

Инструкция по подключению и использованию API

1. Общее описание	2
1.1. Как получить доступ к API	3
1.2. Демо доступ	5
2. Отправка заявки	6
3. Статусы заявок	9
3.1. Получение списка заявок	10
3.2. Получение информации по заявке	11
4. Проверка клиента на дубли	13
5. Проверка клиента по кредитному скорингу	15
5.1. Отправка запроса на проверку	15
5.2. Получение результата проверки	15
6. Сервис получения справочников	19
6.1. Получение справочника городов и регионов	19
6.2. Получение справочника отделений	21
7. Описание ошибок	23
8. Справочники	26
8.1. Справочник продуктов (productCode)	26
8.2. Справочник городов (cityCode)	26
8.3. Справочник регионов (regionCode)	28

1. Общее описание

API – инструмент для автоматизации процесса передачи заявок в рамках партнерского договора.

Протокол API: REST

Формат запроса и ответа: JSON

Кодировка символов: UTF-8

Обязательные заголовки: в каждом запросе должно быть обязательно два заголовка:

```
Content-Type: application/json; charset=UTF-8
API-key: *****
```

Как получить значение api-key описано в п. 1.1

Ответ API: при успешной обработке запроса система ответит HTTP кодом **200 OK**

При ошибке – кодом ошибки и описанием ошибки (см. п. 7. Описание ошибок)

1.1. Как получить доступ к API

Для получения доступа к API необходимо авторизоваться в личном кабинете (<https://partner.alfabank.ru>) используя опцию входа через «интернет банк».

Перейти в раздел «Инструменты» / «API» в боковом меню (или перейти сразу по [ссылке](#)). В данном разделе увидите свой ключ api-key и срок его действия. Также вы можете задать Email сотрудника, ответственного за настройку API, которому будут приходить письма на указанную почту, если срок действия API-ключа истекает (ключ на почту не приходит, только информация об его истечении).

API

Инструмент для автоматизации передачи заявок в рамках партнерского договора. Вы можете интегрироваться и автоматизировать все функции работы личного кабинета в своей CRM системе.

Все заявки переданные через API вы сможете посмотреть в личном кабинете. [Инструкция по интеграции с API](#).

API Ключ

1162*****	15.04.2021 11:54:32
-----------	---------------------

Email сотрудника
Не задан [Задать](#)

[Перевыпустить](#) [Отключить](#)

Создайте ключ и передайте инструкцию разработчикам своей CRM системы для интеграции.
Для удобства тестирования и разработки вы можете использовать API Swagger

Если у вас еще нет ключа, то вы можете его создать. Укажите Email человека ответственного за настройку API и нажмите на кнопку «Получить ключ API».

API

Инструмент для автоматизации передачи заявок в рамках партнерского договора. Вы можете интегрироваться и автоматизировать все функции работы личного кабинета в своей CRM системе.

Все заявки переданные через API вы сможете посмотреть в личном кабинете. [Инструкция по интеграции с API](#).

[Получить ключ API](#)

Email человека ответственного за настройку API

Создайте ключ и передайте инструкцию разработчикам своей CRM системы для интеграции.

Система сгенерирует случайный ключ. Его нужно скопировать (предварительно нажав на «глазик») и передать специалисту, настраивающему интеграцию по АПИ.

API Ключ

1162*****🕒 15.04.2021 11:54:32

Email сотрудника
test@alfabank.ru Изменить

Перевыпустить Отключить

Если вам необходимо изменить Email сотрудника, то нажмите на соответствующую кнопку.

Ведите новый Email и нажмите «Сохранить». Email будет изменен без перевыпуска ключа API.

API

Инструмент для автоматизации передачи заявок в рамках партнерского договора. Вы можете интегрироваться и автоматизировать все функции работы личного кабинета в своей CRM системе.

Все заявки переданные через АПИ вы сможете посмотреть в личном кабинете. [Инструкция](#) по интеграции с API.

API Ключ

1162*****🕒 15.04.2021 11:54:32

NEW test@alfabank.ru Сохранить

Email человека ответственного за настройку API

Перевыпустить Отключить

Создайте ключ и передайте инструкцию разработчикам своей CRM системы для интеграции.

Для удобства тестирования и разработки вы можете использовать [API Swagger](#)

У ключа api-key срок действия – **1 год**.

При компрометации ключа или приближении срока истечения его нужно сбросить и использовать новый.

Важно: при сбросе ключа, старый ключ мгновенно перестает действовать.

1.2. Демо доступ

Для удобства тестирования и разработки вы можете использовать [API Swagger](#) и/или api-key для демо доступа.

В демо доступе доступно два метода:

- [2. Отправка заявки](#)
- [4. Проверка клиента на дубли](#)

api-key демо-доступа: **aaaaaaaa-aaaa-aaaa-aaaaaaaaaaaaa**.

Для метода создания всегда возвращается идентификатор созданной заявки.

Пример ответа:

```
HTTP/1.1 200 Ok
Content-Type: application/json; charset=UTF-8
Date: Wed, 03 Apr 2019 15:32:40 GMT
{
  "id": "aaaaaaaa-aaaa-aaaa-aaaaaaaaaaaaa"
}
```

В демо-доступе для метода проверки по ИНН, ошибка вернется для любых ИНН начинающегося с 1. Для всех остальных вариантов, запрос будет обработан без ошибок.

2. Отправка заявки

POST <https://partner.alfabank.ru/public-api/v2/leads>

Запрос отправляет заявку в Альфа-Банк. Параметры содержимого запроса:

	Параметр	Описание	Обязательный	Формат
	organizationInfo	Организация	Да	
1	organizationName	Наименование ЮЛ\ИП	Да	string
2	inn	ИНН	Да	string (10 или 12 цифр) ^(\d{10})\$ ^(\\d{12})\$ ^\$
3	ogrn	ОГРН или ОГРНИП	Да (если есть блок reserveAccount)	string (13 или 15 цифр) ^(\d{13})\$ ^(\\d{15})\$ ^\$
4	kpp	КПП для ООО	Да (если есть блок reserveAccount)	string (9 цифр) \d{9}
	contactInfo	Массив данных о контактах	Да	
5	fullName	ФИО одной строкой		string
6	phoneNumber	Номер телефона основной	Да	string (11 цифр: 7\d{10})
7	contactEmail	Email		string
8	contractorsPosition	Должность		string
9	extraPhoneNumbers	Доп. номера телефонов		string[] (11 цифр: 7\d{10})
	requestInfo	Данные заявки	Да	
10	advCode*	Код партнера/тип заявки*		string
11	platformId**	Идентификатор платформы**		string
12	comment	Комментарий к заявке		string
13	cityCode	Код города (ФИАС)	Да	Справочник городов
	productInfo	Массив данных о продуктах	Да	
14	[productCode]	Код продукта	Да	Справочник продуктов
	reserveAccount	Резервирование счета	Нет	
15	branchId	Код отделения	Да (только если есть блок reserveAccount)	Справочник отделений
16	address	Адрес организации		string
17	email	Email на который уйдет письмо о резерве		string

* , ** поля **advCode** и **platformId** передавать нужно, только если у вас есть договоренность с банком о передаче этого полей, в противном случае их передавать не нужно.

