

Привет.

Потребуется пройти первый этап квеста.

Задание составить и выслать Postman коллекцию (файл JSON) на основе описанной документации. Часть примеров запросов была умышленно скрыта, чтобы ты мог составить запрос самостоятельно по аналогии с прошлыми примерами.

Цель:

С помощью полученной коллекции дойти до этапа “Расчет полиса” в страховой компании и получить ответ.

Примечание

Данные можно использовать свои или тестовые.

На что мы будем обращать внимание?

1. На затраченное время на ТЗ.
2. На соблюдение сроков исполнения.
3. Дополнительную проработку коллекции. (Тест/автоматизация, переменные которые будут использоваться в запросах, окружение)

Host: <https://partner.agentapp.ru/>

Login: qa@qa.qa

Password: 111

api_version: v1

Не забудь сразу после ознакомления с данным документом И до выполнения задания, сообщить оценку в днях или в часах и дату завершения.

Требования:

1. Готовое ТЗ отправляйте на почту mba@b2bpolis.ru с темой “Агент УП - %ФИО% - %Вакансия% - ТЗ”
2. Приложите скрин ответа последнего метода расчета договора.
3. Укажите что было трудно или непонятно в документации
4. Укажите имя коллекции как “ФИО ТЗ AgentApp”

Максимальное время на выполнения задание 8 часов.
Если есть конкретные вопросы сообщите в письме.

AgentApp b2b

МЕТОДЫ ЗАПРОСОВ.	2
АУТЕНТИФИКАЦИЯ	2
Создание водителя	3
Создание собственника	4
Создание страхователя	5
Создание автомобиля	5
Объединение субъектов и объектов страхования в одну сущность	6
Создание договора	6
Обновление Договора в системе AgentApp	7
Получение полного расчёт по СК	7

МЕТОДЫ ЗАПРОСОВ.

- GET - получить объект/объекты. Если не указывать ID объекта, то получаем все доступные объекты. Пример, /dicts/credential_types
- POST - создать объект.
- PUT - обновить полностью объект. Требуется ID объекта
- PATCH - обновить частично объект. Требуется ID объекта
- DELETE - удалить объект. Требуется ID объекта

АУТЕНТИФИКАЦИЯ

- Все запросы, кроме получения токена требуют передачу вместе с запросом заголовка:

Authorization: Token {{token}},

где {{token}} - это токен

- Получение токена осуществляется через метод /users/obtain-token

ТИП ПЕРЕДАВАЕМОЙ И ПОЛУЧАЕМОЙ ИНФОРМАЦИИ (ACCEPT, CONTENT-TYPE)

- Каждый запрос, как ожидает, так и отвечает application/json

POST

{{http_schema}}://{{host}}/{{api_version}}/users/obtain-token

Сервис работает только с аутентифицированным пользователем.

Аутентификация работает через заголовок

"Authorization: Token {{token}}",

где {{token}} - токен, полученный от системы в ответ на данный /account/obtain-token рест

Запрос - POST-запрос с инфой:

- "username" -- логин пользователя
- "password" -- пароль

В ответе получаем "token"

Создание водителя

POST

{{http_schema}}://{{host}}/{{api_version}}/insured_objects/drivers

В ответе находятся параметры по КБМ: "kbm_value" -- коэф. КБМ "is_rsa_checked" -- true/false (ответ из PCA о наличии водителя в базе PCA)

На каждого водителя требуется отдельный запрос.

Из ответа берём "id" -- потребуется при создании сущности "объект страхования"

Body

```
{
  "first_name": "Имя",
  "last_name": "Фамилия",
  "patronymic": "Отчество",
  "birth_date": "1990-01-01",
  "driving_experience_started": "2010-10-10",
  "driver_licenses": [
    {
      "credential_type": "DRIVER_LICENSE",
      "number": "012345",
      "series": "1234",
      "issue_date": "2010-10-10"
    }
  ]
}
```

Создание собственника

POST

{{http_schema}}://{{host}}/v1/insured_objects/owners/natural_persons

Создание собственника ФЛ требует следующих полей:

- "last_name" -- Фамилия
- "first_name" -- Имя
- "patronymic" -- Отчѣтство
- "gender" -- Пол (M,F)
- "birth_date" -- дата рождения в формате "1987-01-13"
- "credential" -- документы. Массив. Первым элементом необходимо добавить российский паспорт
- "address" -- адреса. Массив. Два типа адресов - фактический и юридический. Различаются по параметру "address_type": "ACTUAL_ADDRESS" и "LEGAL_ADDRESS" соответственно.

Указание LEGAL_ADDRESS необходимо для получения расчетов страховых компаний.

Для создания страхового объекта брать ID из поля Person

Body

```
{
  "last_name": "Фамилия",
  "first_name": "Имя",
  "patronymic": "Отчество",
  "birth_date": "1990-01-01",
  "credential": [
    {
      "credential_type": "RUSSIAN_INTERNAL_PASSPORT",
      "issue_date": "2017-03-08",
      "issue_point": "УФМС",
      "issue_point_code": "123-456",
      "number": "123456",
      "series": "1234"
    }
  ],
  "address": [
    {
      "address_query": "г Санкт-Петербург, г Ломоносов, ул Швейцарская, д 1 к 1, кв 1",
      "address_type": "LEGAL_ADDRESS",
      "region_kladr_id": "7800000000000",
      "city_kladr_id": "7800000600000"
    },
    {
      "address_query": "г Санкт-Петербург, г Ломоносов, ул Швейцарская, д 1 к 1, кв 1",
      "address_type": "ACTUAL_ADDRESS",
      "region_kladr_id": "7800000000000",
      "city_kladr_id": "7800000600000"
    }
  ]
}
```

```
]
}
```

