

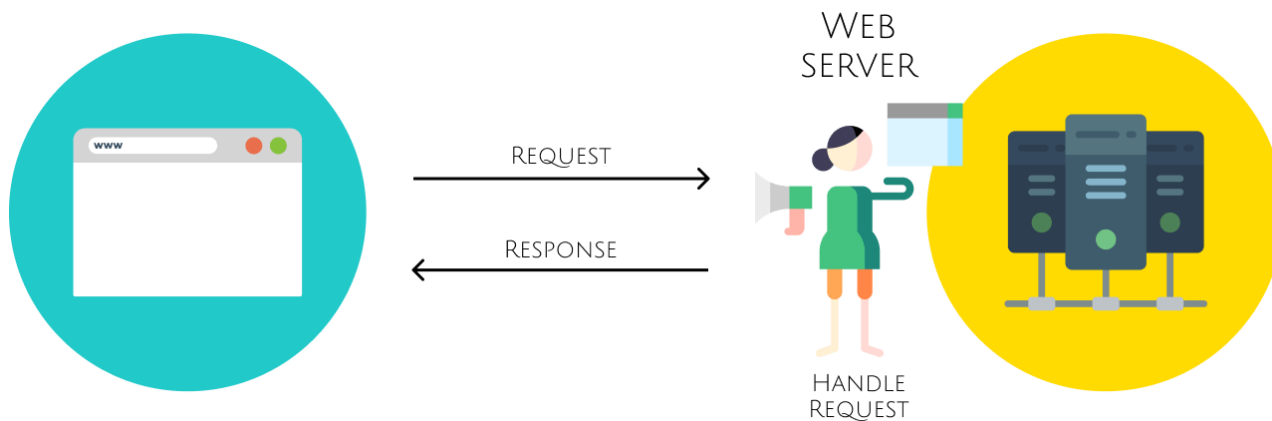
#T62

# Интеграция и обмен данными. Протокол HTTP. Интерфейс OData в 1С

Программист 1С: с нуля до профи



# Протокол HTTP (Теория)



# Протокол HTTP

- **Hypertext Transfer Protocol** – протокол передачи гипертекста.
- **Гипертекст** – это тип разметки, который добавляется в текстовые документы, который позволяет определить способ отображения текста
  - Пример: HTML