В случае успешной обработки запроса в ответе возвращаются HTTP код **200 OK** и номер созданного запроса **id**, который нужно сохранить для дальнейшего получения статуса заявки.

Если в запросе передан блок **reserveAccount** — то клиенту зарезервируется счет и на **email** будут отправлены реквизиты расчетного счета.

Поле **cityCode** важно передавать корректным, так как на основе значения этого поля вычисляется часовой пояс для звонка клиенту.

Пример запроса:

```
$ curl https://partner.alfabank.ru/public-api/v2/leads \
-H "Content-Type: application/json; charset=UTF-8" \
-H "API-key: *****" \
-d '{
    "organizationInfo": {
        "organizationName": "ООО Тестовая компания",
        "inn": "9685952780",
        "ogrn": "7037770735259",
        "kpp": "780301031"
    },
    "contactInfo": [
        {
            "fullName": "Иванов Иван Иванович",
            "phoneNumber": "79000000111",
            "contactEmail": "mail@example.com",
            "extraPhoneNumbers": [
                "79000000222",
                "79000000333"
            ]
        }
    ],
    "requestInfo": {
        "advCode": "advcode_1",
        "platformId": "platform",
        "comment": "Тестовая заявка. Не звонить!",
        "cityCode": "7dfa745e-aa19-4688-b121-b655c11e482f"
    },
    "productInfo": [
        {
            "productCode": "LP_RKO"
        },
        {
            "productCode": "LP_ACQ_TR"
        },
        {
            "productCode": "LP_LOAN_BUS"
        }
    ],
    "reserveAccount": {
        "branchId": "0126",
        "address": "г. Москва, ул. Маломосковская, д. 16 стр. 9 пом. 14",
        "email": "mail@example.com"
    }
}'
```

! Важно: обратите внимание, что блоки **contactInfo** и **productInfo** обязательны и передаются как **массив**.

! Важно: при отправке заявки с кредитным продуктом нужно обязательно проводить предварительный кредитный скоринг клиента (см. п. 5. Проверка клиента по кредитному скорингу), в противном случае система вернет ошибку (см. ошибку 17 п. 7. Описание ошибок).

Пример ответа:

```
HTTP/1.1 200 Ok
Content-Type: application/json; charset=UTF-8
Date: Wed, 03 Apr 2019 15:32:40 GMT
{
  "id": "25411f46-e40c-74df-bdd8-74b79fd75b34"
}
```

3. Статусы заявок

В системе существует два статуса: статус заявки и статус продукта/сделки.

Т.к. в заявке может быть много продуктов/сделок и срок открытия их разный, то смотреть необходимо на оба статуса.

- **Статус заявки и детальный статус заявки**

- необходимо ориентироваться на него если по заявке получен отказ. В детальном статусе можно понять причину отказа;
- промежуточный статус «В работе». В детальном статусе можно увидеть этап заявки: клиент на перезвоне или встреча с менеджером;
- итоговый статус заявки – «Работа завершена» (по всем продуктам/сделкам есть финальный статус - оформлен или отказ);

- **Статус продукта/сделки**

- необходимо ориентироваться на него если заявка находится в статусе «В работе» или «Работа завершена»;
- итоговый статус продукта/сделки – Продукт оформлен (для РКО означает что счет открыт, для кредита – кредит выдан. Также передается сумма выданного кредита (поле *dealFinAmount*).

Таблица статусов (state) заявок и продуктов

Статус заявки	Статус продукта	Описание
SENT	NEW	Заявка отправлена. Технический статус, который мгновенно меняется.
DECLINED	DECLINED	Отказ по заявке (тип отказа указан в status. Детальная причина указана в detailedStatus)
IN_PROGRESS	UNDEFINED	«В работе» Заявка поступила в работу. Расшифровка статуса заявки указана в поле в поле status. Статус продукта не определен.
	IN_PROGRESS	«В работе» Заявка поступила в работу. Расшифровка статуса заявки и указана в поле status detailedStatus. Продукта в поле status
	DECLINED	В заявке было больше одного продукта. По одному отказ, а остальные «в работе»
	COMPLETED	В заявке больше 1 продукта. Часть продуктов оформлены (для РКО - счет открыт) остальные в работе.
CLOSED	DECLINED	Работа с заявкой завершена. Продукт закрыт отказом.
	COMPLETED	Работа с заявкой завершена. Продукт оформлен / для РКО - счет открыт.
	UNDEFINED	Работа с заявкой завершена. Продукт не удалось открыть.

В системе есть два метода получения статуса заявок:

- медленный: получения списка заявок;
- быстрый: получения информации по одной заявке.

3.1. Получение списка заявок

GET <https://partner.alfabank.ru/public-api/v2/leads>

Запрос возвращает список uid заявлений с их статусами. Если параметры запроса **fromDate** и **toDate** не были указаны, то по умолчанию список заявлений будет отдан за 1 календарный месяц. Данный метод возвращает заявки, переданные с 22/07/2020, заявки отправленные ранее могут отобразиться не полностью.

