catalog.TaxSystemCatalogDto'); var taxSystemDto = dealApplicationData.fetchFields[taxSystemClass][clientCompany.taxSystem]; if (clientCompany.legalType == null || taxSystemDto == null || dealApplicationData.result.productDealState.amount == null) { return null; } var taxSystems = ['OSNO']; return taxSystems.indexOf(taxSystemDto.code) !== -1 && clientCompany.legalType.id == 'ORGANIZATION' && dealApplicationData.result.productDealState.amount > 10000000; } "**

**- #docsPackage 4**

**#name varchar(500) NOT NULL**

**#Указать полное наименование пакета документов, включая наименование продукта и сервиса для полноты понимания**

**name: "Пакет документов для ЮЛ (БГ на исполнение, БИН)"**

**#shortname varchar(50) NOT NULL**

**#Указать краткое наименование пакета документов, чтобы по нему понять условия применимости пакета документов**

**shortName: "Пакет документов для ЮЛ (БГ, БИН)"**

**#description varchar(1000) NULL**

**description: NULL**

**#данные по табл. public.packages\_docs\_types**

**#списком перечисляются мнемокоды типов обязательных документов в пакете**

**docsTypes:**

**- "CHARTER"**

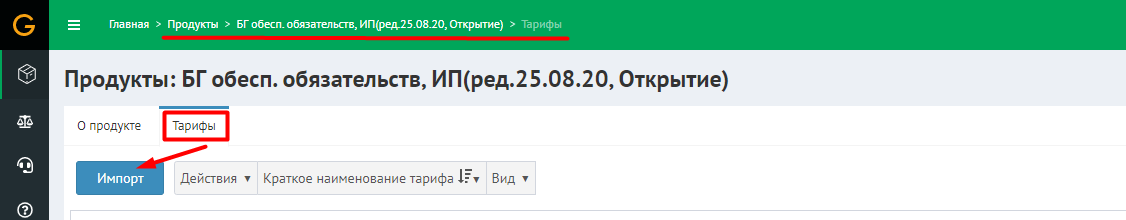
**- "HEAD\_APPOINTMENT\_PROTOCOL"**

**- "HEAD\_PASSPORT\_COPY"**

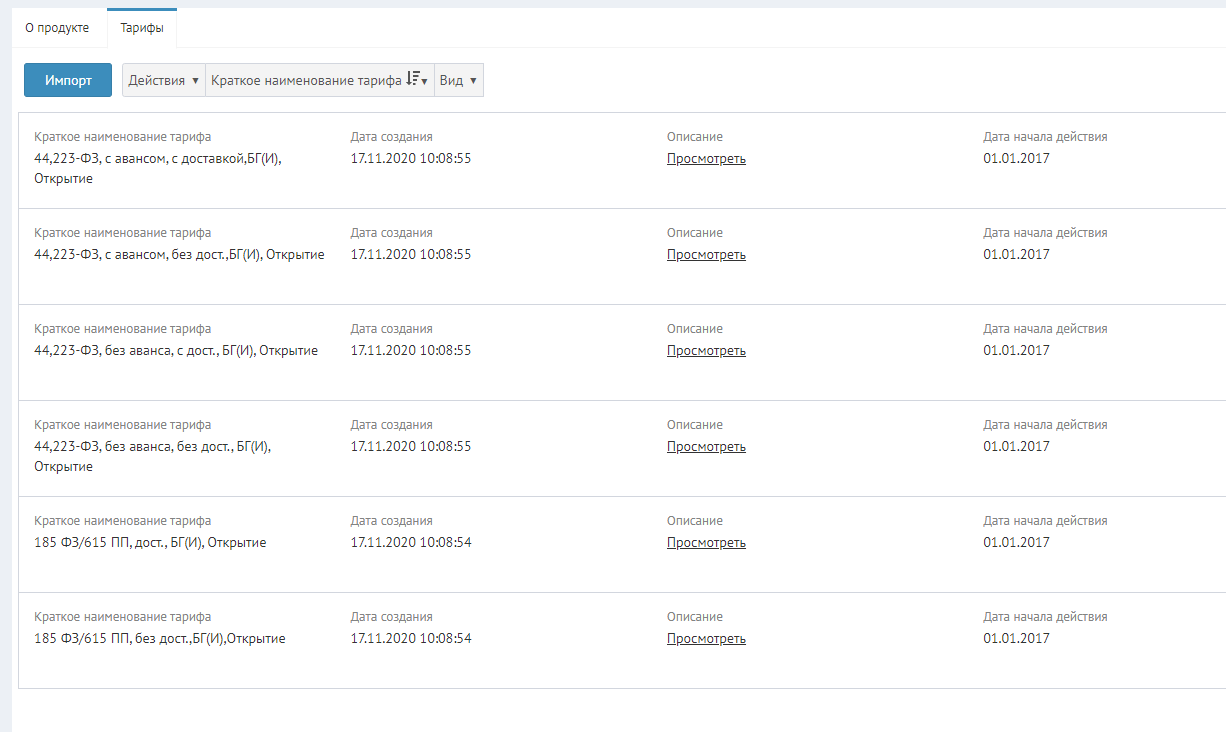
**#нет спец. правила применимости пакета документов - применимо безусловно для всего продукта**