# Использование гипертекста в HTML

В текстовом редакторе

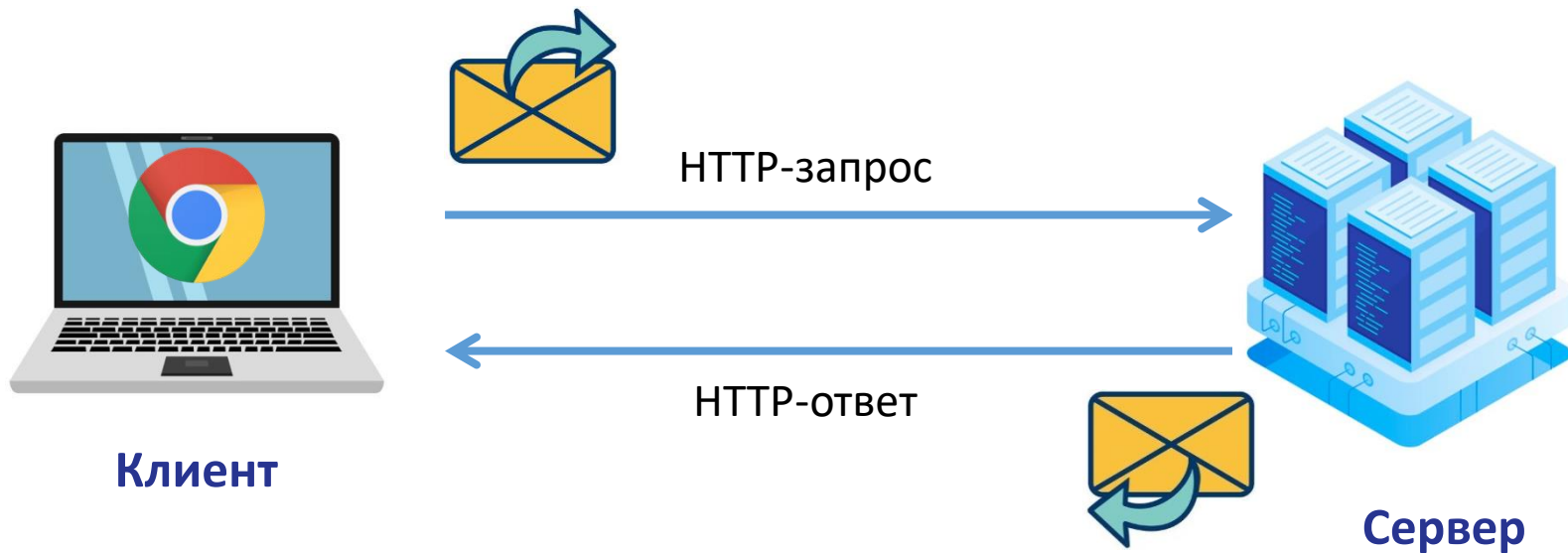


В браузере

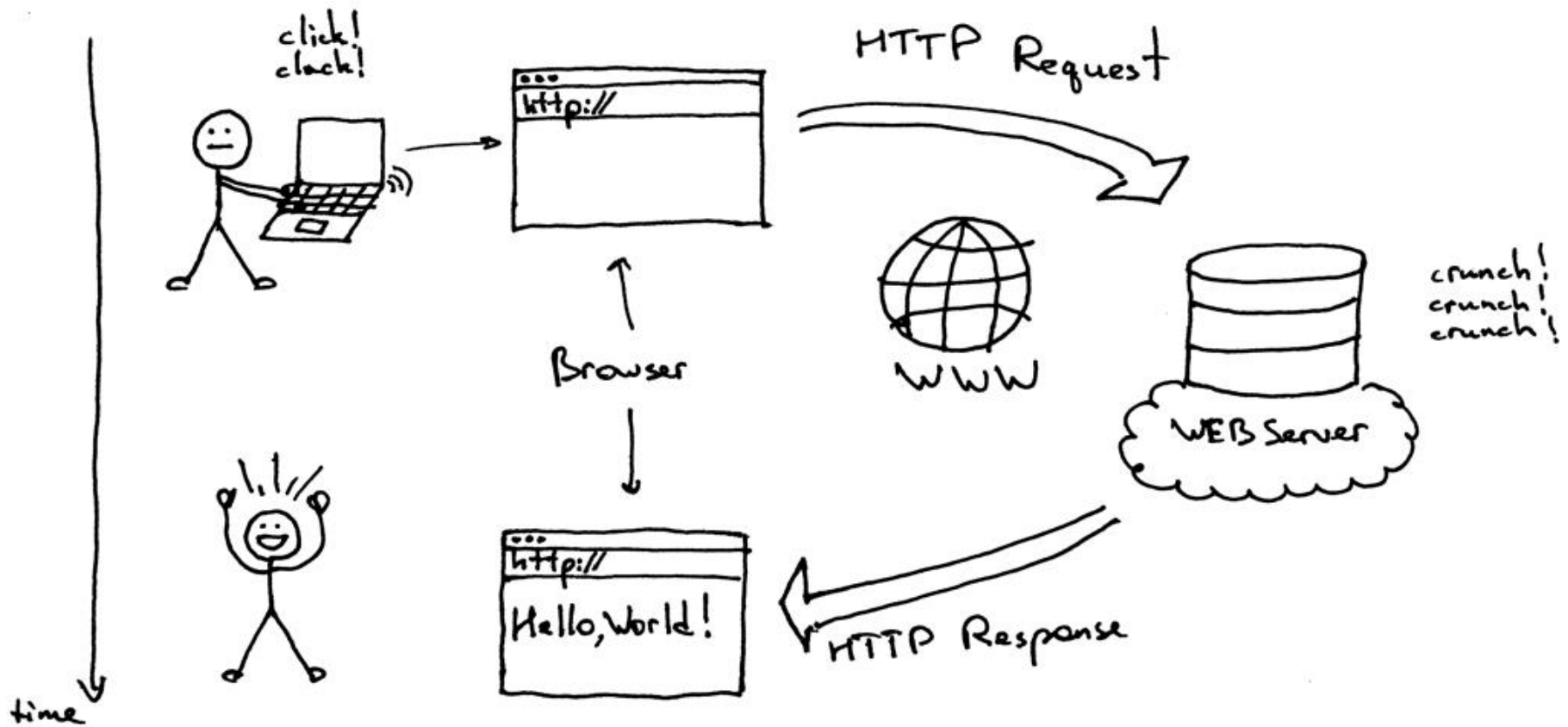


<pre>&lt;html&gt; &lt;body&gt;  &lt;h1&gt;Пример HTML документа&lt;/h1&gt; &lt;h2&gt;Простой документ&lt;/h2&gt;  &lt;p&gt;Мой первый параграф.&lt;/p&gt;  Простой список: &lt;ul&gt; &lt;li&gt; 1-й элемент &lt;/li&gt; &lt;li&gt; 2-й элемент &lt;/li&gt; &lt;li&gt; 2-й элемент &lt;/li&gt; &lt;/ul&gt;  &lt;img src="web.jpg" width="343" height="214" /&gt;  &lt;/body&gt; &lt;/html&gt;</pre>	<p><b>Пример HTML документа</b></p> <p><b>Простой документ</b></p> <p>Мой первый параграф.</p> <p>Простой список:</p> <ul style="list-style-type: none"><li>• 1-й элемент</li><li>• 2-й элемент</li><li>• 2-й элемент</li></ul> 
---	---

# Как это работает?



# Как это работает?

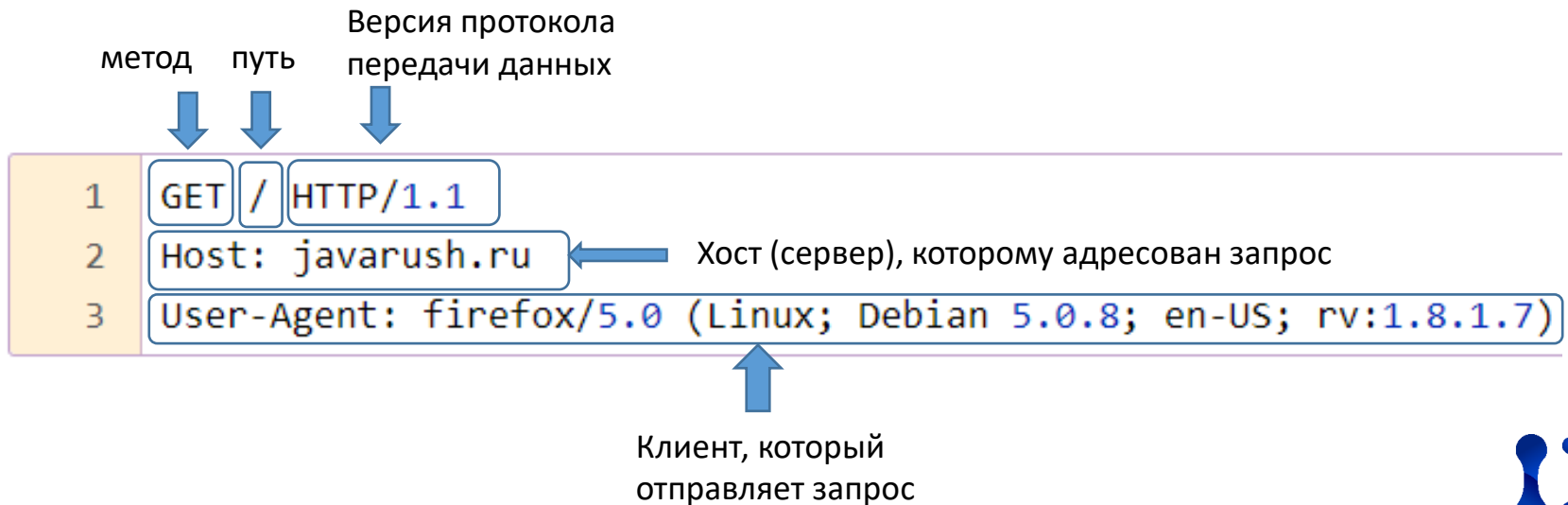


# Что такое веб-сервер?

- **Веб-сервер** – это программа, которая предназначена для следующих целей:
  - **принимать запросы** от других программ (клиентов) и обрабатывать их
  - **отправлять ответы** другим программам (клиентам)
- Клиентское приложение взаимодействует с веб-сервером по протоколу **HTTP**

# HTTP-запросы

- **HTTP-запрос** – это сообщение, которое клиент, отправляет в сторону сервера.
- **Пример HTTP-запроса:**



# Что такое URL?

- **Uniform Resource Locator (URL)** – унифицированный указатель ресурса

- Простыми словами:

URL – уникальный адрес ресурса в интернете

- **Структура URL:**

<протокол>:[//[<логин>[:<пароль>]@]<хост>[:<порт>]][/<URL-путь>][?<параметры>][#<якорь>]