Параметры запроса:

	Параметр	Описание	Обязательный	Формат
1	pageNumber	Номер запрашиваемой страницы	Да	number
2	perPage	Количество записей на страницу	Да (max=100)	number
3	fromDate	Дата, с которой будут браться заявки	Нет	number (unix timestamp ms)
4	toDate	Дата, по которую будут браться заявки	Нет	number (unix timestamp ms)
5	filterDateBy	Сортировка для указанных дат. Например, получить все изменившиеся за вчера. Или получить все созданные вчера	Нет	modifiedAt / createdAt

Пример запроса:

```
$ curl https://partner.alfabank.ru/public-api/v2/leads?pageNumber=0&perPage=10&fromDate=1586957555000&toDate=1589117555000&filterDateBy=modifiedAt
HTTP/1.1[\r][\n] \
-H "Accept: application/json; charset=UTF-8" \
-H "API-key: *****"
```

Пример ответа:

```
HTTP/1.1 200 Ok
Content-Type: application/json; charset=UTF-8
Date: Wed, 03 Apr 2019 15:32:40 GMT
[
{
  "uuid": "b5ddf6ff-eacb-493a-836d-19539ecd3b5f",
  "state": "IN_PROGRESS",
  "status": "В работе",
  "detailedStatus": "Перезвонить. Не отвечает.",
  "productInfo": [
    {
      "productCode": "LP_RKO",
      "state": "IN_PROGRESS",
      "status": "В работе"
    },
    {
      "productCode": "LP_AKASSA",
      "state": "DECLINED",
      "status": "Отказ клиента"
    }
  ],
  {...}
}
```

Рекомендация! Для получения статусов заявлений рекомендуется вызывать раз в день сервис получения заявлений с сортировкой по изменениям «filterDateBy=modifiedAt» с

датой со вчера по сегодня. Тем самым в выборку попадут в том числе старые заявки, созданные давно, но закрывшиеся в этом интервале.

3.2. Получение информации по заявке

GET <https://partner.alfabank.ru/public-api/v2/leads/{id}>

Запрос возвращает статус заявки с номером *uuid* (тот же что и id).

Пример запроса:

```
$ curl https://partner.alfabank.ru/public-api/v2/leads/25411f46-e40c-74df-bdd8-74b79fd75b34 \
-H "Accept: application/json; charset=UTF-8" \
-H "API-key: *****"
```

Параметры ответа:

Параметр	Описание	Обязательный	Формат
1 uuid	Идентификатор заявки	Да	string
2 state	Статус технический	Да	string
3 status	Статус заявки	Да	string
4 detailedStatus	Подробный статус заявки	Да	string
5 productInfo	Массив данных о продуктах из заявки	Да	массив
6 productCode	Код продукта	Да	string
7 state (из массива productInfo)	Статус по продукту технический	Да	string
8 status (из массива productInfo)	Статус по продукту	Да	string
9 productParams	Параметры продукта	Нет	string
10 dealFinAmount	Сумма кредитного продукта	Нет	integer

Пример ответа:

```
HTTP/1.1 200 Ok
Content-Type: application/json; charset=UTF-8
Date: Wed, 03 Apr 2019 15:32:40 GMT
{
    "uuid": "25411f46-e40c-74df-bdd8-74b79fd75b34",
    "state": "DECLINED",
    "status": "Отказ клиента",
    "detailedStatus": "Временная приостановка деятельности",
    "productInfo": [
        {
            "productCode": "LP_RKO",
            "state": "DECLINED",
            "status": "Отказ клиента"
        },
        {...}
    ]
}
```

Пример ответа с суммой кредита:

```
HTTP/1.1 200 Ok
Content-Type: application/json; charset=UTF-8
Date: Wed, 03 Apr 2019 15:32:40 GMT
{
    "uuid": "25411f46-e40c-74df-bdd8-74b79fd75b34",
    "productInfo": [
        {
            "productCode": "LP_RKO",
            "state": "IN_PROGRESS",
            "status": "В работе",
            "productParams": null
        },
        {
            "productCode": "LP_LOAN_BUS",
            "state": "COMPLETED",
            "status": "Продукт оформлен",
            "productParams": {
                "dealFinAmount": 1230000
            }
        }
    ],
    "state": "IN_PROGRESS",
    "status": "В работе",
    "detailedStatus": null
}
```

4. Проверка клиента на дубли

POST <https://partner.alfabank.ru/public-api/v2/checks>

Данный метод предназначен в первую очередь партнерам call-центрам, для проверки клиента перед звонком. Проверка осуществляется по ИНН или телефону на банковский скоринг и на дубликаты по продукту РКО.

При больших объемах проверок, данный метод можно вызывать в многопоточном режиме – до 5 потоков.

Параметры содержимого запроса:

	Параметр	Описание	Обязательность	Формат
	organizationInfo	Организация		
1	inn	ИНН	Да	string (10 или 12 цифр)
	contactInfo	Массив данных о номерах телефона		
2	[phoneNumber]	Номер телефона	Да	string (11 цифр: 7\d{10})
	productInfo	Массив данных о продуктах		
3	[productCode]	Код продукта	Да	«LP_RKO» - для РКО «LP_LOAN_BUS» - по кредитам

! Важно: поле «Код продукта» нужно передавать строго по формату. Пустое значение трактуется как пустой продукт и выдаст неверный результат. Все возможные значения данного поля приведены в [8.1. Справочник продуктов \(productCode\)](#).

В случае если запрос был обработан без ошибок и заявок с данным ИНН/Телефоном не найдено (можно звонить Клиенту) система отвечает HTTP кодом **200 OK**

Во всех остальных случаях система отвечает ошибкой (см. описание ошибок).

Пример запроса:

```
$ curl https://partner.alfabank.ru/public-api/v2/checks \
-H "Content-Type: application/json; charset=UTF-8" \
-H "API-key: *****" \
-d '{
  "organizationInfo": {
    "inn": "9685952780"
  },
  "contactInfo": [
    {
      "phoneNumber": "79371942512"
    }
  ],
  "productInfo": [
    {
      "productCode": "LP_RKO"
    }
  ]
}'
```

или без телефона, если по каким-то причинам его нет:

```
$ curl https://partner.alfabank.ru/public-api/v2/checks \
-H "Content-Type: application/json; charset=UTF-8" \
-H "API-key: *****" \
-d '{
  "organizationInfo": {
    "inn": "9685952780"
  },
  "productInfo": [
    {
      "productCode": "LP_RKO"
    }
  ]
}'
```

Пример успешного ответа – не дубль, можно звонить клиенту:

```
HTTP/1.1 200 ok
Content-Type: application/json; charset=UTF-8
Date: Wed, 03 Apr 2019 15:32:40 GMT
{}
```

Пример неуспешного ответа – дубль, звонить запрещено:

```
HTTP/1.1 403 forbidden
Content-Type: application/json; charset=UTF-8
Date: Wed, 03 Apr 2019 15:32:40 GMT
{
  "errors": [
    {
      "status": 403,
      "code": "UNACCEPTABLE_COMPANY",
      "title": "Unacceptable company",
      "detail": "Unfortunately, the Bank cannot accept the company with the provided TIN [5609187
820]."
    }
  ]
}
```

5. Проверка клиента по кредитному скорингу

Проверка клиента выполнена в виде асинхронного сервиса, то есть сначала необходимо отправить запрос на проверку, который вернет в ответ присвоенный уникальный идентификатор. Через определенный промежуток времени (рекомендуем не более 10 попыток каждые 30 секунд), необходимо запросить результат проверки, передав ранее полученный идентификатор.