## **Управление тарифами**

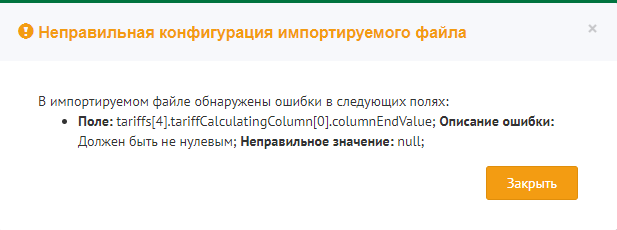
Для добавления к продукту тарификатора требуется нажать на продукт, перейти во вкладку Тарифы и нажать кнопку «Импорт» и выбрать yaml-файл тарификатора. Пользователь не сможет отправить заявку по продукту, пока в нём не будет тарифа. Поэтому «Активировать продукт» следует после успешного импорта тарифов.



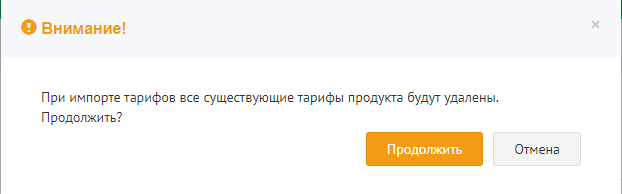
Тариф загружается в виде отдельных подтарифов, которые были указаны в yaml – файле. Пример файла тарифа приведён ниже.



Если в тарифе допущена ошибка, система не даст его загрузить. Требуется исправить ошибку и загрузить тариф ещё раз.



Чтобы обновить тарифы требуется загрузить новый файл тарифов в продукт, когда он НЕ активен. Система спросит уверены ли вы в загрузке и напомнит о том, что все предыдущие тарифы будут удалены.



**В настройках тарифа указываются ограничения по сроку выдачи продукта и его сумме, а также сама ставка тарифа. Тариф может быть фиксированным числом, рассчитываться процентом от суммы и срока выдаваемого продукта, либо только от суммы. В тарифе можно настроить относится он к продукту с авансом/ без аванса, с доставкой/ без доставки, конкретные ФЗ и т.д.**

**Пример yaml – файла тарифа**

#-------------------------------------------------------------------

#тарифы, связанные с продуктом

#тарифов может быть несколько в зависимости от ограничений, для каждого набора ограничений свой тариф, если тариф зависит от набора ограничений

tariffs:

- #tariff:

  #name varchar(250) NOT NULL

  name: "185 ФЗ/615 ПП, Тендербанк, БГ на исполнение"

  #shortName varchar(50) NOT NULL

  shortName: "185 ФЗ/615 ПП, Тендербанк, БГ на исполнение"

  #description varchar(500) NULL

  description: "185 ФЗ/615 ПП, Тендербанк, БГ на исполнение"

    productType: "BG"

