

# API Эльбы

В данный момент API поддерживает две операции: создание счетов в Эльбе и отслеживания их статуса.

Соответственно, работа с API осуществляется HTTP запросами к следующим адресам:

- `https://elba.kontur.ru/API/CreateBill.ashx` — создание счета.
- `https://elba.kontur.ru/API/GetChanges.ashx` — отслеживание статусов счетов.

Формат взаимодействия и авторизации будет описан далее.

Счета, созданные с помощью API будут отображаться в интерфейсе Эльбы без каких-либо особенностей, и взаимодействие с ними остаётся стандартным. Пользователь может выполнять над ними все обычные операции, в том числе — удаление.

Отслеживать изменение статусов можно только у счетов, созданных через API. Изменения счетов, созданных вручную через интерфейс не будут присутствовать в ответе API.

Условия для работы с API:

- Учетная запись пользователя должна быть оплачена и иметь тариф, позволяющий создавать счета;
- Абонент не должен являться обслуживаемой бухгалтерией;
- SMS-авторизация должна быть выключена.

В случае ошибок, не связанных с авторизацией, ответ сервера будет иметь статус 400 `Bad Request`, а тело — содержать описание ошибки и её номер, с которым можно обратиться к нам для выяснения деталей.

---

## 1. Авторизация

При любом запросе необходимо передавать в заголовках HTTP запроса логин и пароль пользователя в Эльбе.

Имена заголовков: `X-Login` и `X-Password`, соответственно.

Значения заголовков должны быть URL Encoded. Средства для URL-кодирования строк предоставляют большинство популярных языков программирования.

Если передана неверная пара логин/пароль, то статус ответа сервера будет 403 `Forbidden`. Кроме того, ошибка с этим кодом возникнет если учетная запись является обслуживаемой бухгалтерией или включена SMS-авторизация.

---

## 2. Создания счета

Для создания счета необходимо выполнить POST запрос на адрес <https://elba.kontur.ru/API/CreateBill.ashx>.

В теле запроса передается контракт в формате JSON, описывающий новый счет в Эльбе.

Если счет будет успешно создан, то будет получен ответ сервера со статусом 200 OK, а в теле ответа будет **ID** (GUID) созданного счета.

Полная структура контракта:

```
{
  "Number": (string),
  "Date": (string),
  "WithNds": (bool),
  "SumsWithNds": (bool),
  "Comment": (string),
  "BankAccount": {
    "AccountNumber": (string)
    "Bank": {
      "Name": (string),
      "Bik": (string),
      "City": (string),
      "CityType": (string),
      "CorrAccount": (string)
    }
  },
  "Contractor": {
    "Name": (string),
    "Inn": (string),
    "Kpp": (string)
  },
  "Items": [
    {
      "ProductName": (string),
      "UnitName": (string),
      "Quantity": (decimal),
      "Price": (decimal),
      "PriceWithoutNds": (decimal),
      "Sum": (string),
      "NdsRate": (int)
    },
    ...
  ]
}
```

Описание полей:

Параметр	Описание	Требуется
<b>Счет</b>		
Number	Номер счета.	Да
Date	Дата в формате ISODate.	Да
WithNds	Флаг. Является ли счет новый счет счетом с НДС. Аналог флага "Выставить счет с НДС" в интерфейсе.	Да
SumsWithNds	Флаг. Включает ли цена и сумма строк фактурной части НДС. Аналог флага "Вводить цену с учетом НДС" в интерфейсе. Актуально только если счет выставлен с НДС, иначе - принимает значение false.	Да
Comment	Комментарий.	Нет
BankAccount	Расчетный счет организации.	Нет
Contractor	Контрагент, которому выставлен счет.	Нет
Items	Массив. Строки фактурной части счета.	Нет
<b>BankAccount</b>		
AccountNumber	Номер расчетного счета	Да
Bank	См. раздел "сопоставление и создание расчетных счетов"	Нет
<b>Contractor</b>		
Name	Имя контрагента.	Нет *
Inn	Иинн контрагента.	Нет *
Kpp	Кпп контрагента.	Нет
* Примечание: см. раздел "сопоставление и создание контрагентов"		
<b>Item</b>		
ProductName	Наименование товара или услуги. Длина не превышает 1000 символов.	Нет
UnitName	Единица измерения. Длина не превышает 1000 символов.	Нет
Quantity	Количество.	Да
Price	Цена за единицу. Если счет выставлен с НДС, то данное значение включает сумму НДС.	Да
PriceWithoutNds	Цена за единицу без учета НДС. Если счет выставлен без НДС, значение в точности равно Price.	Да
Sum	Сумма по текущей строке (с учетом НДС). В идеале должно быть равно Price * Quantity :)	Да
NdsRate	Ставка НДС. Целое число от 0 до 3. Без НДС = 0, НДС 0% = 1, НДС 10% = 2, НДС 18% = 3	Да