- **Пример:**

<https://ironskills.by/blog/tpost/http-v-1c>



протокол



доменное  
имя сервера  
(хост)



адрес  
ресурса

# Примеры URL

**Протокол**

**Хост**

**Адрес ресурса**

**Параметры**

`https://www.nbrb.by/api/exrates/rates?periodicity=0&ondate=2021-02-20`

**Протокол**

**Хост**

**Адрес ресурса**

`https://api.telegram.org/bot123456:ABC-DEF1234ghIkl-zyx57W2v1u123ew11/getMe`

**Протокол**

**Хост**

**Адрес ресурса**

`http://www.cbr.ru/DailyInfoWebServ/DailyInfo.aspx`

# Структура пакета HTTP

Пример HTTP-запроса



Запрос/статус ответа



**GET** /ironskills.by/blog/tpost/http-v-1c

Заголовки  
(необязательно)



*Host: ru.wikipedia.org*  
*User-Agent: Mozilla/5.0 (X11; U; Linux)*  
*Content-Type: text/html*  
*Connection: close*

Тело сообщения  
(необязательно)

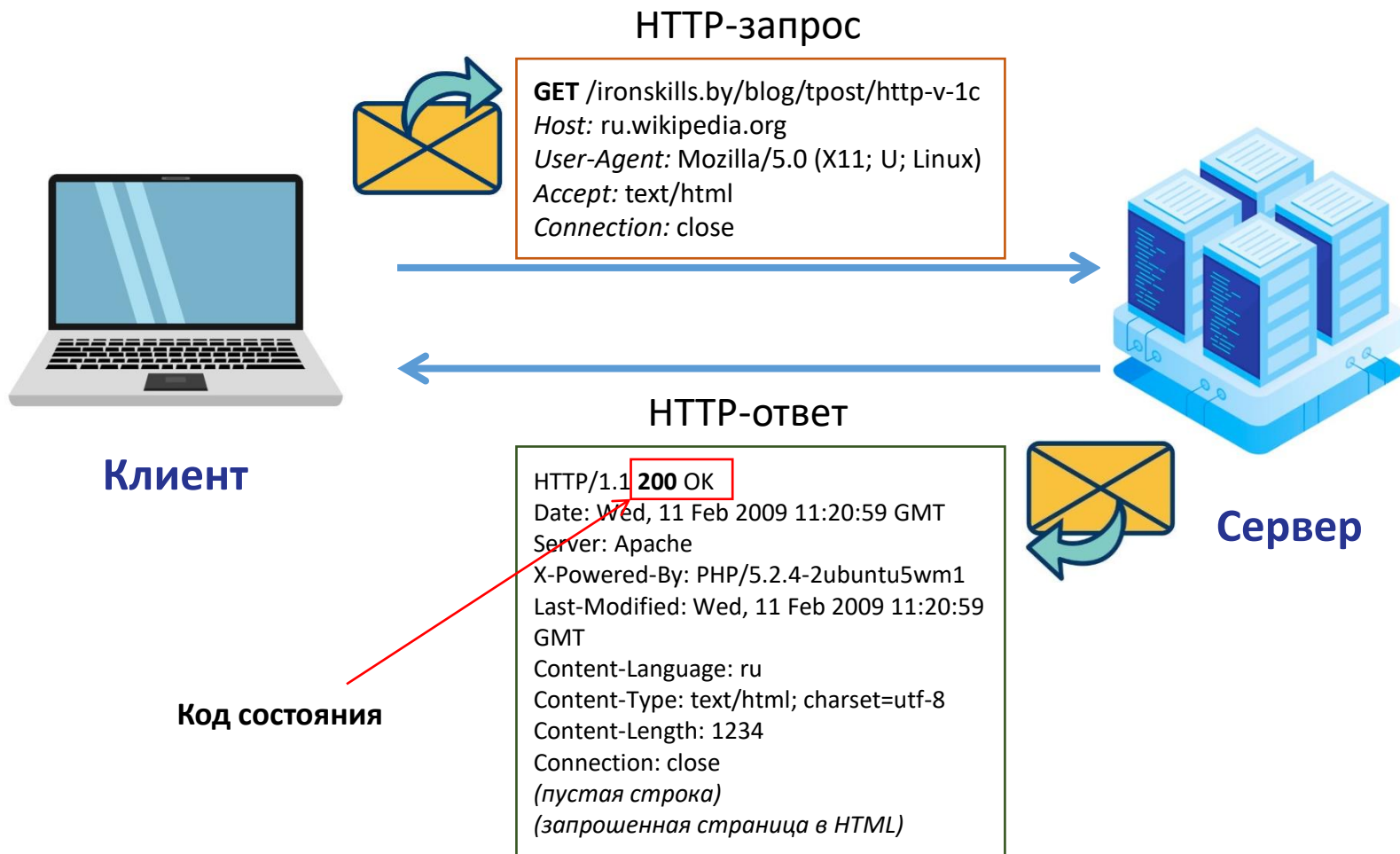


(отсутствует)

# Основные методы HTTP

- **GET** – получение ресурса
- **POST** – создание ресурса
- **PUT** – обновление ресурса
- **PATCH** – частичное обновление ресурса
- **DELETE** – удаление указанного ресурса

# Статусы обработки запроса



# Коды состояний HTTP

- **1XX** – информация
- **2XX** – успешное выполнение (200 OK)
- **3XX** – перенаправление
- **4XX** – ошибка на стороне клиента (403 – доступ запрещен, 404 – ресурс не найден)
- **5XX** – ошибка на стороне сервера (500 – внутренняя ошибка сервера)

# Коды состояний HTTP

Часто встречающиеся коды состояний HTTP:




- **200** – успешный запрос
- **301** – запрошенный ресурс перенесен
- **401** – для доступа к ресурсу требуется аутентификация
- **403** – доступ к запрошенному ресурсу запрещен
- **404** – запрошенный ресурс не найден
- **500** – внутренняя ошибка сервера

