**Тинькофф автокредитный брокер. API для взаимодействия с партнерами**

Описание сервиса и методов:   
тест - <https://qa.forma.tcsbank.ru/api/autobroker/v1/cobroker/swagger#/>   
прод - <https://forma.tinkoff.ru/api/autobroker/v1/cobroker/swagger#/>

Сервис состоит из 4 методов:

1. **POST | /api/autobroker/v1/cobroker/app/create**   
   Данным методом осуществляется создание заявки – передаем id точки, в ответ получаем id заявки и token (потребуется для работы с последующими методами).

|  |  |
| --- | --- |
| **Параметр** | **Описание** |
| pointId\* | id точки, которую предоставляется Тинькофф |
| contact |  |
| fio | ФИО |
| mobilePhone | Номер мобильного телефона |
| email | Почта |
| referrer | Источник запроса |

запрос:

**{**

**"pointId": "513c6bef-5f27-4288-92a0-35924dd5b320",**

**"referrer": "ROLF",**

**"branding": "China",**

**"car": {**

**"brand": "Haval",**

**"model": "Jolion",**

**"year": 2022**

**},**

**"contact": {**

**"fio": "Коброкеров Евгений Николаевич",**

**"mobilePhone": "99791234567",**

**"email": "auto@mail.ru"**

**},**

**"creditInfo": {**

**"carCost": 30000000,**

**"creditType": "new",**

**"initialFee": 100000,**

**"creditTerm": 5**

**}**

**}**

|  |  |
| --- | --- |
| **Параметр** | **Описание** |
| values |  |
| statusCode | Код ответа |
| id | id созданной заявки |
| token | токен для авторизации |
| link | Ссылка |

ответ:

**{**

**"values": null,**

**"statusCode": "Ok",**

**"id": "b55e07cc-6094-49e7-a05b-de442fbfcc0f",**

**"token": "f6bfb5a2-00e6-41d2-ab2b-5de1433c82c4",**

**"link": "https://qa.forma.tcsbank.ru/cobroker/b55e07cc-6094-49e7-a05b-de442fbfcc0f?mobilePhone=99791234567&token=f6bfb5a2-00e6-41d2-ab2b-5de1433c82c4&fio=%D0%9A%D0%BE%D0%B1%D1%80%D0%BE%D0%BA%D0%B5%D1%80%D0%BE%D0%B2%20%D0%95%D0%B2%D0%B3%D0%B5%D0%BD%D0%B8%D0%B9%20%D0%9D%D0%B8%D0%BA%D0%BE%D0%BB%D0%B0%D0%B5%D0%B2%D0%B8%D1%87&email=auto%40maul.ru&branding=China"**

**}**

1. **POST|/api/autobroker/v1/cobroker/app/next**

Данным методом осуществляется обогащение заявки данными (схема запроса см. в swagger). При вызове метода необходима авторизация bearer (в заголовках передаем токен, полученный в методе /create).