  #startDate date NOT NULL, Дата начала действия тарифа

  startDate: "2019-06-02"

  #endDate date NULL, Дата окончания действия тарифа, если EndDate = Null и текущая дата > StartDate, то считаем, что тариф действует

  endDate: Null

  #вводится раздел настройки = settings, состав которого будет зависеть от типа тарификатор

  settings:

    #тариф может быть встроенный = INTERNAL, при котором в goodfin импортируются данные для расчета предварительного тарификатор

    #тариф может определяться по запросу к API сервиса в момент рассмотрения предложения

    #тариф может определяться сервисом только в момент рассмотрения заявки, следовательно на нашей стороне мы можем только указать "Определяется в сервисе"

    type: "INTERNAL"

    #fieldColumn - это блок для выражения FieldColumnID bigint NOT NULL - Ссылка на таблицу Product\_Tariff\_Fields для указания поля, по диапазонам значений которого будут рисоваться колонки тарификатора

    #Надо выбирать поле того типа продукта, для которого строится тарификатор

    #Поле также должно быть указано обязательным для продукта

    fieldColumn:

      #type varchar(50) NOT NULL - cсылка на Fields\_Collections\_Types, который указывает на один из справочников (источник поля FieldID)

      #указать COMPANY\_FIELDS или INDIVIDUAL\_FIELDS, или PRODUCT\_FIELDS, или  QUESTIONS\_LIST

      type: "PRODUCT\_FIELDS"

      #field varchar(100) NOT NULL - Ссылка на поле одной из таблиц Products\_Special\_Fields, Products\_Company\_Fields, Products\_Individual\_Fields, Products\_Questions\_List, которое указывает на поле, участвующее в расчете

      field: "BG:productDealState.amount"

    #fieldRow - это блок для выражения FieldRowID bigint NOT NULL - Ссылка на таблицу Product\_Tariff\_Fields для указания поля, по диапазонам значений которого будут рисоваться строки тарификатора.

    #Надо выбирать поле того типа продукта, для которого строится тарификатор

    #Поле также должно быть указано обязательным для продукта

    fieldRow:

      #type varchar(50) NOT NULL - cсылка на Fields\_Collections\_Types, который указывает на один из справочников (источник поля FieldID)

      #указать COMPANY\_FIELDS или INDIVIDUAL\_FIELDS, или PRODUCT\_FIELDS, или  QUESTIONS\_LIST

      type: "PRODUCT\_FIELDS"

      #field varchar(100) NOT NULL - Ссылка на поле одной из таблиц Products\_Special\_Fields, Products\_Company\_Fields, Products\_Individual\_Fields, Products\_Questions\_List, которое указывает на поле, участвующее в расчете

      field: "BG:productDealState.termDays"

    #calcField - это блок для выражения calcFieldID bigint NOT NULL - Ссылка на таблицу Product\_Tariff\_Fields для указания поля, которое является базовым для определения тарифа.

    #От значения данного поля будут браться проценты, суммироваться надбавки или вычитаться скидки и пр.

    #Надо выбирать поле того типа продукта, для которого строится тарификатор

    #Поле также должно быть указано обязательным для продукта

    calcField:

      #type varchar(50) NOT NULL - cсылка на Fields\_Collections\_Types, который указывает на один из справочников (источник поля FieldID)

      #указать COMPANY\_FIELDS или INDIVIDUAL\_FIELDS, или PRODUCT\_FIELDS, или  QUESTIONS\_LIST

      type: "PRODUCT\_FIELDS"

      #field varchar(100) NOT NULL - Ссылка на поле одной из таблиц Products\_Special\_Fields, Products\_Company\_Fields, Products\_Individual\_Fields, Products\_Questions\_List, которое указывает на поле, участвующее в расчете

      field: "BG:productDealState.amount"

    #строки тарифа

    tariffСalculatingRows:

    #rowStartValue - указание, с какой величины начинается расчет

    # rowStartValue < проверяемое значение <= rowEndValue,  другими словами диапазон указывается по правилу (rowStartValue,rowEndValue]