# Протокол HTTP (Практика)

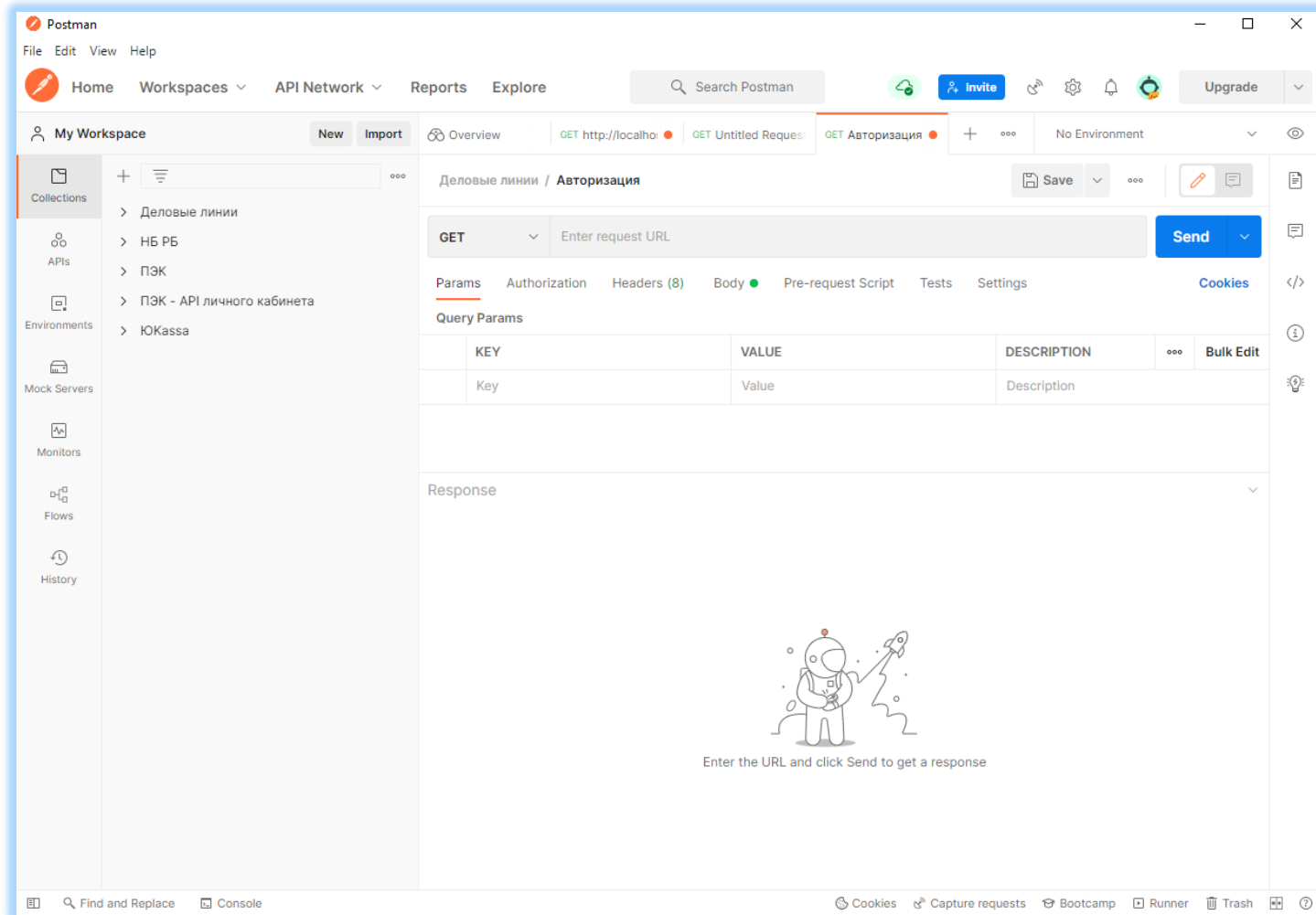


# HTTP-клиенты

Программы, с помощью которых можно выполнять HTTP-запросы и проверять HTTP-ответы:

- **Fiddler**   
<https://www.telerik.com/fiddler>
- **Postman API Client**   
POSTMAN  
<https://www.postman.com/product/api-client/>
- **cURL**   
<https://curl.se/>

# Интерфейс программы Postman



# Примеры задач (запросы из Fiddler)

## Задача № 1

- ~~Получить курсы валют~~

—Источник: <https://www.nbrb.by/>

## Задача № 2

- ~~Получить данные контрагента по УНП~~

• Источник: <http://www.portal.nalog.gov.by/grp/>

## Задача № 3

- ~~Получить склонения строки «Директор»~~

• Источник: <https://morpher.ru/ws3/>

# Инструменты для 1С

## Консоль HTTP-запросов:

← → Отправить HTTP Запрос

Получить код    Получить код по методу ▾    Еще ▾

Сервер:     Порт:      HTTPS     Использовать прокси

Логин:     Пароль:

[Настройки прокси](#)

POST    GET    PUT    HEAD    PATCH    DELETE

Сервис:

Запрос/Результат    Заголовки (Headers)    Параметры    Данные формы (form data)    Генератор JSON

Текст запроса:

```
{
  "name": "Иванов Иван Иванович",
  "age": 33,
  "id": "155897",
  "contact": {
    "phone": "8-800-2000-600",
    "mail": "test@mysite.com"
  }
}
```

Код ответа:

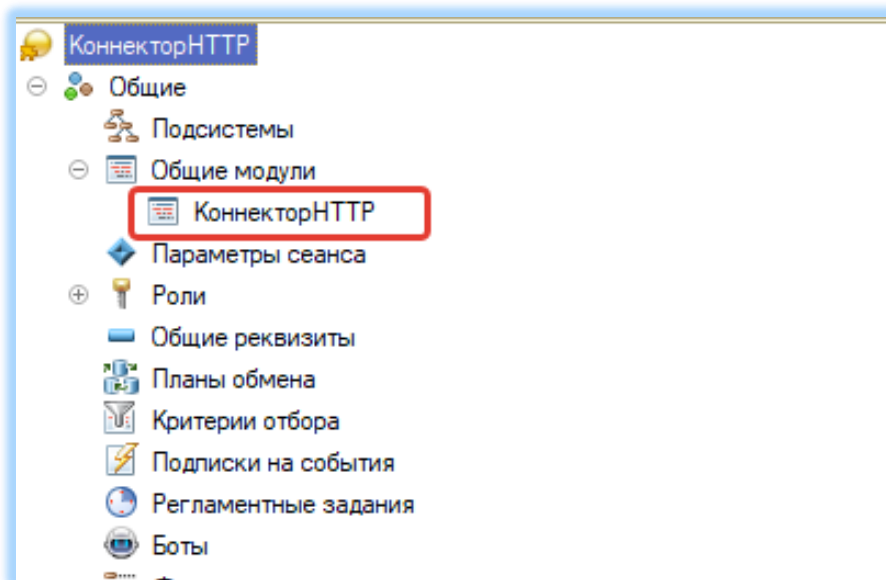
Результат запроса:

```
{
  "name": "Иванов Иван Иванович",
  "age": 33,
  "id": "155897",
  "contact": {
    "phone": "8-800-2000-600",
    "mail": "test@mysite.com"
  }
}
```