! Важно: доступ к данным проверкам предоставляется по запросу.

5.1. Отправка запроса на проверку

POST <https://partner.alfabank.ru/public-api/v2/scoring>

Параметры запроса:

Параметр	Описание	Обязательный	Формат
1 inn	ИНН проверяемой организации	Да	string (10 или 12 цифр)
2 okved	ОКВЭД организации	Нет	string
3 phone	Контактный номер телефона организации	Нет	string (11 цифр: 7\d{10})

Пример запроса:

```
$ curl https://partner.alfabank.ru/public-api/v2/scoring \
-H "Content-Type: application/json; charset=UTF-8" \
-H "API-key: *****" \
-d '{
  "inn": "1831139275",
  "okved": "01.11.11",
  "phone": "79261234578"
}'
```

Пример ответа:

```
HTTP/1.1 200 Ok
Content-Type: application/json; charset=UTF-8
Date: Wed, 03 Apr 2019 15:32:40 GMT
{"requestId":"414d512042524b303120202020202020f056e4527e91cf"}
```

5.2. Получение результата проверки

GET <https://partner.alfabank.ru/public-api/v2/scoring/{id}>

В качестве параметра {id} передается идентификатор **requestId**, полученный на предыдущем шаге, при отправке запроса.

Пример запроса:

```
$ curl https://partner.alfabank.ru/public-
api/v2/scoring/414d512042524b3031202020202020205f056e4527e91cf \
-H "Accept: application/json; charset=UTF-8" \
-H "API-key: *****"
```

Параметры ответа:

	Параметр	Описание	Обязательный	Формат
1	requestId	Идентификатор, полученный на предыдущем шаге	Да	string
2	decision	Решение по скорингу	Да	string
3	limits	Массив предодобренных приложений	Да	массив
4	monthsNumber	Срок выдачи предодобренного предложения	Нет	integer
5	prodCode	Код предодобренного кредитного продукта	Нет	string
6	loanProduct	Техническое поле (название)	Нет	string
7	productDescription	Описание предодобренного кредитного продукта	Нет	string
8	limitEndDate	Дата окончания предодобренного предложения (по умолчанию null)	Нет	null
9	interestRate	Техническое поле (ставка)	Нет	float
10	limitMaxSum	Максимальная сумма кредита	Нет	массив
11	amount	Сумма кредита	Нет	integer
12	product	Название предодобренного кредитного продукта	Нет	string
13	creditRate	Процентная ставка по предодобренному предложению	Нет	float
14	companyCreditRating	Кредитный рейтинг ЮЛ-клиента	Да	string

Рекомендация! В случае creditRate = null и product = null, смотреть на значения из технических полей.

Пример ответа, в случае, если вернулась информация по лимитам:

```
HTTP/1.1 200 Ok
Content-Type: application/json; charset=UTF-8
Date: Wed, 03 Apr 2019 15:32:40 GMT
{
  "requestId": "414d512042524b3031202020202020205f056e4527e91cf",
  "decision": "ACCEPT",
  "limits": [
    {
      "monthsNumber": 12,
      "prodCode": "UP03",
      "loanProduct": "Овердрафт",
      "productDescription": "Возможность уходить по счёту в минус и пользоваться средствами банка , если срочно нужны деньги.",
      "limitEndDate": null,
    }
  ]
}
```

```
"interestRate": 30.5,
"limitMaxSum": {
    "amount": 1970000
},
"product": "Овердрафт",
"creditRate": 29.5
},
{
    "monthsNumber": 18,
    "prodCode": "VMS2",
    "loanProduct": "Возобновляемая кредитная линия",
    "productDescription": "Многоразовый лимит. Вы можете брать необходимую сумму в пределах лимита и установленного срока. Платите один раз в месяц. Когда выплатите лимит полностью, он возобновится.",
    "limitEndDate": null,
    "interestRate": 29,
    "limitMaxSum": {
        "amount": 1970000
    },
    "product": "Возобновляемая кредитная линия",
    "creditRate": 28
},
{
    "monthsNumber": 36,
    "prodCode": "UL01",
    "loanProduct": "Залоговый Лайт",
    "productDescription": "Кредит на развитие бизнеса с залогом. Зачислим на ваш расчётный счёт",
    "limitEndDate": null,
    "interestRate": 22,
    "limitMaxSum": {
        "amount": 2690000
    },
    "product": "Залоговый лайт - минимальный лимит",
    "creditRate": 22
},
{
    "monthsNumber": 60,
    "prodCode": "UL60",
    "loanProduct": "Залоговый Лайт",
    "productDescription": "Кредит на развитие бизнеса с залогом. Зачислим на ваш расчётный счёт",
    "limitEndDate": null,
    "interestRate": 22,
    "limitMaxSum": {
        "amount": 3820000
    },
    "product": "Залоговый лайт - максимальный лимит",
    "creditRate": 22
}
],
"companyCreditRating": null
}
```

Пример положительного ответа, в случае, если НЕ вернулась информация по
ЛИМИТАМ:

```
HTTP/1.1 200 Ok
Content-Type: application/json; charset=UTF-8
Date: Wed, 03 Apr 2019 15:32:40 GMT
{
    "requestId": "414d512042524b3031202020202020205f056e4527e91cf",
    "decision": "ACCEPT",
```

```
"limits": null,  
"companyCreditRating": null  
}
```

Пример отрицательного ответа:

```
HTTP/1.1 200 Ok  
Content-Type: application/json; charset=UTF-8  
Date: Wed, 03 Apr 2019 15:32:40 GMT  
{  
    "requestId": "414d512042524b3031202020202020205f056e4527e91cf",  
    "decision": "DECLINE",  
    "limits": null,  
    "companyCreditRating": null  
}
```

6. Сервис получения справочников

6.1. Получение справочника городов и регионов

GET <https://partner.alfabank.ru/public-api/v2/dictionaries?code=<код справочника>>

Запрос возвращает полностью справочник городов или регионов, в зависимости от указанного значения **code** (*regions* или *cities*). Справочники обновляются не чаще чем один раз в сутки, происходит это в ночное время,

Важно! Не рекомендуется вызывать справочник **чаще одного раза в день**.