    -

      rowName: "row1\_365"

      rowStartValue: 1

      rowEndValue: 365

    -

      rowName: "row365\_732"

      rowStartValue: 365

      rowEndValue: 732

    -

      rowName: "row732\_1155"

      rowStartValue: 732

      rowEndValue: 1155

    #колонка тарифа

    tariffСalculatingColumns:

      #columnStartValue < период <= columnEndValue, другими словами диапазон указывается по правилу (columnStartValue,columnEndValue]

    -

      columnName: "col0\_10000"

      columnStartValue: 0

      columnEndValue: 10000

    -

      columnName: "col10000\_30000"

      columnStartValue: 10000

      columnEndValue: 30000

    -

      columnName: "col100000\_500000"

      columnStartValue: 100000

      columnEndValue: 500000

    -

      columnName: "col1500000\_5000000"

      columnStartValue: 1500000

      columnEndValue: 5000000

    -

      columnName: "col30000\_50000"

      columnStartValue: 30000

      columnEndValue: 50000

    -

      columnName: "col50000\_100000"

      columnStartValue: 50000

      columnEndValue: 100000

    -

      columnName: "col500000\_1500000"

      columnStartValue: 500000

      columnEndValue: 1500000

    tariffCells:

    - {rowName: "row1\_365", columnName: "col0\_10000", cellValue: 500.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row1\_365", columnName: "col10000\_30000", cellValue: 1000.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row1\_365", columnName: "col100000\_500000", cellValue: 3.40, cellValueType: "ANNUAL\_PERCENT", cellMinValue: 5000.00, cellMinValueType: "RUB"}

    - {rowName: "row1\_365", columnName: "col1500000\_5000000", cellValue: 3.50, cellValueType: "ANNUAL\_PERCENT", cellMinValue: 5000.00, cellMinValueType: "RUB"}

    - {rowName: "row1\_365", columnName: "col30000\_50000", cellValue: 2000.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row1\_365", columnName: "col50000\_100000", cellValue: 3000.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row1\_365", columnName: "col500000\_1500000", cellValue: 3.40, cellValueType: "ANNUAL\_PERCENT", cellMinValue: 5000.00, cellMinValueType: "RUB"}

    - {rowName: "row365\_732", columnName: "col0\_10000", cellValue: 2000.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row365\_732", columnName: "col10000\_30000", cellValue: 2000.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row365\_732", columnName: "col100000\_500000", cellValue: 3.40, cellValueType: "ANNUAL\_PERCENT", cellMinValue: 5000.00, cellMinValueType: "RUB"}

    - {rowName: "row365\_732", columnName: "col1500000\_5000000", cellValue: 3.50, cellValueType: "ANNUAL\_PERCENT", cellMinValue: 5000.00, cellMinValueType: "RUB"}

    - {rowName: "row365\_732", columnName: "col30000\_50000", cellValue: 3000.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row365\_732", columnName: "col50000\_100000", cellValue: 6000.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row365\_732", columnName: "col500000\_1500000", cellValue: 3.40, cellValueType: "ANNUAL\_PERCENT", cellMinValue: 5000.00, cellMinValueType: "RUB"}

    - {rowName: "row732\_1155", columnName: "col0\_10000", cellValue: 3000.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row732\_1155", columnName: "col10000\_30000", cellValue: 3000.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row732\_1155", columnName: "col100000\_500000", cellValue: 3.40, cellValueType: "ANNUAL\_PERCENT", cellMinValue: 5000.00, cellMinValueType: "RUB"}

    - {rowName: "row732\_1155", columnName: "col1500000\_5000000", cellValue: 3.50, cellValueType: "ANNUAL\_PERCENT", cellMinValue: 5000.00, cellMinValueType: "RUB"}

    - {rowName: "row732\_1155", columnName: "col30000\_50000", cellValue: 4500.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row732\_1155", columnName: "col50000\_100000", cellValue: 9000.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row732\_1155", columnName: "col500000\_1500000", cellValue: 3.40, cellValueType: "ANNUAL\_PERCENT", cellMinValue: 5000.00, cellMinValueType: "RUB"}

    #Таблица, хранящая список полей, которые указывают контекст и ограничения применимости тарифа.