<https://infostart.ru/public/835540/>

# Инструменты для 1С

**КоннекторНТТР** - библиотека для работы с НТТР запросами



<https://infostart.ru/1c/articles/709325/>

# Примеры задач (запросы из 1С)

## Задача № 1

- ~~Получить курсы валют~~

—Источник: <https://www.nbrb.by/>

## Задача № 2

- ~~Получить данные контрагента по УНП~~

• Источник: <http://www.portal.nalog.gov.by/grp/>

## Задача № 3

- **Получить склонения строки «Директор»**

• Источник: <https://htmlweb.ru/service/sklonjator.php>

# Объекты встроенного языка для работы с HTTP-запросами

## HTTPСоединение (HTTPConnection)

### Свойства:

[ЗащищенноеСоединение \(SecureConnection\)](#)  
[ИспользоватьАутентификациюОС \(UseOSAuthentication\)](#)  
[Пароль \(Password\)](#)  
[Пользователь \(User\)](#)  
[Порт \(Port\)](#)  
[Прокси \(Proxy\)](#)  
[Сервер \(Host\)](#)  
[Таймаут \(Timeout\)](#)

### Методы:

[ВызватьHTTPМетод \(CallHTTPMethod\)](#)  
[Записать \(Put\)](#)  
[Изменить \(Patch\)](#)  
[ОтправитьДляОбработки \(Post\)](#)  
[Получить \(Get\)](#)  
[ПолучитьЗаголовки \(Head\)](#)  
[Удалить \(Delete\)](#)

### Конструкторы:

[По умолчанию](#)

### Описание:

Предназначен для взаимодействия с внешними системами по протоколу HTTP, в т.ч. для передачи файлов.  
Поддерживаются следующие протоколы: HTTP 1.0 и HTTP 1.1. Для защиты соединения может быть использован протокол TLS 1.2.

## HTTPЗапрос (HTTPRequest)

### Свойства:

[АдресРесурса \(ResourceAddress\)](#)  
[Заголовки \(Headers\)](#)

### Методы:

[ПолучитьИмяФайлаТела \(GetBodyFileName\)](#)  
[ПолучитьТелоКакДвоичныеДанные \(GetBodyAsBinary\)](#)  
[ПолучитьТелоКакПоток \(GetBodyAsStream\)](#)  
[ПолучитьТелоКакСтроку \(GetBodyAsString\)](#)  
[УстановитьИмяФайлаТела \(SetBodyFileName\)](#)  
[УстановитьТелоИзДвоичныхДанных \(SetBodyFromBinary\)](#)  
[УстановитьТелоИзСтроки \(SetBodyFromString\)](#)

### Конструкторы:

[По адресу ресурса и заголовкам](#)  
[Формирование неинициализированного объекта](#)

### Описание:

Предназначен для описания HTTP-запросов, отправляемых через объект [HTTPСоединение](#).  
Позволяет задать адрес запрашиваемого ресурса, заголовки и тело запроса.

### Доступность:

Тонкий клиент, мобильный клиент, сервер, толстый клиент, внешнее соединение, мобильное приложение (клиент), мобильное приложение (сервер).

## HTTPОтвет (HTTPResponse)

### Свойства:

[Заголовки \(Headers\)](#)  
[КодСостояния \(StatusCode\)](#)  
[Причина \(Reason\)](#)

### Методы:

[ПолучитьИмяФайлаТела \(GetBodyFileName\)](#)  
[ПолучитьТелоКакДвоичныеДанные \(GetBodyAsBinaryData\)](#)  
[ПолучитьТелоКакПоток \(GetBodyAsStream\)](#)  
[ПолучитьТелоКакСтроку \(GetBodyAsString\)](#)

### Описание:

Предоставляет доступ к содержимому ответа HTTP-сервера на запрос.

### Доступность:

Тонкий клиент, мобильный клиент, сервер, толстый клиент, внешнее соединение, мобильное приложение (клиент), мобильное приложение (сервер).

### Использование в версии:

Доступен, начиная с версии 8.2.

# Пример кода отправки HTTP-запроса из 1С

```
//Ниже описан код по выполнению следующего HTTP-запроса:  
//https://www.nbrb.by/api/exrates/rates?periodicity=0  
  
//Параметры соединения  
Сервер      = "www.nbrb.by";  
Порт       = 443;//для http - 80, для https - 443  
Пользователь = "";  
Пароль      = "";  
Прокси     = Неопределено;  
Таймаут    = 30;//время ожидания ответа в секундах  
  
ЗащищенноеСоединение = Новый ЗащищенноеСоединениеOpenSSL();//для https  
  
//Создание соединения  
Соединение = Новый HTTPСоединение(Сервер, Порт, Пользователь, Пароль, Прокси, Таймаут, ЗащищенноеСоединение);  
  
//Формирование HTTP-запроса  
АдресРесурса = "/api/exrates/rates?periodicity=0";  
HTTPЗапрос = Новый HTTPЗапрос(АдресРесурса);  
  
//Отправка HTTP-запроса на сервер (метод "GET")  
HTTPОтвет = Соединение.ВызватьHTTPМетод("GET", HTTPЗапрос);  
  
//Анализ кода состояния HTTP-ответа  
Если HTTPОтвет.КодСостояния = 200 Тогда  
  
    //Получение тела ответа  
    ТелоОтвета = HTTPОтвет.ПолучитьТелоКакСтроку();  
    ЗаголовкиОтвета = "Status Code: " + HTTPОтвет.КодСостояния + Символы.ПС;  
    Для каждого ЗначениеЗаголовка Из HTTPОтвет.Заголовки Цикл  
        ЗаголовкиОтвета = ЗаголовкиОтвета + Символы.ПС +  
            ЗначениеЗаголовка.Ключ + " : " + ЗначениеЗаголовка.Значение;  
    КонецЦикла;  
  
Иначе  
    Сообщение = Новый СообщениеПользователю;  
    Сообщение.Текст = "Не удалось выполнить запрос.  
        |Код состояния: " + HTTPОтвет.КодСостояния + "  
        |Тело ответа: " + HTTPОтвет.ПолучитьТелоКакСтроку();  
    Сообщение.Сообщить();  
КонецЕсли;
```

# Примеры задач (запросы из кода 1С)

## Задача № 1

- **Получить курсы валют**

Источник: <https://www.nbrb.by/>

## Задача № 2

- **Получить данные контрагента по УНП**

- Источник: <http://www.portal.nalog.gov.by/grp/>

## Задача № 3

- **Получить склонения строки «Директор»**

- Источник: <https://htmlweb.ru/service/sklonjator.php>

# Стандартный интерфейс OData



# Стандартный интерфейс OData

## Предназначение:

- Получение доступа к данным информационной базы 1С из внешнего приложения без модификация кода прикладного решения.