- **code=cities**

Сервис вернет справочник городов. Формат ответа сервиса в таблице ниже.

	Параметр	Описание	Пример
1	fias	Код фиас	42a02e11-a337-4d50-8596-fc76dae7c62a
2	id	Внутренний идентификатор	30
3	name	Значение	Абакан
4	regionCode	Код региона	19
5	regionFias	Код региона ФИАС	8d3f1d35-f0f4-41b5-b5b7-e7cadf3e7bd7
6	bmb	Город присутствия банка (город в котором есть отделения)	false
7	remote	Удаленный город (города, в которых выезжают курьеры для открытия счета)	true
8	satellite	Город-спутник (выезжают менеджеры из основного города присутствия)	false

- **code=regions**

Сервис вернет справочник регионов. Формат ответа сервиса в таблице ниже.

	Параметр	Описание	Пример
1	fias	Код фиас региона	d02f30fc-83bf-4c0f-ac2b-5729a866a207
2	id	Внутренний идентификатор	2
3	name	Значение	Камчатский край
4	Code	Код региона	41
5	mnem	Внутренний идентификатор	MOZQ

Пример запроса:

```
$ curl https://partner.alfabank.ru/public-api/v2/leads/dictionaries?code=regions' \
-H "Accept: application/json; charset=UTF-8" \
-H "API-key: *****"
```

Пример ответа:

```
HTTP/1.1 200 Ok
Content-Type: application/json; charset=UTF-8
Date: Wed, 08 Apr 2019 15:32:40 GMT
{
  "id": "5cfa432f9194be0001c36d28",
  "code": "regions",
  "name": "Справочник регионов",
  "values": [
    {
      "id": 2,
      "name": "Камчатский край",
      "fias": "d02f30fc-83bf-4c0f-ac2b-5729a866a207",
      "mnem": "MOZQ",
      "code": "41"
    },
    {...},
  ],
  "createdAt": "07-06-2019 13:57:51.958",
  "modifiedAt": "05-07-2019 15:32:33.068",
  "deleted": false
}
```

Запрос для городов

```
$ curl https://partner.alfabank.ru/public-api/v2/leads/dictionaries?code=cities' \
-H "Accept: application/json; charset=UTF-8" \
-H "API-key: *****"
```

Пример ответа:

```
HTTP/1.1 200 Ok
Content-Type: application/json; charset=UTF-8
Date: Wed, 08 Apr 2019 15:32:40 GMT
{
  "id": "5c5ac4effe7ee20001a29ae3",
  "code": "cities",
  "name": "Справочник городов",
  "values": [
    {
      "name": "Одинцово",
      "id": 521,
      "courier": true,
      "mnem": "PTCO",
      "fias": "6ae8ad9e-aa6c-4849-92d8-8e637059c9b0",
      "regionCode": "50",
      "regionFias": "29251dcf-00a1-4e34-98d4-5c47484a36d4",
      "bmb": true,
      "remote": false,
      "satellite": true
    },
    {...},
  ],
  "createdAt": "10-06-2019 13:57:51.958",
  "modifiedAt": "05-07-2019 15:32:33.068",
  "deleted": false
}
```

6.2. Получение справочника отделений

GET <https://partner.alfabank.ru/public-api/v2/branches/{fiasId}>

Справочник нужен только для отправки заявок с предварительным резервом счета.

Входной параметр — *fiasId* города (берем из справочника городов [6. Сервис получения справочников](#) или как справочник ФИАС на сайте <https://fias.nalog.ru/>)

Запрос возвращает список отделений в указанном городе. Если в городе нет отделений, то список вернется пустым.

Формат ответа сервиса в таблице ниже.

	Параметр	Описание	Пример
1	uuid	Идентификатор офиса, автоматически сгенерированный при получении из мастерсистемы	5c027cc1-896b-4042-bfb5-922a8ee9a788
2	pid	Внутрибанковский цифровой код офиса	0237
3	mnemonic	Мнемоника	МОQJ
4	title	Наименование для сайта	ДО «АВТОЗАВОДСКИЙ ПРОЕЗД»
6	lat	Широта географическая	55.708496
7	lon	Долгота географическая	37.66232
8	street	Улица	ул Автозаводская
9	house	Дом	5

Пример запроса:

```
$ curl https://partner.alfabank.ru/public-api/v2/branches/0c5b2444-70a0-4932-980c-b4dc0d3f02b5 \
-H "Accept: application/json; charset=UTF-8" \
-H "API-key: *****"
```

Пример ответа:

```
HTTP/1.1 200 Ok
Content-Type: application/json; charset=UTF-8
Date: Wed, 08 Apr 2020 15:32:40 GMT
{
    "uuid": "5c027cc1-896b-4042-bfb5-922a8ee9a788",
    "pid": "0237",
    "mnemonic": "MOQJ",
    "title": "ДО «АВТОЗАВОДСКИЙ ПРОЕЗД»",
    "locations": [
        {
            "lat": 55.708496,
            "lon": "37.66232",
            "address": {
                "street": "ул Автозаводская",
                "house": "5"
            }
        }
    ]
}
```

Поле **pid** — код отделения, выбранного клиентом — нужно передавать в сервис отправки заявок в поле **branchId**.

! Важно: не рекомендуется вызывать справочник **чаще одного раза в день**.