|  |  |  |
| --- | --- | --- |
| **Параметр** | **Описание** | **Варианты** |
| id |  |  |
| *values\** |  |  |
| ***contact****\** | Контактная информация |  |
| lastName\* | Фамилия |  |
| firstName\* | Имя |  |
| middleName\* | Отчество |  |
| birthdate\* | Дата рождения |  |
| birthPlace\* | Место рождения |  |
| gender | Пол | male/female |
| email | Электронная почта |  |
| personalIncome\* | Ежемесячный доход |  |
| familyIncome\* | Ежемесячный общий семейный доход |  |
| education\* | Уровень образования | low/incomplete\_high/high/second/ academic\_degree |
| numberOfChildren\* | Количество детей | zero/one/two/over\_two |
| maritalStatus\* | Семейный статус | single/separated/partner/married/widowed |
| mobile\* | Телефон |  |
| number\* | Мобильный телефон |  |
| ***jobInfo****\** | Работа по найму (если безработный) |  |
| type\* | Тип занятости | unemployed |
| cause\* | Причина отсутствия работы | Если type: unemployed;  pension\_age/pension\_disability/husband\_wife\_contains/look\_for\_job/other |
| comment | Комментарий, если cause: other | свободный ввод |
| ***jobInfo****\** | Работа по найму (если официальная работа по найму) |  |
| *phones\** |  |  |
| *phone\** |  |  |
| number\* | Телефон организации |  |
| type\* | Тип занятости | official |
| companyName\* | Название организации места работы |  |
| inn | ИНН организации |  |
| numberOfEmployees\* | Количество сотрудников в организации | less\_than\_20/up\_to\_50/up\_to\_250/over\_250 |
| *address\** |  |  |
| kladr | Кладр организации |  |
| kladr\_id | Кладр id организации |  |
| postalCode | Почтовый индекс организации |  |
| country | Страна адреса организации |  |
| state | Регион адреса организации |  |
| city | Город адреса организации |  |
| settlement | Поселение адреса организации |  |
| street | Улица адреса организации |  |
| house | Номер дома адреса организации |  |
| building | Номер строения адреса организации |  |
| construction | Номер корпуса адреса организации |  |
| apartment | Номер офиса адреса организации |  |
| detachedHouse | Отдельно стоящий офис | true/false |
| fiasCode | ФИАС код адреса организации |  |
| fiasId | ФИАС id адреса организации |  |
| fiasLevel | ФИАС уровень адреса организации |  |
| correspondenceFlag |  | true/false |
| workExperience\* | Опыт работы | up\_to\_6\_months/up\_to\_3\_years/up\_to\_5\_years/up\_to\_7\_years/over\_7\_years |
| position\* | Должность | Из справочника должностей (свободный ввод) |
| positionType\* | Тип должности | support/working/specialist/manager\_department/manager\_organization |
| ***businessInfo****\** | Свой бизнес (если официальный) |  |
| type\* | Тип занятости | official |
| companyName\* | Название компании |  |
| workPhone\* | Рабочий номер телефона |  |
| ***businessInfo****\** | Свой бизнес (если неофициальный) |  |
| type\* | Тип занятости | unofficial |
| workDescription\* | Вид деятельности | Свободный ввод |
| workPhone\* | Рабочий номер телефона |  |
| ***сontacts\**** | Информация о дополнительных контактах |  |
| additional\* |  |  |
| type\* | Тип доп контакта | friend/family/spouse/colleague/add\_mobile/home/ registration |
| firstName\* | Имя доп контакта |  |
| middleName\* | Отчество доп контакта |  |
| lastName\* | Фамилия доп контакта |  |
| phone\* |  |  |
| number\* | Мобильный телефон доп контакта |  |
| **driverType**\* | Тип водительских прав | сlient/none/anotherDriver |
| ***driverLicense*** | Информация о водительских правах |  |
| num\* | Номер водительских прав |  |
| date\* | Дата выдачи водительских прав |  |
| drivingStartYear\* | Год начала стажа |  |
| ***creditInfo\**** | Информация о запрошенных условиях кредита |  |
| carCost\* | Запрошенная стоимость автомобиля |  |
| creditType\* | Тип авто (новый/бу) | new/used |
| initialFee\* | Запрошенный первоначальный взнос |  |
| creditTerm\* | Запрошенный срок кредита |  |
| ***addr\_reg\**** | Информация о месте регистрации |  |
| kladr | Кладр места регистрации |  |
| kladr\_id | Кладр id места регистрации |  |
| postalCode | Почтовый индекс места регистрации |  |
| country | Страна места регистрации |  |
| state | Регион места регистрации |  |
| city | Город места регистрации |  |