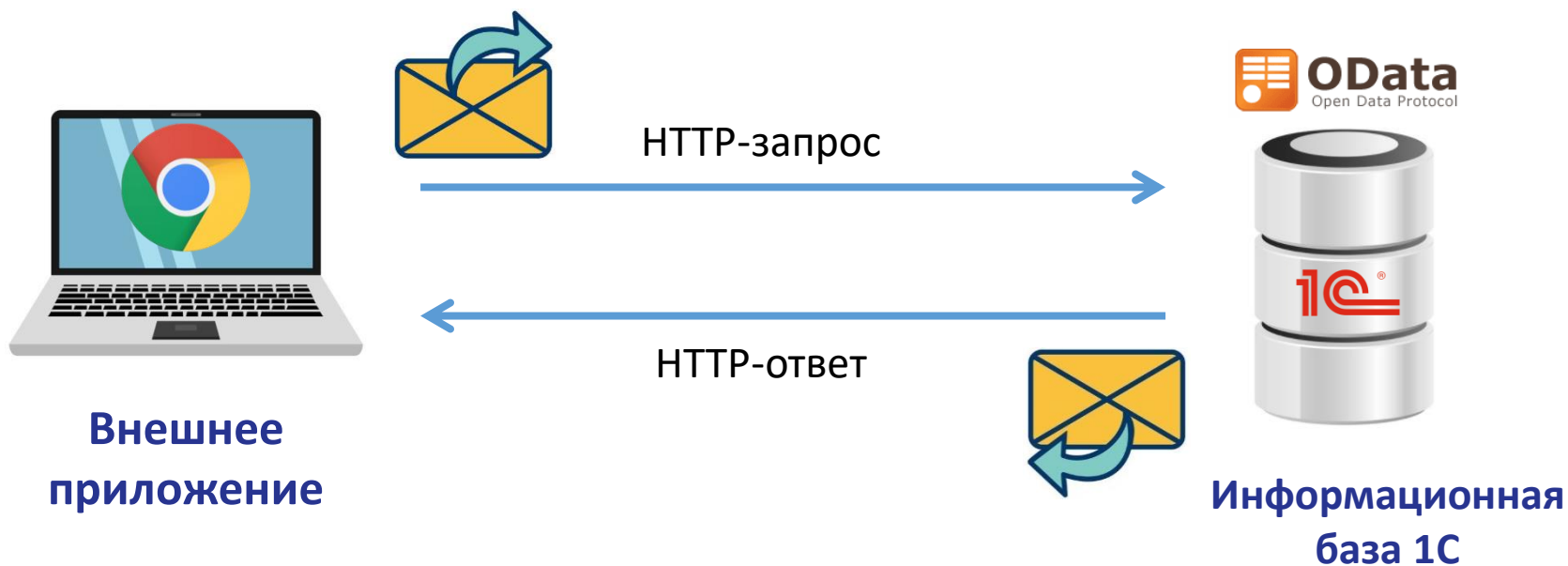
Для доступа к данным используется протокол OData версии 3 (<https://www.odata.org/>)



# OData. Терминология

- **Сущность** - нечто, обладающее идентичностью. Сущность также обладает набором свойств. Некоторые свойства описывают ее идентичность, комбинация которых определяет ключ сущности. По этому ключу можно получить конкретную сущность.
- **Набор сущностей** - коллекция сущностей определенного типа
- **Составной тип** - набор свойств, не обладающий идентичностью
- **Функция** - набор некоторых операций, выполняемых на стороне сервера, возвращающий данные (не обязательно сущность или набор сущностей) и не приводящий к наблюдаемым побочным эффектам (изменениям данных). Функция обязательно связана с сущностью или набором сущностей
- **Действие** - функция, которая может изменять данные

# Взаимодействие с интерфейсом OData



`http://host/Demo/odata/standard.odata/Catalog_Контрагенты`

↑  
Адрес ИБ

↑  
Признак обращения к  
интерфейсу OData

↑  
Имя ресурса, к которому  
выполняется обращение

# Правила формирования имени ресурса

Объект конфигурации	Префикс имени для указания в URL
Справочник	Catalog
Документ	Document
Журнал документов	DocumentJournal
Константа	Constant
План обмена	ExchangePlan
План счетов	ChartOfAccounts
План видов расчета	ChartOfCalculationTypes
План видов характеристик	ChartOfCharacteristicTypes
Регистр сведений	InformationRegister
Регистр накопления	AccumulationRegister
Регистр расчета	CalculationRegister
Регистр бухгалтерии	AccountingRegister
Бизнес-процесс	BusinessProcess
Задача	Task

http://host/Demo/odata/standard.odata/Catalog\_Контрагенты



Имя ресурса, к которому выполняется обращение

# Параметры обращения к ресурсу

[http://host/Demo/odata/standard.odata/Catalog\\_Контрагенты?ключ=значение&ключ2=значение2](http://host/Demo/odata/standard.odata/Catalog_Контрагенты?ключ=значение&ключ2=значение2)

Параметры обращения к ресурсу

Можно использовать следующие ключевые слова:

- **\$format** - формат получения данных
  - \$format=atom - возвращает данные в формате **atom+xml**
  - \$format=json - возвращает данные в формате **json**
- **\$metadata** - указывает, что требуется получить описание стандартного интерфейса Odata
- **\$filter** - описывает отбор, применяемый при получении данных
- **\$select** - описывает перечень свойств сущности, которые нужно получить

# Правила формирования условия отбора

## Логические операции:

Описание	Имя	Пример
Равно	eq	/Catalog_Города?\$filter=Description eq 'Главный'
Не равно	ne	/Catalog_Города?\$filter=Description ne 'Пермь'
Больше	gt	/Catalog_Товары?\$filter=Цена gt 10
Больше или равно	ge	/Catalog_Товары?\$filter=Цена ge 10
Меньше	lt	/Catalog_Товары?\$filter=Цена lt 10
Меньше или равно	le	/Catalog_Товары?\$filter=Цена le 10
Логическое Или	or	/Catalog_Товары?\$filter=Цена lt 10 or Цена gt 100
Логическое И	and	/Catalog_Товары?\$filter=Цена gt 10 and Цена lt 100
Отрицание	not	/Catalog_Товары?\$filter=not (Цена eq 10)

## Пример:

[http://host/odata/standard.odata/Catalog\\_Товары?\\$filter=Имя eq 'Молоко' and Цена lt 2500](http://host/odata/standard.odata/Catalog_Товары?$filter=Имя eq 'Молоко' and Цена lt 2500)