7. Описание ошибок

	Статус	Code	Title	Detail	Описание и сценарий	Сервис
1	401	NO_REQUIRED_HEADER	Unauthenticated	Header 'api-key' is required.	Не передан обязательный header "API-key"	Все
2	401	WRONG_PARAMETERS	Invalid api-key	The provided api-key is invalid	Передан невалидный header "API-key"	Все
3	401	WRONG_PARAMETERS	Expired api-key	The provided api-key has expired at 2019-03-25T15:46:45.756.	Передан header "API-key" с истекшим сроком действия. Рекомендация: проверить "API-key" и сбросить его при необходимости.	Все
4	400	validationError	Ошибка обработки данных	Поле \$.contactInfo[*].phoneNumber не может быть пустым	Не передан/передан пустой номер телефона Рекомендация: проверить наличие номера телефона	Создание заявки
5	400	validationError	Ошибка обработки данных	Значение XXXXXXXXXX поля \$.contactInfo[*].phoneNumber не соответствует регулярному выражению 7\\d{10}	Номер телефона передан в неподходящем формате Рекомендация: проверить соответствие номера телефона формату «11 цифр, первая 7»	Создание заявки
6	400	validationError	Ошибка обработки данных	Значение XXXXXXXXXX поля \$.contactInfo[0].extraPhoneNumbers [*] не соответствует регулярному выражению 7\\d{10}	Дополнительный номер телефона передан в неподходящем формате Рекомендация: проверить соответствие дополнительного номера телефона формату «11 цифр, первая 7»	Создание заявки
6	400	validationError	Ошибка обработки данных	Поле \$.productInfo[*].productCode не может быть пустым	Не передан/передан пустой номер код продукта Рекомендация: проверить наличие кода продукта	Создание заявки
7	400	validationError	Ошибка обработки данных	Значение 1234567890 поля \$.organizationInfo.inn не прошло проверку по контрольному числу	Передан некорректный ИНН Рекомендация: проверить правильность указания ИНН	Создание заявки
8	400	validationError	Ошибка обработки данных	Значение XXX поля \$.organizationInfo.kpp не соответствует регулярному выражению \\d{9}	Передан некорректный КПП Рекомендация: проверить правильность указания КПП	Создание заявки

	Статус	Code	Title	Detail	Описание и сценарий	Сервис
9	400	validationError	Ошибка обработки данных	Значение XXX поля \$.contactInfo[*].contactEmail не соответствует регулярному выражению ^[a-zA-Z0-9_.+-]+@[a-zA-Z0-9-]+\.[a-zA-Z0-9-.]+\\$" } }}	Передан некорректный email Рекомендация: проверить правильность указания email	
10	500	Repeated request error	Request was sent less than 15 minutes ago	Null	Технический дубликат. Нужно проверить на дубликаты и при необходимости переслать запрос.	Создание заявки
11	403	UNACCEPTABLE_COMPANY	Unacceptable company	Unfortunately, the Bank cannot accept the company with the provided TIN [1234567890]	Передан ИНН компании, которую Банк не может принять на обслуживание. Рекомендация: предварительно проверять ИНН	Создание заявки; Проверка по ИНН
12	403	BLOCKED_PARTNER	Forbidden	This partner is blocked.	Партнер заблокирован. Рекомендация: обратиться на Partners_Mail@alfabank.ru для уточнения причин блокировки	Создание заявки; Проверка по ИНН
13	403	LEADS_LIMIT_ERROR	Leads limit exceeded	The allowed limit of 30 leads has already exceeded.	Достигнут лимит на количество заявок в день. Рекомендация: повторить на следующий день	Создание заявки
14	403	CHECKS_LIMIT_ERROR	Checks limit exceeded	The allowed limit of 300 checks has already exceeded.	Достигнут лимит на количество проверок в день. Рекомендация: повторить на следующий день	Создание заявки; Проверка по ИНН
19	403	NEW_COMPANY_ERROR	No dadata info	Unfortunately, the Bank cannot accept the company with the provided TIN this way.	В некоторых случаях мы не можем принять контакт недавно зарегистрированных ЮЛ. Рекомендация: отправить заявку через полное заведение в интерфейсе ЛК Партнёра.	Создание заявки
15	500	UNEXPECTED_ERROR	Internal Server Error	Some additional information	Внутренняя ошибка сервера Рекомендация: 5 попыток отправки с интервалом 1 минута.	Все
16	503	SERVICE_UNAVAILABLE	Service Unavailable	Service is temporary unavailable. Please, try call later	Сервис недоступен Рекомендация: 5 попыток отправки с интервалом 1 минута	Все
17	400	CLIENT_ERROR	Create lead error	There was no successful scoring check for this TIN	В заявке отправляется кредитный продукт без предварительного прохождения кредитного скринга	Создание заявки

Статус	Code	Title	Detail	Описание и сценарий	Сервис
				Рекомендация: перед отправкой заявки необходимо проводить кредитный скоринг (раздел 5)	
18	400	CLIENT_ERROR	Create lead error	It is impossible to create an lead for a credit product without a TIN В заявке без ИНН отправляется кредитный продукт. Рекомендация: указывать ИНН и проводить кредитный скоринг (раздел 5)	Создание заявки
19	429	RATE_LIMIT_EXCEEDED	Rate limit exceeded	Too many requests, please try again later Общее количество запросов на кредитный скоринг превышено Рекомендация: попробуйте повторить отправку позднее	Проверка клиента по кредитному скорингу

Пример ответа с ошибкой:

```
HTTP/1.1 401 WRONG_PARAMETERS
Content-Type:application/json;charset=UTF-8
{
  "errors": [
    {
      "status": 401,
      "code": "WRONG_PARAMETERS",
      "title": "Invalid api-key",
      "detail": "The provided api-key is invalid."
    }
  ]
}
```

8. Справочники

8.1. Справочник продуктов (productCode)

	productCode	Описание	Зависимости
1.	LP_RKO	РКО	-
2.	LP_ACQ_TR	Эквайринг (Торговый)	Только с РКО
3.	LP_AKASSA	Альфа-касса	Только с РКО
4.	LP_LOAN_BUS	Бизнес кредит - кредит на ЮЛ/ИП	Только с РКО
5.	LP_OVER_ADV	Овердрафт для новых клиентов банка	Только с РКО
6.	LP_ACQ_E	Эквайринг (Интернет)	Только с РКО
7.	LP_PLEDGE	Бизнес Кредит с залогом	По запросу
8.	LP_VKL	Возобновляемая кредитная линия	По запросу
9.	LP_OVER_ADV	Авансовый овердрафт	По запросу
10.	LP_POPSHOP	Залоговый Лайт	По запросу

8.2. Справочник городов (cityCode)