| settlement | Поселение/село места регистрации |  |
| street | Улица места регистрации |  |
| house | Номер дома места регистрации |  |
| building | Номер строения места регистрации |  |
| construction | Номер корпуса места регистрации |  |
| apartment | Номер квартиры места регистрации |  |
| fiasCode | ФИАС код места регистрации |  |
| fiasId | ФИАС id места регистрации |  |
| fiasLevel |  |  |
| ***addr\_home\**** | Информация о месте проживания |  |
| kladr | Кладр места проживания |  |
| kladr\_id | Кладр id места проживания |  |
| postalCode | Почтовый индекс места проживания |  |
| country | Страна места проживания |  |
| state | Регион места проживания |  |
| city | Город места проживания |  |
| settlement | Поселение/село места проживания |  |
| street | Улица места проживания |  |
| house | Номер дома места проживания |  |
| building | Номер строения места проживания |  |
| construction | Номер корпуса места проживания |  |
| apartment | Номер квартиры места проживания |  |
| fiasCode | ФИАС код места проживания |  |
| fiasId | ФИАС id места проживания |  |
| fiasLevel |  |  |
| ***passport\**** | Информация о паспортных данных |  |
| num\* | Номер паспорта |  |
| division\* | Код подразделения, выдавшего паспорт |  |
| organization\* | Кем выдан паспорт |  |
| date\* | Дата выдачи паспорта |  |
| ***photos*** |  |  |
| *clientPhoto* | Фото клиента |  |
| fileId | id файла фото клиента |  |
| fileName | Имя файла фото клиента |  |
| contentType | Mime-type файла |  |
| *passportMain* | Фото страницы основного разворота паспорта |  |
| fileId | id файла страницы основного разворота паспорта |  |
| fileName | Имя файла страницы основного разворота паспорта |  |
| contentType | Mime-type файла |  |
| *passportRegistration* | Фото страницы регистрации паспорта |  |
| fileId | id файла страницы регистрации паспорта |  |
| fileName | Имя файла страницы регистрации паспорта |  |
| contentType | Mime-type файла |  |
| *passportPrevious* | Фото страницы паспорта — сведения о ранее выданных паспортах |  |
| fileId | id файла страницы паспорта — сведения о ранее выданных паспортах |  |
| fileName | Имя файла страницы паспорта — сведения о ранее выданных паспортах |  |
| contentType | Mime-type файла |  |
| *brokerAgreement* | Фото согласия на обработку персональных данных |  |
| fileId | id файла согласия на обработку персональных данных |  |
| fileName | Имя файла согласия на обработку персональных данных |  |
| contentType | Mime-type файла |  |
| *driverLicense* | Фото водительских прав |  |
| fileId | id файла водительских прав |  |
| fileName | Имя файла водительских прав |  |
| contentType | Mime-type файла |  |
| *anotherDriverLicense* | Фото водительских прав другого водителя |  |
| fileId | id файла водительских прав другого водителя |  |
| fileName | Имя файла водительских прав другого водителя |  |
| contentType | Mime-type файла |  |
| *ptsPhoto* | Фото ПТС/ЭПТС |  |
| fileId | id файла ПТС/ЭПТС |  |
| fileName | Имя файла ПТС/ЭПТС |  |
| contentType | Mime-type файла |  |
| *creditTerms* | Фото ИУ |  |
| fileId | id файла ИУ (индивидуальных услвоий) |  |
| fileName | Имя файла ИУ (индивидуальных услвоий) |  |
| contentType | Mime-type файла |  |
| *contractOfSale* | Фото ДКП |  |
| fileId | id файла ДКП |  |
| fileName | Имя файла ДКП |  |
| contentType | Mime-type файла |  |
| *declaration* | Фото Оферты |  |
| fileId | id файла Оферты |  |
| fileName | Имя файла Оферты |  |
| contentType | Mime-type файла |  |
| *contractNS* | Фото полиса страхования Тинькофф НС3 |  |
| fileId | id файла Договора полиса страхования жизни Тинькофф НС3 |  |
| fileName | Имя файла Договора полиса страхования жизни Тинькофф НС3 |  |
| contentType | Mime-type файла |  |
| *additionalDocuments* | Фото Других документов |  |
| fileId | id файла Других документов |  |
| fileName | Имя файла Других документов |  |
| contentType | Mime-type файла |  |
| *car* | ТС |  |
| model | модель |  |
| name | Наименование модели |  |
| id | id модели |  |
| brand | марка |  |
| name | Наименование марки |  |
| id | id марки |  |
| generation | поколение |  |
| name | Наименование поколение |  |
| id | id поколения |  |
| year | год ТС |  |