# HTTP-методы при работе с OData

- **Получение** данных - метод **GET**
- **Создание** объекта - метод **POST**
- **Обновление** данных:
  - метод **PATCH** - в этом случае можно указывать только те свойств, которые необходимо обновить
  - метод **PUT** - в этом случае необходимо указывать все свойства сущности
- **Удаление** данных - метод **DELETE**

# Что нужно сделать для работы с 1С через OData

1. Установить веб-сервер (IIS или Apache)
2. Опубликовать стандартный интерфейс OData на веб-сервере:  
*в конфигураторе «Администрирование» -> «Публикация на веб-сервере»*

Публикация на веб-сервере

Основные OpenID Прочие

Имя: Demo TrdBase

Веб-сервер: Internet Information Services

Каталог: C:\inetpub\wwwroot\Demo TrdBase\

Публиковать точный клиент и веб-клиент

Публиковать стандартный интерфейс OData

Web-сервисы HTTP сервисы

Публиковать Web-сервисы по умолчанию

Публиковать Web-сервисы:

Имя	Адрес
<input type="checkbox"/> Exchange	exchange.1cws
<input type="checkbox"/> Exchange_2_0_1_6	exchange_2_0_1_6.1cws
<input type="checkbox"/> InterfaceVersion	InterfaceVersion.1cws

Публиковать Web-сервисы расширений по умолчанию

Публиковать дистрибутив

Расположение публикуемого дистрибутива:

x86:

x86\_64:

Использовать аутентификацию операционной системы

Адрес перехода при окончании работы веб-клиента:

Опубликовать Отключить Сохранить Загрузить Закрнуть Справка

# Что нужно сделать для работы с 1С через OData

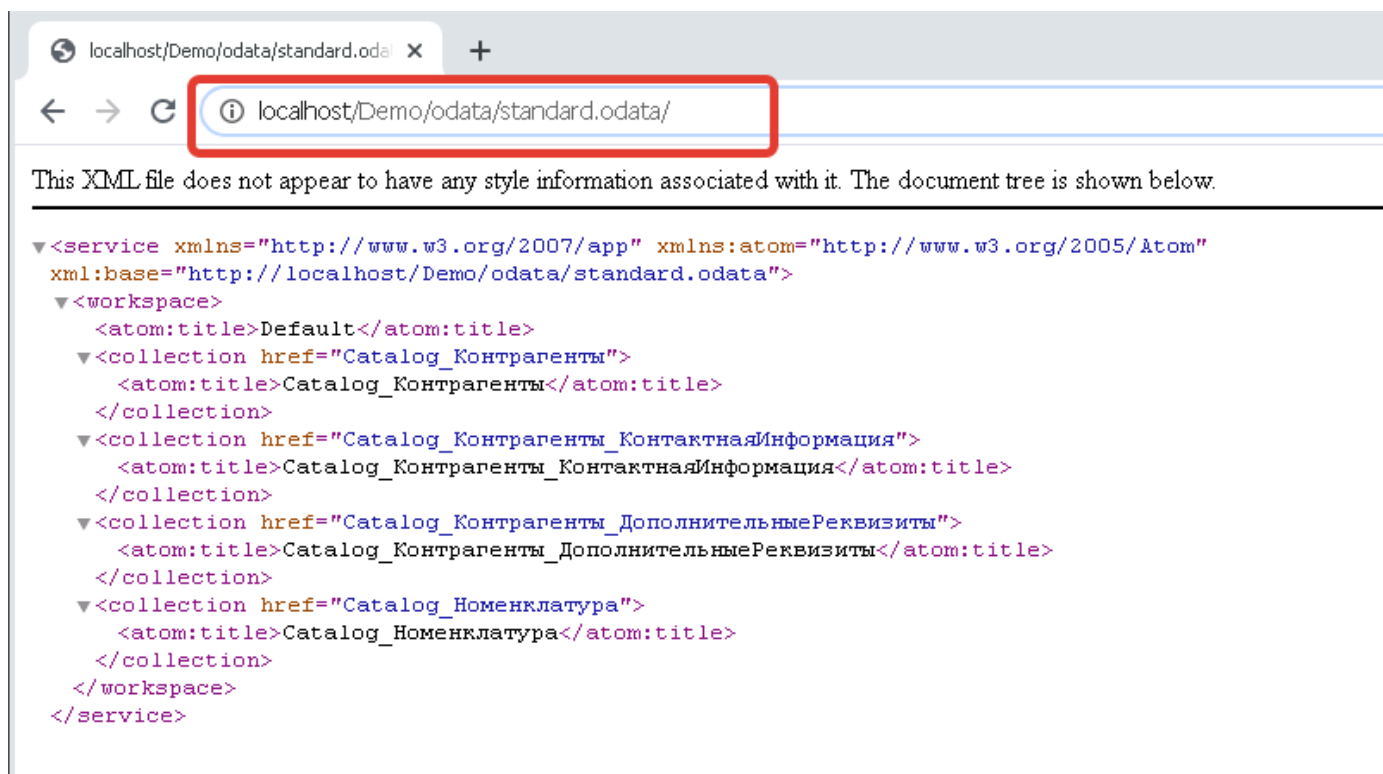
3. Определить состав объектов, с которыми нужно будет работать через интерфейс OData и открыть доступ к ним, используя метод глобального контекста:

**УстановитьСоставСтандартногоИнтерфейсаOData(МассивОбъектов)**

```
//Пример добавления в состав OData справочника "Номенклатура  
Массив = Новый Массив;  
Массив.Добавить (Метаданные.Справочники.Номенклатура) ;  
  
УстановитьСоставСтандартногоИнтерфейсаOData (Массив) ;
```