Справочник городов ФИАС. В данном разделе он приведен для примера для основных городов присутствия Альфа-Банка. Его можно получить через сервис получения справочников (см. [Сервис получения справочников](#)) или как справочник ФИАС на сайте <https://fias.nalog.ru/>

cityCode	city
82b6b7c8-82a4-44b2-8bc7-691373706b89	Ангарск
06814fb6-0dc3-4bec-ba20-11f894a0faf5	Архангельск
d13945a8-7017-46ab-b1e6-edc1e89317ad	Барнаул
02e9c019-ab4d-4fa0-928e-d6c0a41dc256	Белгород
fd0191f4-4687-42d9-be64-4ebc7824bd08	Бердск
653b316e-d0ca-414e-9211-03a96643e91b	Бузулук
7b6de6a5-86d0-4735-b11a-499081111af8	Владивосток
a52b7389-0cfe-46fb-ae15-298652a64cf8	Волгоград
bc5ed788-84c8-493e-9598-7a15a9f1e4c1	Волжский
5bf5ddff-6353-4a3d-80c4-6fb27f00c6c1	Воронеж
2763c110-cb8b-416a-9dac-ad28a55b4402	Екатеринбург
deb1d05a-71ce-40d1-b726-6ba85d70d58f	Ижевск
98da33d3-9455-497a-9703-a0f89c95a901	Иннополис
8eeded222-72e7-47c3-ab3a-9a553c31cf72	Иркутск
93b3df57-4c89-44df-ac42-96f05e9cd3b9	Казань
df679694-d505-4dd3-b514-4ba48c8a97d8	Калининград
b502ae45-897e-4b6f-9776-6ff49740b537	Калуга
ecc16584-9814-44d7-a0f3-c79601caa734	Каменск-Уральский

cityCode	city
94bb19a3-c1fa-410b-8651-ac1bf7c050cd	Кемерово
7dfa745e-aa19-4688-b121-b655c11e482f	Краснодар
9b968c73-f4d4-4012-8da8-3dacd4d4c1bd	Красноярск
3bbda77d-ba3f-4457-9d44-c440815cda89	Курган
eacb5f15-1a2e-432e-904a-ca56bd635f1b	Липецк
988157bf-d6d5-4c2a-80ec-4ad531eea056	Магнитогорск
0c5b2444-70a0-4932-980c-b4dc0d3f02b5	Москва
b7127184-ead6-422b-b7cf-4fc7725b47a5	Мурманск
0bf0f4ed-13f8-446e-82f6-325498808076	Нижневартовск
b28b6f6f-1435-444e-95a6-68c499b0d27a	Новокузнецк
8dea00e3-9aab-4d8e-887c-ef2aaa546456	Новосибирск
5cb431ac-6645-47b2-9716-5fcc9ec1d7f	Новотроицк
140e31da-27bf-4519-9ea0-6185d681d44e	Омск
dce97bff-deb2-4fd9-9aec-4f4327bbf89b	Оренбург
71e7cb45-f365-4e6b-b068-a7d35349c3ba	Орск
ff3292b1-a1d2-47d4-b35b-ac06b50555cc	Пенза
45bad628-61b7-4445-b62a-fa4c7bc71ccd	Первоуральск
a309e4ce-2f36-4106-b1ca-53e0f48a6d95	Пермь
ccc34487-8fd4-4e71-b032-f4e6c82fb354	Петрозаводск
2858811e-448a-482e-9863-e03bf06bb5d4	Псков
c1cfe4b9-f7c2-423c-abfa-6ed1c05a15c5	Ростов-на-Дону
86e5bae4-ef58-4031-b34f-5e9ff914cd55	Рязань
bb035cc3-1dc2-4627-9d25-a1bf2d4b936b	Самара
c2deb16a-0330-4f05-821f-1d09c93331e6	Санкт-Петербург
bf465fda-7834-47d5-986b-ccdb584a85a6	Саратов
79da737a-603b-4c19-9b54-9114c96fb912	Сочи
2a1c7bdb-05ea-492f-9e1c-b3999f79dcbc	Ставрополь
84e0b23d-82fe-40a8-8739-55e679780dc3	Стерлитамак
f1eb1809-47d4-4f0b-9a74-fa416e9d3df2	Сургут
c52ea942-555e-45c6-9751-58897717b02f	Тверь
242e87c1-584d-4360-8c4c-aae2fe90048e	Тольятти
e3b0eae8-a4ce-4779-ae04-5c0797de66be	Томск
b2601b18-6da2-4789-9fbe-800dde06a2bb	Тула
9ae64229-9f7b-4149-b27a-d1f6ec74b5ce	Тюмень
bebfd75d-a0da-4bf9-8307-2e2c85eac463	Ульяновск
7339e834-2cb4-4734-a4c7-1fcac2c66e562	Уфа
a4859da8-9977-4b62-8436-4e1b98c5d13f	Хабаровск
dd8caeab-c685-4f2a-bf5f-550aca1bbc48	Чебоксары
a376e68d-724a-4472-be7c-891bdb09ae32	Челябинск
c58d0505-54eb-4c34-8216-b14f7cdb0ecb	Энгельс
44388add0-06aa-49b0-bbf9-1704629d1d68	Южно-Сахалинск
6b1bab7d-ee45-4168-a2a6-4ce2880d90d3	Ярославль

8.3. Справочник регионов (regionCode)

Его можно получить через сервис получения справочников или как справочник ФИАС на сайте <https://fias.nalog.ru/>