    #Например, если тариф рассчитывается по сумме обеспечения и сроку для тендеров по ФЗ-44 или ФЗ-223, то указание на поле "Закон, по которому формируется тендер" будет как раз в этой таблице.

    #А поля срок и сумма будут образовывать колонки и строки тарифной сетки и будут описаны в таблице Tariffs.

    tariffConstraints:

    #блок повторяется столько раз, сколько ограничений в применимости тарифа

    #type и field - это блок для выражения FieldColumnID bigint NOT NULL - Ссылка на таблицу Product\_Tariff\_Fields для указания поля, по диапазонам значений которого будут рисоваться колонки тарификатора

    #Надо выбирать поле того типа продукта, для которого строится тарификатор

    #Поле также должно быть указано обязательным для продукта

    #FieldID bigint NOT NULL,    -- Ссылка на поле, значения которого определяют контекст применимости тарифа.

    #valueTypeID varchar(50) NOT NULL,    -- Ссылка на справочник Tariff\_Field\_Value\_Types. Это может быть простое перечисление значений, может быть диапазон значений.

    #варианты значений ENUM или RANGE

    #для типа valueType = ENUM values оформляется списком допустимых значений

    #для типа valueType = RANGE values оформляется обязательным списком из двух значений: начального и конечного

    #примеры перечисления значений values

    -

      type: "PRODUCT\_FIELDS"

      field: "BG:productDealState.tender.federalLaw"

      valueType: "ENUM"

      values:

      - "185FL615PP"

- #tariff:

  #name varchar(250) NOT NULL

  name: "223-ФЗ, 44-ФЗ - без аванса, Тендербанк, БГ на исполнение"

  #shortName varchar(50) NOT NULL

  shortName: "223-ФЗ, 44-ФЗ - без аванса, Тендербанк, БГ на исп"

  #description varchar(500) NULL

  description: "223-ФЗ, 44-ФЗ - без аванса, Тендербанк, БГ на исполнение"

  productType: "BG"

  #startDate date NOT NULL, Дата начала действия тарифа

  startDate: "2019-06-02"

  #endDate date NULL, Дата окончания действия тарифа, если EndDate = Null и текущая дата > StartDate, то считаем, что тариф действует

  endDate: Null

  #вводится раздел настройки = settings, состав которого будет зависеть от типа тарификатор

  settings:

    #тариф может быть встроенный = INTERNAL, при котором в goodfin импортируются данные для расчета предварительного тарификатор

    #тариф может определяться по запросу к API сервиса в момент рассмотрения предложения

    #тариф может определяться сервисом только в момент рассмотрения заявки, следовательно на нашей стороне мы можем только указать "Определяется в сервисе"

    type: "INTERNAL"

    #fieldColumn - это блок для выражения FieldColumnID bigint NOT NULL - Ссылка на таблицу Product\_Tariff\_Fields для указания поля, по диапазонам значений которого будут рисоваться колонки тарификатора

    #Надо выбирать поле того типа продукта, для которого строится тарификатор

    #Поле также должно быть указано обязательным для продукта

    fieldColumn:

      #type varchar(50) NOT NULL - cсылка на Fields\_Collections\_Types, который указывает на один из справочников (источник поля FieldID)

      #указать COMPANY\_FIELDS или INDIVIDUAL\_FIELDS, или PRODUCT\_FIELDS, или  QUESTIONS\_LIST

      type: "PRODUCT\_FIELDS"

      #field varchar(100) NOT NULL - Ссылка на поле одной из таблиц Products\_Special\_Fields, Products\_Company\_Fields, Products\_Individual\_Fields, Products\_Questions\_List, которое указывает на поле, участвующее в расчете

      field: "BG:productDealState.amount"

    #fieldRow - это блок для выражения FieldRowID bigint NOT NULL - Ссылка на таблицу Product\_Tariff\_Fields для указания поля, по диапазонам значений которого будут рисоваться строки тарификатора.