**Запрос:**

**{**

**"id": "285ed121-cbe9-40c0-9acb-e1c6d300c854",**

**"values": {**

**"contact": {**

**"lastName": "Коброкеров",**

**"firstName": "Евгений",**

**"middleName": "Николаевич",**

**"birthDate": "1993-11-23",**

**"birthPlace": "Москва",**

**"personalIncome": 90000,**

**"familyIncome": 150000,**

**"education": "high",**

**"numberOfChildren": "zero",**

**"maritalStatus": "single",**

**"mobile": {**

**"number": "9585835244"**

**}**

**},**

**"jobInfo": {**

**"phones": [**

**{**

**"phone": {**

**"number": "9708780878"**

**}**

**}**

**],**

**"type": "official",**

**"companyName": "ФГБОУ ВО МГМСУ ИМ. А.И. ЕВДОКИМОВА МИНЗДРАВА РОССИИ",**

**"inn": "7707082145",**

**"numberOfEmployees": "over\_250",**

**"address": {**

**"postalCode": "117463",**

**"state": "г Москва",**

**"street": "пр-кт Новоясеневский",**

**"house": "40",**

**"building": "3",**

**"construction": "2",**

**"apartment": "6548",**

**"detachedHouse": false**

**},**

**"workExperience": "up\_to\_7\_years",**

**"position": "Ведущий специалист",**

**"positionType": "specialist"**

**},**

**"contacts": {**

**"additional": []**

**},**

**"driverType": "client",**

**"driverLicense": {**

**"num": "1546165165",**

**"date": "2018-03-02",**

**"drivingStartYear": 2018**

**},**

**"creditInfo": {**

**"carCost": 30000000,**

**"creditType": "new",**

**"initialFee": 100000,**

**"creditTerm": 5**

**},**

**"car": {**

**"brand": {"id": 177, "name": "Renault"},**

**"model": {"id": 1499, "name": "Logan"},**

**"generation": {"id": 368, "name": "I Рестайлинг"},**

**"year": 2020**

**},**

**"addr\_reg": {**

**"kladr": "5400000100002880133",**

**"kladr\_id": "string",**

**"postalCode": "630005",**

**"country": "Россия",**

**"state": "Московская область",**

**"district": "Солнцево",**

**"city": "Москва",**

**"settlement": "string",**

**"street": "ул Ленина",**

**"house": "12",**

**"building": "1",**

**"construction": "Б",**

**"apartment": "121",**

**"subwayStation": "Ясенево",**

**"fiasCode": "54000001000000002880133",**

**"fiasId": "440cf147-d3d8-4744-afbc-d27e93c12fc3",**

**"fiasLevel": 8,**

**"correspondenceFlag": true**

**},**

**"addr\_home": {**

**"kladr": "5400000100002880133",**

**"kladr\_id": "string",**

**"postalCode": "630005",**

**"country": "Россия",**

**"state": "Московская область",**

**"district": "Солнцево",**

**"city": "Москва",**

**"settlement": "string",**

**"street": "ул Ленина",**

**"house": "12",**

**"building": "1",**

**"construction": "Б",**

**"apartment": "121",**

**"subwayStation": "Ясенево",**

**"fiasCode": "54000001000000002880133",**

**"fiasId": "440cf147-d3d8-4744-afbc-d27e93c12fc3",**

**"fiasLevel": 8,**

**"correspondenceFlag": true**

**},**

**"passport": {**

**"num": "4620032618",**

**"division": "540011",**

**"organization": "ОТДЕЛОМ УФМС РОССИИ ПО НОВОСИБИРСКОЙ ОБЛАСТИ В ЦЕНТРАЛЬНОМ РАЙОНЕ Г. НОВОСИБИРСКА",**

**"date": "2013-12-03"**

**},**

**"photos": {**

**"clientPhoto": [**

**{**

**"fileId": "27c2190b-2ea4-4673-822c-ce4fa3969cfa",**

**"fileName": "Фото.jpeg",**

**"contentType": "string"**

**}**

**],**

**"passportMain": [**

**{**

**"fileId": "903a179d-6b30-495d-94bc-7a81d8d64c0e",**

**"fileName": "Паспорт.jpeg",**

**"contentType": "string"**

**}**

**],**

**"passportRegistration": [**

**{**

**"fileId": "903a179d-6b30-495d-94bc-7a81d8d64c0e",**

**"fileName": "ПаспортР.jpg",**

**"contentType": "string"**

**}**

**],**

**"passportPrevious": [**

**{**

**"fileId": "string",**

**"fileName": "string",**