## Пример запроса:

```
POST
  https://elba.kontur.ru/API/CreateBill.ashx
Headers:
  X-Login: urlencoded(login)
  X-Password: urlencoded(password)
Body:
{
  "Number": "а6324",
  "Date": "2014-02-07T00:00:00.000Z",
  "WithNds": true,
  "SumsWithNds": true
  "Comment": "test comment",
  "BankAccount": {
    "AccountNumber": "40802810363030003088",
    "Bank": {
      "Name": "ОАО \"АЛЬФА-БАНК\"",
      "Bik": "044525593",
      "City": "МОСКВА",
      "CityType": "Г.",
      "CorrAccount": "30101810200000000593"
    }
  },
  "Contractor": {
    "Name": "ИП Иванов Иван Иванович",
    "Inn": "6660013279",
    "Kpp": "666100042"
  },
  "Items": [
    {
      "ProductName": "стул",
      "UnitName": "шт",
      "Quantity": 10.0,
      "Price": 123.06,
      "PriceWithoutNds": 104.29,
      "Sum": 1230.6,
      "NdsRate": 3
    },
    {
      "ProductName": "кабель",
      "UnitName": "м",
      "Quantity": 10.0,
      "Price": 100.0,
      "PriceWithoutNds": 100.0,
      "Sum": 1000.0,
      "NdsRate": 0
    }
  ]
}
```

```
        "ProductName": "апельсины",
        "UnitName": "кг",
        "Quantity": 10.0,
        "Price": 189.56,
        "PriceWithoutNds": 160.64,
        "Sum": 1895.60,
        "NdsRate": 2
    }
]
}
```

### Сопоставление и создание расчетных счетов

Если при создании счета указан **BankAccount**, то действует следующая логика: Созданный счет будет связан с существующим расчетным счетом организации, если таковой будет найден по номеру (**AccountNumber**). В таком случае поле **Bank** можно не передавать. Это наиболее частая ситуация.

В случае, если существующего расчетного счета с указанным номером нет, то будет создан новый. В этом случае необходимо передать поле **Bank**. Полная структура контракта:

```
{
    "Bik": (string),
    "City": (string),
    "CityType": (string),
    "CorrAccount": (string),
    "Name": (string)
}
```

Однако, если банк является действующим на данный момент, то достаточно указать только поле **Bik** - БИК банка, остальная информация будет подгружена из наших справочников.

### Сопоставление и создание контрагентов

Аналогичная логика действует в случае если указан **Contractor** - клиент, которому выставлен счет.

Сопоставление с существующими контрагентами происходит по имени или инн/кпп, соответственно должно быть указано как минимум одно из этих полей. При поиске по имени регистр не имеет значения.

Если существующий контрагент не найден, то будет создан новый (имя должно быть не пусто).

---

### 3. Отслеживание актуальных статуса счетов

Для получения измененных статусов счетов надо выполнить GET запрос на адрес `https://elba.kontur.ru/API/GetChanges.ashx?fromRevision=[revision]`.

Каждый счет имеет специальный параметр — "ревизию". При каждом изменении счета ревизия увеличивается на некоторую величину. Номера ревизий уникальны в рамках системы.

При обращении к API необходимо указать в параметрах запроса начальную ревизию (имя параметра **fromRevision**), начиная с которой необходимо получить актуальные статусы счетов.

Такая схема позволяет получать только те счета, которые изменились с момента предыдущей проверки.

Ответ сервера содержит контракт в формате JSON, описывающий статусы счетов, созданных ранее через API.

Структура контракта:

```
[
  {
    "Id": (string),
    "Revision": (long),
    "Status": (int)
  }
]
```

Описание полей:

Параметр	Описание
Id	ID (GUID) счета, полученный ранее при создании счета через API.
Revision	Ревизия текущего состояния счета.
Status	Статус счета. Целое число от 0 до 3: 0 - Не оплачен 1 - Оплачен 2 - Частично оплачен 3 - Отклонён

Таким образом, при первом обращении к API в качестве начальной ревизии можно указать 0, а при последующих обращениях — максимальное значение ревизии из последнего полученного ответа. Это позволит при каждом запросе получать информацию только по тем счетам, которые изменились со времени последнего обращения.

Следует заметить, что счет может попасть в список измененных не только вследствие изменения статуса, но и вследствие любых других операций редактирования через интерфейс Эльбы, например — изменения номера.

Если всегда передавать в качестве начальной ревизии 0, то в результате каждого запроса будут получены актуальные статусы всех счетов, созданным ранее через API. Это тоже является корректным сценарием работы.

### Пример запроса:

GET

`https://elba.kontur.ru/API/GetChanges.ashx?fromRevision=42`

Headers:

X-Login: `urlencoded(login)`

X-Password: `urlencoded(password)`

### Пример ответа:

body:

```
[
  {
    "Id": "7b9c0329-c640-4c4e-868a-900df8f6e5c5",
    "Revision": 123,
    "Status": 1
  },
  {
    "Id": "53df594a-eed4-44ba-bde1-f1d26b42c3b3",
    "Revision": 89,
    "Status": 0
  }
]
```