    #Надо выбирать поле того типа продукта, для которого строится тарификатор

    #Поле также должно быть указано обязательным для продукта

    fieldRow:

      #type varchar(50) NOT NULL - cсылка на Fields\_Collections\_Types, который указывает на один из справочников (источник поля FieldID)

      #указать COMPANY\_FIELDS или INDIVIDUAL\_FIELDS, или PRODUCT\_FIELDS, или  QUESTIONS\_LIST

      type: "PRODUCT\_FIELDS"

      #field varchar(100) NOT NULL - Ссылка на поле одной из таблиц Products\_Special\_Fields, Products\_Company\_Fields, Products\_Individual\_Fields, Products\_Questions\_List, которое указывает на поле, участвующее в расчете

      field: "BG:productDealState.termDays"

    #calcField - это блок для выражения calcFieldID bigint NOT NULL - Ссылка на таблицу Product\_Tariff\_Fields для указания поля, которое является базовым для определения тарифа.

    #От значения данного поля будут браться проценты, суммироваться надбавки или вычитаться скидки и пр.

    #Надо выбирать поле того типа продукта, для которого строится тарификатор

    #Поле также должно быть указано обязательным для продукта

    calcField:

      #type varchar(50) NOT NULL - cсылка на Fields\_Collections\_Types, который указывает на один из справочников (источник поля FieldID)

      #указать COMPANY\_FIELDS или INDIVIDUAL\_FIELDS, или PRODUCT\_FIELDS, или  QUESTIONS\_LIST

      type: "PRODUCT\_FIELDS"

      #field varchar(100) NOT NULL - Ссылка на поле одной из таблиц Products\_Special\_Fields, Products\_Company\_Fields, Products\_Individual\_Fields, Products\_Questions\_List, которое указывает на поле, участвующее в расчете

      field: "BG:productDealState.amount"

    #строки тарифа

    tariffСalculatingRows:

    #rowStartValue - указание, с какой величины начинается расчет

    # rowStartValue < проверяемое значение <= rowEndValue,  другими словами диапазон указывается по правилу (rowStartValue,rowEndValue]

    -

      rowName: "row1\_365"

      rowStartValue: 1

      rowEndValue: 365

    -

      rowName: "row365\_732"

      rowStartValue: 365

      rowEndValue: 732

    -

      rowName: "row732\_1155"

      rowStartValue: 732

      rowEndValue: 1155

    #колонка тарифа

    tariffСalculatingColumns:

      #columnStartValue < период <= columnEndValue, другими словами диапазон указывается по правилу (columnStartValue,columnEndValue]

    -

      columnName: "col0\_10000"

      columnStartValue: 0

      columnEndValue: 10000

    -

      columnName: "col10000\_30000"

      columnStartValue: 10000

      columnEndValue: 30000

    -

      columnName: "col100000\_500000"

      columnStartValue: 100000

      columnEndValue: 500000

    -

      columnName: "col1500000\_5000000"

      columnStartValue: 1500000

      columnEndValue: 5000000

    -

      columnName: "col30000\_50000"

      columnStartValue: 30000

      columnEndValue: 50000

    -

      columnName: "col50000\_100000"

      columnStartValue: 50000

      columnEndValue: 100000

    -

      columnName: "col500000\_1500000"

      columnStartValue: 500000

      columnEndValue: 1500000

    tariffCells:

    - {rowName: "row1\_365", columnName: "col0\_10000", cellValue: 500.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row1\_365", columnName: "col10000\_30000", cellValue: 1000.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row1\_365", columnName: "col100000\_500000", cellValue: 3.20, cellValueType: "ANNUAL\_PERCENT", cellMinValue: 5000.00, cellMinValueType: "RUB"}

    - {rowName: "row1\_365", columnName: "col1500000\_5000000", cellValue: 3.30, cellValueType: "ANNUAL\_PERCENT", cellMinValue: 12000.00, cellMinValueType: "RUB"}