**"contentType": "string"**

**}**

**],**

**"brokerAgreement": [**

**{**

**"fileId": "string",**

**"fileName": "string",**

**"contentType": "string"**

**}**

**]**

**}**

**},**

**}**

{"id":"75f3ac9b989442518f87e806f709593a","values":{"creditInfo":{"carCost":799000,"initialFee":99000,"creditTerm":60,"creditSum":700000,"creditType":"used"},"contact":{"firstName":"\*\*\*\*\*\*","lastName":"\*\*\*\*\*\*\*","middleName":"\*\*\*\*\*\*\*\*\*","gender":"male","email":"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*","mobile":{"number":"\*\*\*\*\*\*\*\*\*\*"}},"car":{"model":{"name":"Rio","id":1980},"brand":{"name":"Kia","id":114},"year":2020}},"step":"sms-confirmation","config":{"name":"cobroker.common.default","version":"0.0.1"},"actions":["autofill"],"triggerComplete":false}

**Ответ:**

**{**

**"values": null,**

**"statusCode": "Ok",**

**"id": "7007a1f6-661f-44a8-b839-45b53a81af5c",**

**"options": {**

**"car": {**

**"brand": "Haval",**

**"model": "Jolion",**

**"year": 2022**

**},**

**"creditInfo": {**

**"carCost": 30000000,**

**"creditType": "new",**

**"initialFee": 100000,**

**"creditTerm": 5**

**}**

**},**

**"status": "shortApp",**

**"step": "finished",**

**"config": {**

**"name": "cobroker.common.default",**

**"version": "0.0.1"**

**}**

**}**

{"values":null,"statusCode":"Ok","id":"75f3ac9b989442518f87e806f709593a","options":{"car":{"brand":"Kia","model":"Rio","year":2020},"creditInfo":{"carCost":799000,"creditType":"used","initialFee":99000,"creditTerm":60}},"status":null,"step":"sms-confirmation","config":{"name":"cobroker.common.default","version":"0.0.1"}}

**Запрос:**

{"id":"75f3ac9b989442518f87e806f709593a","values":{"contact":{"firstName":"\*\*\*\*\*\*","lastName":"\*\*\*\*\*\*\*","middleName":"\*\*\*\*\*\*\*\*\*","gender":"male","birthDate":"\*\*\*\*\*\*\*\*\*\*","birthPlace":"г.Тверь","education":"high","maritalStatus":"single","numberOfChildren":"zero","personalIncome":100000,"familyIncome":100000,"email":"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"},"passport":{"organization":"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*","date":"\*\*\*\*\*\*\*\*\*\*","division":"\*\*\*\*\*\*","num":"\*\*\*\*\*\*\*\*\*\*"},"addr\_reg":{"postalCode":"170001","state":"Тверская обл","city":"г Тверь","street":"ул Виноградова","house":"\*","apartment":"\*\*","detachedHouse":false,"kladr":"6900000100000760001","fiasId":"c74caede-457d-414b-817c-e452c03710ee","fiasLevel":9},"addr\_home":{"postalCode":"170001","state":"Тверская обл","city":"г Тверь","street":"ул Виноградова","house":"\*","apartment":"\*\*","detachedHouse":false,"kladr":"6900000100000760001","fiasId":"c74caede-457d-414b-817c-e452c03710ee","fiasLevel":9}},"step":"contacts","config":{"name":"cobroker.common.default","version":"0.0.1"},"triggerComplete":false}

**Ответ:**  
{"values":{"contact":{"maritalStatus":"single","numberOfChildren":"zero","education":"high","familyIncome":100000,"personalIncome":100000,"email":"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*","gender":"male","birthPlace":"г. Тверь","birthDate":"\*\*\*\*\*\*\*\*\*\*","middleName":"\*\*\*\*\*\*\*\*\*","firstName":"\*\*\*\*\*\*","lastName":"\*\*\*\*\*\*\*","mobile":{"number":"\*\*\*\*\*\*\*\*\*\*"}}},"statusCode":"Ok","id":"75f3ac9b989442518f87e806f709593a","options":{"car":{"brand":"Kia","model":"Rio","year":2020},"creditInfo":{"carCost":799000,"creditType":"used","initialFee":99000,"creditTerm":60}},"status":"uncompleted","step":"contacts","config":{"name":"cobroker.common.default","version":"0.0.1"}}