regionCode	Регион
0c5b2444-70a0-4932-980c-b4dc0d3f02b5	Москва г
c2deb16a-0330-4f05-821f-1d09c93331e6	Санкт-Петербург г
29251dcf-00a1-4e34-98d4-5c47484a36d4	Московская обл
ee594d5e-30a9-40dc-b9f2-0add1be44ba1	Курская обл
d02f30fc-83bf-4c0f-ac2b-5729a866a207	Камчатский край
d00e1013-16bd-4c09-b3d5-3cb09fc54bd8	Краснодарский край
1b507b09-48c9-434f-bf6f-65066211c73e	Еврейская Аобл
54049357-326d-4b8f-b224-3c6dc25d6dd3	Тюменская обл
248d8071-06e1-425e-a1cf-d1ff4c4a14a8	Карелия Респ
327a060b-878c-4fb4-8dc4-d5595871a3d8	Ставропольский край
826fa834-3ee8-404f-bdbc-13a5221cfb6e	Ямало-Ненецкий АО
639efe9d-3fc8-4438-8e70-ec4f2321f2a7	Белгородская обл
d66e5325-3a25-4d29-ba86-4ca351d9704b	Ханты-Мансийский Автономный округ - Югра АО
43909681-d6e1-432d-b61f-ddac393cb5da	Приморский край
5e465691-de23-4c4e-9f46-f35a125b5970	Орловская обл
f136159b-404a-4f1f-8d8d-d169e1374d5c	Чукотский АО
61b95807-388a-4cb1-9bee-889f7cf811c8	Карачаево-Черкесская Респ
de459e9c-2933-4923-83d1-9c64cf7a817	Северная Осетия - Алания Респ
18133adf-90c2-438e-88c4-62c41656de70	Калужская обл
889b1f3a-98aa-40fc-9d3d-0f41192758ab	Томская обл
878fc621-3708-46c7-a97f-5a13a4176b3e	Чувашская Республика - Чувашия
9c05e812-8679-4710-b8cb-5e8bd43cdf48	Магаданская обл
393aeccb-89ef-4a7e-ae42-08d5cebc2e30	Кемеровская обл
6466c988-7ce3-45e5-8b97-90ae16cb1249	Иркутская обл
e8502180-6d08-431b-83ea-c7038f0df905	Смоленская обл
844a80d6-5e31-4017-b422-4d9c01e9942c	Амурская обл
fee76045-fe22-43a4-ad58-ad99e903bd58	Ульяновская обл
c20180d9-ad9c-46d1-9eff-d60bc424592a	Коми Респ
491cde9d-9d76-4591-ab46-ea93c079e686	Калмыкия Респ
c99e7924-0428-4107-a302-4fd7c0cca3ff	Пензенская обл
92b30014-4d52-4e2e-892d-928142b924bf	Свердловская обл
f5807226-8be0-4ea8-91fc-39d053aec1e2	Брянская обл
0c089b04-099e-4e0e-955a-6bf1ce525f1a	Татарстан Респ
a84b2ef4-db03-474b-b552-6229e801ae9b	Ярославская обл
8bcec9d6-05bc-4e53-b45c-ba0c6f3a5c44	Оренбургская обл
90c7181e-724f-41b3-b6c6-bd3ec7ae3f30	Калининградская обл
4a3d970f-520e-46b9-b16c-50d4ca7535a8	Курганская обл
d028ec4f-f6da-4843-ada6-b68b3e0efa3d	Тульская обл

regionCode	Регион
4f8b1a21-e4bb-422f-9087-d3cbf4bebc14	Пермский край
63ed1a35-4be6-4564-a1ec-0c51f7383314	Байконур г
0b940b96-103f-4248-850c-26b6c7296728	Кировская обл
294277aa-e25d-428c-95ad-46719c4ddb44	Архангельская обл
df594e0e-a935-4664-9d26-0bae13f904fe	Саратовская обл
1c727518-c96a-4f34-9ae6-fd510da3be03	Мурманская обл
db9c4f8b-b706-40e2-b2b4-d31b98dc3d1	Красноярский край
1490490e-49c5-421c-9572-5673ba5d80c8	Липецкая обл
a9a71961-9363-44ba-91b5-ddf0463aebc2	Тамбовская обл
0bb7fa19-736d-49cf-ad0e-9774c4dae09b	Дагестан Респ
15784a67-8cea-425b-834a-6afe0e3ed61c	Костромская обл
52618b9c-bcbb-47e7-8957-95c63f0b17cc	Удмуртская Респ
c225d3db-1db6-4063-ace0-b3fe9ea3805f	Саха /Якутия/ Респ
8d3f1d35-f0f4-41b5-b5b7-e7cadf3e7bd7	Хакасия Респ
1ac46b49-3209-4814-b7bf-a509ea1aec9	Новосибирская обл
61723327-1c20-42fe-8dfa-402638d9b396	Тверская обл
e5a84b81-8ea1-49e3-b3c4-0528651be129	Новгородская обл
de67dc49-b9ba-48a3-a4cc-c2ebfec46c5e	Чеченская Респ
d8327a56-80de-4df2-815c-4f6ab1224c50	Адыгея Респ
6fdec8-893a-4e3f-a5ba-aa062459463b	Севастополь г
bd8e6511-e4b9-4841-90de-6bbc231a789e	Крым Респ
ed36085a-b2f5-454f-b9a9-1c9a678ee618	Вологодская обл
6f2cbfd8-692a-4ee4-9b16-067210bde3fc	Башкортостан Респ
963073ee-4dfc-48bd-9a70-d2dfc6bd1f31	Рязанская обл
6d1ebb35-70c6-4129-bd55-da3969658f5d	Ленинградская обл
de2cbfdf-9662-44a4-a4a4-8ad237ae4a3e	Марий Эл Респ
b6ba5716-eb48-401b-8443-b197c9578734	Забайкальский край
7d468b39-1afa-41ec-8c4f-97a8603cb3d4	Хабаровский край
83009239-25cb-4561-af8e-7ee111b1cb73	Астраханская обл
b8837188-39ee-4ff9-bc91-fcc9ed451bb3	Владимирская обл
88cd27e2-6a8a-4421-9718-719a28a0a088	Нижегородская обл
f6e148a1-c9d0-4141-a608-93e3bd95e6c4	Псковская обл
0824434f-4098-4467-af72-d4f702fed335	Ивановская обл
05426864-466d-41a3-82c4-11e61cdc98ce	Омская обл
37a0c60a-9240-48b5-a87f-0d8c86cdb6e1	Мордовия Респ
da051ec8-da2e-4a66-b542-473b8d221ab4	Волгоградская обл
5c48611f-5de6-4771-9695-7e36a4e7529d	Алтай Респ
b2d8cd20-cabc-4deb-afad-f3c4b4d55821	Ингушетия Респ
026bc56f-3731-48e9-8245-655331f596c0	Тыва Респ
1781f74e-be4a-4697-9c6b-493057c94818	Кабардино-Балкарская Респ
89db3198-6803-4106-9463-cbf781eff0b8	Ненецкий АО
aea6280f-4648-460f-b8be-c2bc18923191	Сахалинская обл
27eb7c10-a234-44da-a59c-8b1f864966de	Челябинская обл

regionCode	Регион
8276c6a1-1a86-4f0d-8920-aba34d4cc34a	Алтайский край
b756fe6b-bbd3-44d5-9302-5bfcc740f46e	Воронежская обл
f10763dc-63e3-48db-83e1-9c566fe3092b	Ростовская обл
df3d7359-afa9-4aaa-8ff9-197e73906b1c	Самарская обл
a84ebed3-153d-4ba9-8532-8bdf879e1f5a	Бурятия Респ