    - {rowName: "row1\_365", columnName: "col30000\_50000", cellValue: 2000.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row1\_365", columnName: "col50000\_100000", cellValue: 3000.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row1\_365", columnName: "col500000\_1500000", cellValue: 3.00, cellValueType: "ANNUAL\_PERCENT", cellMinValue: 8000.00, cellMinValueType: "RUB"}

    - {rowName: "row365\_732", columnName: "col0\_10000", cellValue: 2000.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row365\_732", columnName: "col10000\_30000", cellValue: 2000.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row365\_732", columnName: "col100000\_500000", cellValue: 3.20, cellValueType: "ANNUAL\_PERCENT", cellMinValue: 5000.00, cellMinValueType: "RUB"}

    - {rowName: "row365\_732", columnName: "col1500000\_5000000", cellValue: 3.30, cellValueType: "ANNUAL\_PERCENT", cellMinValue: 12000.00, cellMinValueType: "RUB"}

    - {rowName: "row365\_732", columnName: "col30000\_50000", cellValue: 3000.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row365\_732", columnName: "col50000\_100000", cellValue: 6000.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row365\_732", columnName: "col500000\_1500000", cellValue: 3.00, cellValueType: "ANNUAL\_PERCENT", cellMinValue: 8000.00, cellMinValueType: "RUB"}

    - {rowName: "row732\_1155", columnName: "col0\_10000", cellValue: 3000.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row732\_1155", columnName: "col10000\_30000", cellValue: 3000.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row732\_1155", columnName: "col100000\_500000", cellValue: 3.20, cellValueType: "ANNUAL\_PERCENT", cellMinValue: 5000.00, cellMinValueType: "RUB"}

    - {rowName: "row732\_1155", columnName: "col1500000\_5000000", cellValue: 3.30, cellValueType: "ANNUAL\_PERCENT", cellMinValue: 12000.00, cellMinValueType: "RUB"}

    - {rowName: "row732\_1155", columnName: "col30000\_50000", cellValue: 4500.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row732\_1155", columnName: "col50000\_100000", cellValue: 9000.00, cellValueType: "RUB", cellMinValue: Null, cellMinValueType: Null}

    - {rowName: "row732\_1155", columnName: "col500000\_1500000", cellValue: 3.00, cellValueType: "ANNUAL\_PERCENT", cellMinValue: 8000.00, cellMinValueType: "RUB"}

    #Таблица, хранящая список полей, которые указывают контекст и ограничения применимости тарифа.

    #Например, если тариф рассчитывается по сумме обеспечения и сроку для тендеров по ФЗ-44 или ФЗ-223, то указание на поле "Закон, по которому формируется тендер" будет как раз в этой таблице.

    #А поля срок и сумма будут образовывать колонки и строки тарифной сетки и будут описаны в таблице Tariffs.

    tariffConstraints:

    #блок повторяется столько раз, сколько ограничений в применимости тарифа

    #type и field - это блок для выражения FieldColumnID bigint NOT NULL - Ссылка на таблицу Product\_Tariff\_Fields для указания поля, по диапазонам значений которого будут рисоваться колонки тарификатора

    #Надо выбирать поле того типа продукта, для которого строится тарификатор

    #Поле также должно быть указано обязательным для продукта

    #FieldID bigint NOT NULL,    -- Ссылка на поле, значения которого определяют контекст применимости тарифа.

    #valueTypeID varchar(50) NOT NULL,    -- Ссылка на справочник Tariff\_Field\_Value\_Types. Это может быть простое перечисление значений, может быть диапазон значений.

    #варианты значений ENUM или RANGE

    #для типа valueType = ENUM values оформляется списком допустимых значений

    #для типа valueType = RANGE values оформляется обязательным списком из двух значений: начального и конечного

    #примеры перечисления значений values

    -

      type: "PRODUCT\_FIELDS"

      field: "BG:productDealState.isPrepayment"

      valueType: "ENUM"

      values:

      - "false"

    -

      type: "PRODUCT\_FIELDS"

      field: "BG:productDealState.tender.federalLaw"

      valueType: "ENUM"

      values:

      - "44FL"

      - "223FL"