1. **POST|/api/autobroker/v1/cobroker/app/push**

Данный метод осуществляет отправку обогащенной заявки на рассмотрение. При вызове метода необходима авторизация bearer (в заголовках передаем токен, полученный в методе /create).

**Запрос:**

**{**

**"id": "7007a1f6-661f-44a8-b839-45b53a81af5c"**

**}**

**Ответ:**

**{**

**"values": null,**

**"statusCode": "Ok",**

**"id": "7007a1f6-661f-44a8-b839-45b53a81af5c",**

**"options": {**

**"car": {**

**"brand": "Haval",**

**"model": "Jolion",**

**"year": 2022**

**},**

**"creditInfo": {**

**"carCost": 30000000,**

**"creditType": "new",**

**"initialFee": 100000,**

**"creditTerm": 5**

**}**

**},**

**"status": "fullApp",**

**"step": "finished",**

**"config": {**

**"name": "cobroker.common.default",**

**"version": "0.0.1"**

**}**

**}**

{"values":null,"statusCode":"Ok","id":"51970144-4c02-4c13-9df8-5c436134432a","options":{"car":{"brand":"Hyundai","model":"Solaris","year":2015},"creditInfo":{"carCost":1060000,"creditType":"used","initialFee":0,"creditTerm":60}},"status":"fullApp","step":null,"config":null}

1. **POST|/api/autobroker/v1/cobroker/app/status**

Данный метод возвращает статусы по заявке. При вызове метода требуется авторизация http-signature.

**Запрос:**

{"id":" a0e40691eff24444ad9220346af4b173"}

**Ответ:**

{"id":"a0e40691eff24444ad9220346af4b173","decisions":[{"bankId":"kvk","application":"pending","client":"pending","car":"not\_requested","documents":"not\_requested"}]}

|  |  |  |
| --- | --- | --- |
| **Параметр** | **Описание** | **Варианты** |
| bankId | Название банка | Kvk (Тинькофф) |
| application | Статус по короткой заявке | pending, not\_requested, pre\_approved, approved |
| client | Статус по человеку | pending, not\_requested, approved, cut |
| car | Статус по авто | not\_requested |
| documents | Статус по документам | not\_requested |

**Authorization – HTTP Signature Authentication** - есть некоторый запрос, который партнер хочет отправить API Тинькофф, а мы хотим убедиться: кто отправляет и используем ассиметричную криптографию.

У партнера есть приватный ключ и с этим ключом они считают хэш, подписывают тело своего запроса и этот посчитанный хэш отправляют нам в заголовке http (айди ключа и этот хэш (подись))

У нас будет публичный ключ и мы с помощью него проверяем контрольную сумму – т.е. что это именно партнер подписал.

Партнеру необходимо сгенерировать ключи, **publicKey** - вставляем в нашу базу, **privateKey** – партнер использует в запросе, то есть нам должны прислать **publicKey**. (Для теста и прода ключи разные)

Примеры команд как сгенерировать ключи:

*openssl genrsa -out private.pem 2048*

*openssl rsa -in private.pem -outform PEM -pubout -out public.pem*

У нас есть два примера кода на python и на java как сделать запрос с http signature:

1)

import base64

from datetime import datetime

import json

import requests

import hashlib

import httpsig

from httpsig.requests\_auth import HTTPSignatureAuth

import http.client as http\_client

http\_client.HTTPConnection.debuglevel = 1

preshared\_key\_id = '<id ключа из нашей базы, мы выдаём в обмен на публичный ключ>'

url = '<УРЛ>'

data = {} #тело запроса

if \_\_name\_\_ == "\_\_main\_\_":

  with open('private.pem', 'rb') as fh:

    secret = fh.read()

    digest = hashlib.sha256(json.dumps(data).encode()).digest()

    res = requests.post(

      url,

      json=data,

      headers={"Date": datetime.utcnow().strftime("%a, %d %b %Y %H:%M:%S GMT"), "Digest": ("SHA-256=" + base64.b64encode(digest).decode("utf-8"))},

      auth=HTTPSignatureAuth(secret=secret, key\_id=preshared\_key\_id, algorithm='rsa-sha256', headers=["Date", "Digest"])