Создание страхователя

Аналогично полному обновлению Собственника (физлицо)

POST

{{http_schema}}://{{host}}/{{api_version}}/insured_objects/insurants/natural_persons

```
{
  "last_name": "Фамилия",
  "first_name": "Имя",
  "patronymic": "Отчество",
  "birth_date": "1990-01-01",
  "credential": [
    {
      "credential_type": "RUSSIAN_INTERNAL_PASSPORT",
      "issue_date": "2010-10-10",
      "issue_point": "УФМС",
      "issue_point_code": "123-456",
      "number": "123456",
      "series": "1234"
    }
  ],
  "address": [
    {
      "address_query": "г Санкт-Петербург, г Ломоносов, ул Швейцарская, д 1 к 1, кв 1",
      "address_type": "LEGAL_ADDRESS",
      "region_kladr_id": "7800000000000",
      "city_kladr_id": "7800000600000"
    },
    {
      "address_query": "г Санкт-Петербург, г Ломоносов, ул Швейцарская, д 1 к 1, кв 1",
      "address_type": "ACTUAL_ADDRESS",
      "region_kladr_id": "7800000000000",
      "city_kladr_id": "7800000600000"
    }
  ]
}
```

Для создания страхового объекта брать ID из поля Person

Создание автомобиля

POST

{{http_schema}}://{{host}}/v3/insured_objects/cars

Создание автомобиля требует следующих полей:

- "car_model" -- id Модели
- "engine_power" -- мощность двигателя в л.с. Уже присутствует, при PATCH'e не требуется указывать.
- "chassis_number" -- Номер шасси (не обязательно)
- "car_body_number" -- Номер кузов (прицеп) (не обязательно)
- "vin_number" -- ВИН (не обязательно)

- "number_plate" -- Номерной знак (не обязательно)
- "manufacturing_year" -- год выпуска в формате "2010"
- "max_mass" -- максимальная масса (для категории Д)
- "credential" -- документ.

Необходимо указать минимум один из четырёх параметров:

- "vin_number" -- ВИН (не обязательно)
- "number_plate" -- Номерной знак (не обязательно)

В ответе придут:

- "id"

```
{
  "car_model_id": 864026180,
  "engine_power": 211,
  "chassis_number": null,
  "car_body_number": null,
  "vin_number": "WAUZZZ8T4BA037241",
  "number_plate": "P904MX178",
  "manufacturing_year": 2010,
  "max_mass": null,
  "credential": [
    {
      "credential_type": "VEHICLE_REGISTRATION",
      "issue_date": "2010-11-01",
      "number": "267461",
      "series": "78УН"
    }
  ]
}
```

Объединение субъектов и объектов страхования в одну сущность “объект страхования”

{{http_schema}}://{{host}}/{{api_version}}/insured_objects/

Создаём техническую сущность - "объект страхования".

- "drivers" -- массив водителей
- "owner" -- собственник
- "car" -- автомобиль
- "insurant" -- Страхователь
-

Из ответа берём "id" -- потребуется при создании договора

Создание договора

POST

`{{http_schema}}://{{host}}/v3/agreements/calculations`

Параметры:

- `valid_from` -- начало действия полиса. В системе срок страхования = 1 год, поэтому разница между датой `valid from` и `valid to` должна составлять 1 год. Например, если `valid from` "2019-06-30", дата `valid to` должна быть "2020-06-29"
- `valid_to` -- окончание действия полиса
- `insurance_period` -- id периода (длительность периода страховки) = 11
- `target_of_using` -- цель использования
- `drivers_ids` -- массив водителей
- `is_car_without_registration` -- ТС без регистрации
- `engine_power` -- мощность ТС
- `has_car_trailer` -- ТС с прицепом
- `car_type` -- тип ТС
- `owner_registration` / `address_query`: --полный адрес регистрации собственника. Формируется аналогично адресу у Собственника. (тип строка см запрос собственник)

Если был выбран период страхования "1 год", то передается пустой массив `periods`. А если был выбран другой период, то в массиве `periods` передаются параметры:

- `start_date` -- начало периода
- `end_date` -- окончание периода

Период не может выходить за даты начала и окончания договора. Возможные периоды: 3-9 месяцев, 1 год.

В ответе получаем:

- `id` -- идентификатор договора. Этот `id` понадобится для дальнейших запросов.
- `calculation` -- информация о расчёте
- `calculation.premium` -- страховая премия

Обновление Договора в системе AgentApp

PATCH

`{{http_schema}}://{{host}}/{{api_version}}/agreements/{{agreement_id}}`

Необходимые параметры:

- `"insured_object"` -- id сущности `InsuredObject`

В ответе получаем:

- `id` -- сохраняем как `agreement_id`, дальше будем использовать.

Получение полного расчёт по СК

POST

{{http_schema}}://{{host}}/{{api_version}}/agreements/{{agreement_id}}/results/RENAISSANCE

По каждой страховой компании необходимо отправить данный запрос. В адресной строке параметры:

- "agreement_id" -- id Договора в системе AgentApp
- "ins_company_code" -- код Страховой компании. Доступные коды: ZETTA, ALPHA_STRAH, SDS, RENAISSANCE, RGS, SNGI, UGORIA, INGOSSTRAH, TINKOFF

В положительном ответе (статус 200) получаем параметры:

- "integrated_company_title" -- наименование Страховой Компании
- "integrated_company_code" -- code Страховой Компании
- "premium" -- Премия

Не забудь сразу после ознакомления с данным документом И до выполнения задания, сообщить оценку в днях или в часах и дату завершения.