    )

    print(res.text)

2) package jhttpsig;

import okhttp3.\*;

import org.bouncycastle.jce.provider.BouncyCastleProvider;

import org.tomitribe.auth.signatures.PEM;

import org.tomitribe.auth.signatures.Signature;

import org.tomitribe.auth.signatures.Signer;

import *[java.io](http://java.io/)*.ByteArrayInputStream;

import *[java.io](http://java.io/)*.IOException;

import java.nio.file.Files;

import java.nio.file.Paths;

import java.security.MessageDigest;

import java.security.NoSuchAlgorithmException;

import java.security.spec.InvalidKeySpecException;

import java.text.SimpleDateFormat;

import java.util.\*;

public class JHttpSig {

    static final MediaType JSON = MediaType.get("application/json; charset=utf-8");

    static final OkHttpClient client = new OkHttpClient();

    static String post(String url, String json, Map<String, String> extraHeaders) throws IOException {

        var body = RequestBody.create(json, JSON);

        var request = new Request.Builder()

                .url(url)

                .post(body)

                .headers(Headers.of(extraHeaders))

                .build();

        try (var response = client.newCall(request).execute()) {

            return response.body().string();

        }

    }

    public static void main(String[] args) throws NoSuchAlgorithmException, IOException, InvalidKeySpecException {

        java.security.Security.addProvider(new BouncyCastleProvider());

        var filePath = Paths.get("private.pem");

        //Let's read key

        var privateKey = PEM.readPrivateKey(new ByteArrayInputStream(Files.readAllBytes(filePath)));

        var url = "<УРЛ>";

        //REQUEST BODY

        var payload = "{}"; //тело запроса

        var keyId = "<id ключа из нашей базы, мы выдаём в обмен на публичный ключ>";

        var dateFormat = new SimpleDateFormat("EEE, dd MMM yyyy HH:mm:ss zzz", Locale.US);

        var signedHeaders = List.of("date", "digest");

        var signature = new Signature(keyId, "hs2019", "rsa-sha256", null, signedHeaders);

        var signer = new Signer(privateKey, signature);

        var headers = new HashMap<String, String>();

        final byte[] digest = MessageDigest.getInstance("SHA-256").digest(payload.getBytes());

        final String digestHeader = "SHA-256=" + new String(Base64.getEncoder().encode(digest));

        headers.put("digest", digestHeader);

        headers.put("date", dateFormat.format(new Date()));

        var signed = signer.sign("POST", url, headers);

        headers.put("authorization", signed.toString());

        // OK, lets do the request

        System.out.println(signed.toString());

        System.out.println(post(url, payload, headers));

    }

}

**Справочник ТС**

**Swagger:**

Тест:  
- <https://qa.tcb-cloud.tinkoff.ru/api/internal/reference/swagger> - API для межсерверных интеграций  
Прод:  
- <http://forma.tinkoff.ru/api/reference/swagger> - публичное API, доступно извне банковской сети  
- <http://tcb-cloud.tinkoff.ru/api/internal/reference/swagger> - API для межсерверных интеграций

**Поиск автомобиля**

**Метод search выбора марки и модели (POST)**/api/internal/reference/car/search

|  |  |
| --- | --- |
| **Параметр** | **Описание** |
| text\* | String. Название автомобиля, принимает на вход возможные варианты:   * марка * модель * марка+модель * название модификации |
| year | Год. При указании в ответе вернет только те автомобили, которые выпускались в этом году |
| brandId | Id марки. Если передать, то вернет только модели, соответствующие данной марке |
| limit | Ограничение на вывод. По умолчанию 50 |

Запрос:

{

"text": "мазда 3",

"limit" : 10

}

Ответ:   
{

    "payload": {

        "items": [

            {

                "brand": 139,

                "model": 159,

                "years": [

                    2003,

                    2004,

                    2005,

                    2006,

                    2007,

                    2008,

                    2009,

                    2010,

                    2011,

                    2012,

                    2013,

                    2014,

                    2015,

                    2016,

                    2017,

                    2018

                ]

            },

            {

                "brand": 139,

                "model": 200,

                "years": [

                    1977,

                    1978,

                    1979,

                    1980,

                    1981,

                    1982,

                    1983,

                    1984,

                    1985,

                    1986,

                    1987,

                    1988,

                    1989,

                    1990,

                    1991,

                    1992,

                    1993,

                    1994,

                    1995,

                    1996,

                    1997,

                    1998,

                    1999,

                    2000,

                    2001,

                    2002,

                    2003

                ]

            },

            {

                "brand": 139,

                "model": 160,

                "years": [

                    2006,

                    2007,

                    2008,

                    2009,

                    2010,

                    2011,

                    2012,

                    2013

                ]

            }

        ],

        "brands": {

            "139": {

                "id": 139,

                "name": "Mazda"

            }

        },

        "models": {

            "159": {

                "id": 159,

                "name": "3",

                "brand": 139

            },

            "160": {

                "id": 160,

                "name": "3 MPS",

                "brand": 139

            },

            "200": {

                "id": 200,

                "name": "323",

                "brand": 139

            }

        }

    },

    "statusCode": "Success"

}

**Метод получения списка поколений (POST)**

/api/internal/reference/car/generations

|  |  |
| --- | --- |
| **Параметр** | **Описание** |
| brandId\* | Id марки |
| modelId\* | Id модели |
| year\* | год |

Запрос:  
{

  "brandId": 139,

  "modelId": 159,

  "year": 2016

}

Ответ:

{

    "payload": {

        "items": [

            {

                "generationId": 657

            },

            {

                "generationId": 656

            }

        ],

        "generations": {

            "656": {

                "id": 656,

                "name": "III (BM)"

            },

            "657": {

                "id": 657,

                "name": "III (BM) Рестайлинг"

            }

        }

    },

    "statusCode": "Success"

}

**Метод получения списка конфигураций (POST)**

/api/internal/reference/car/configurations

|  |  |
| --- | --- |
| **Параметр** | **Описание** |
| brandId\* | Id марки |
| modelId\* | Id модели |
| year\* | год |
| generationId\* | Id поколения. Может отсутствовать |
| allGenerations\* | true/false. Необходимо передавать true в том случае, если не выбирается поколение из метода получения списка поколений, тогда вернутся всевозможные конфигурации по всем поколениям. |

Запрос1:  
{

  "brandId": 139,

  "modelId": 159,

  "year": 2016,

  "generationId": 657,

  "allGenerations": false

}

Запрос2:

{

  "brandId": 139,

  "modelId": 159,

  "year": 2016,

  "allGenerations": true

}

Ответ см. <https://wiki.tcsbank.ru/pages/viewpage.action?pageId=300200895>

**Метод получения приближенной цены (POST)**

/api/internal/reference/car/price

|  |  |
| --- | --- |
| **Параметр** | **Описание** |
| brandId\* | Id марки |
| modelId\* | Id модели |
| generationId | Id поколения. Может отсутствовать |
| bodyId | Id кузова |
| modificationId | Id модификации |
| transmissionId | Id трансмиссии |
| fuelId | Id типа топлива |
| wheelDriveId | Id привода |
| engineId | Id двигателя |
| year\* | Год |

Запрос:

|  |
| --- |
| {  "brandId" : 139,  "modelId" : 159,  "engineId" : 1094,  "year" : 2016  } |

Ответ:

{

    "payload": {

        "avg": 786778.5

    },

    "statusCode": "Success"

}

**Метод получения точной цены (POST)**

/api/internal/reference/car/price

|  |  |
| --- | --- |
| **Параметр** | **Описание** |
| configurationId\* | Id конфигурации |

Запрос:

{

"configurationId" : 174533

}

Ответ:

{

    "payload": {

        "avg": 1119999

    },

    "statusCode": "Success"

}