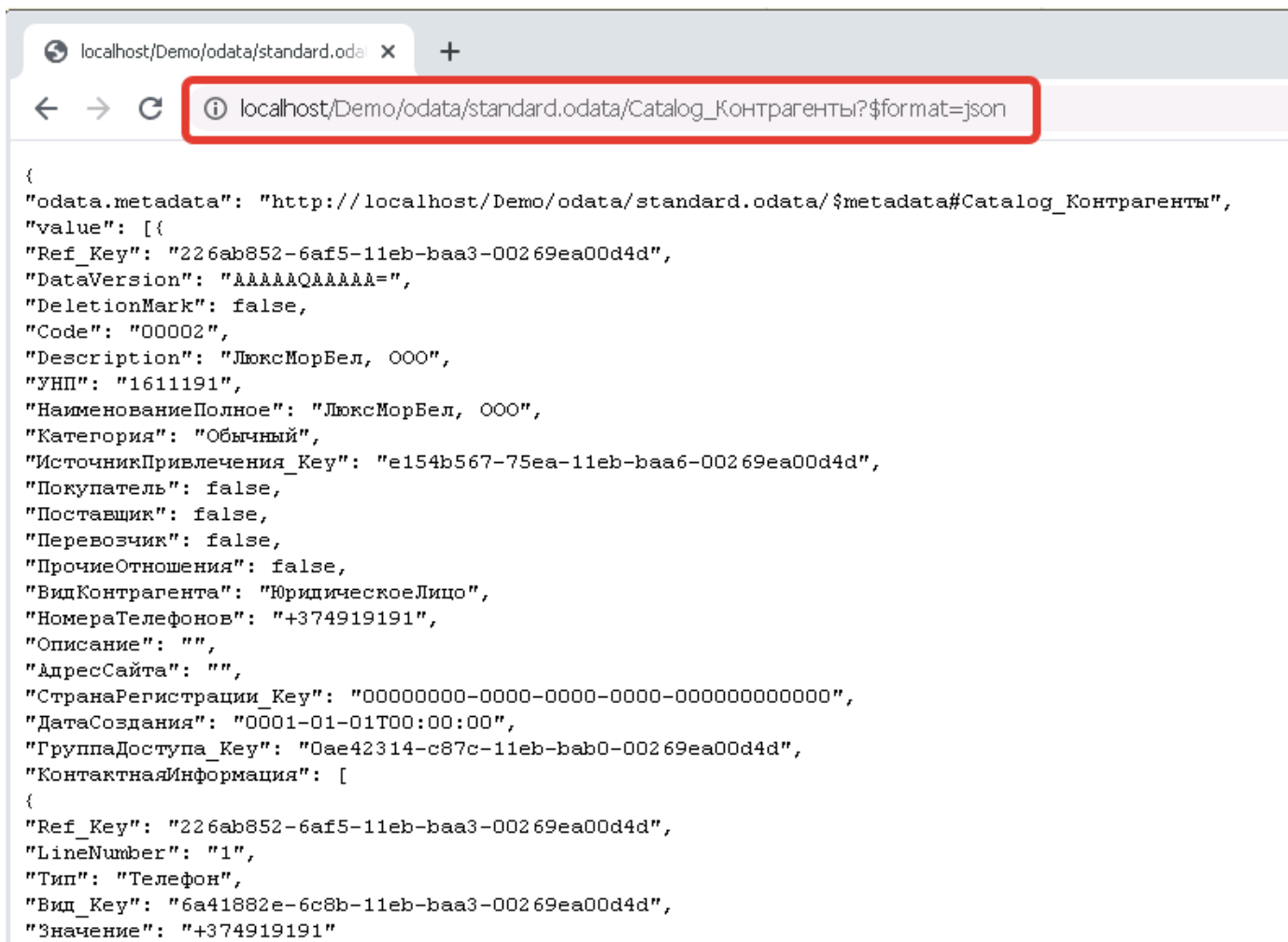
# Как проверить работу?

Заходим в браузер и в адресной строке указываем путь к стандартному интерфейсу OData:



```
<?xml version="1.0" encoding="utf-8" />
<service xmlns="http://www.w3.org/2007/app" xmlns:atom="http://www.w3.org/2005/Atom"
xml:base="http://localhost/Demo/odata/standard.odata">
  <workspace>
    <atom:title>Default</atom:title>
    <collection href="Catalog_Контрагенты">
      <atom:title>Catalog_Контрагенты</atom:title>
    </collection>
    <collection href="Catalog_Контрагенты_КонтактнаяИнформация">
      <atom:title>Catalog_Контрагенты_КонтактнаяИнформация</atom:title>
    </collection>
    <collection href="Catalog_Контрагенты_ДополнительныеРеквизиты">
      <atom:title>Catalog_Контрагенты_ДополнительныеРеквизиты</atom:title>
    </collection>
    <collection href="Catalog_Номенклатура">
      <atom:title>Catalog_Номенклатура</atom:title>
    </collection>
  </workspace>
</service>
```

# Пример обращения к справочнику «Контрагенты»



The screenshot shows a web browser window with the address bar containing the URL: `localhost/Demo/odata/standard.odata/Catalog_Контрагенты?$format=json`. The browser displays a JSON response from an OData service. The response is a single object with a metadata property and a value array containing one contract record. The record includes various fields such as Ref\_Key, DataVersion, Description, and Contact Information.

```
{
  "odata.metadata": "http://localhost/Demo/odata/standard.odata/$metadata#Catalog_Контрагенты",
  "value": [{
    "Ref_Key": "226ab852-6af5-11eb-baa3-00269ea00d4d",
    "DataVersion": "AAAAAAAAAAAA=",
    "DeletionMark": false,
    "Code": "00002",
    "Description": "ЛюксМорБел, ООО",
    "УНП": "1611191",
    "НаименованиеПолное": "ЛюксМорБел, ООО",
    "Категория": "Обычный",
    "ИсточникПривлечения_Key": "e154b567-75ea-11eb-baa6-00269ea00d4d",
    "Покупатель": false,
    "Поставщик": false,
    "Перевозчик": false,
    "ПрочиеОтношения": false,
    "ВидКонтрагента": "ЮридическоеЛицо",
    "НомераТелефонов": "+374919191",
    "Описание": "",
    "АдресСайта": "",
    "СтранаРегистрации_Key": "00000000-0000-0000-0000-000000000000",
    "ДатаСоздания": "0001-01-01T00:00:00",
    "ГруппаДоступа_Key": "0ae42314-c87c-11eb-bab0-00269ea00d4d",
    "КонтактнаяИнформация": [
      {
        "Ref_Key": "226ab852-6af5-11eb-baa3-00269ea00d4d",
        "LineNumber": "1",
        "Тип": "Телефон",
        "Вид_Key": "6a41882e-6c8b-11eb-baa3-00269ea00d4d",
        "Значение": "+374919191"
      }
    ]
  }]
}
```

# Руководство по работе с интерфейсом OData

its.1c.ru/db/v8318doc#bookmark:dev:T1000001358

Об 1С:ИТС Тест-драйв Заказать ИТС Задать вопрос Обновить ПО Оценить 1С Купить кассу

Главная > Инструкции по разработке на 1С > 1С:Предприятие 8.3.18. Документация > Руководство разработчика > Глава 17. Механизмы интернет-сервисов

Глава 17. Механизмы интернет-сервисов

- 17.1. Web-сервисы
- 17.2. HTTP-сервисы
  - 17.2.1. Стандартный интерфейс OData**
    - 17.2.1.1. Общая информация
    - 17.2.1.2. Общие принципы работы
    - 17.2.1.3. Представление данных
    - 17.2.1.4. Правила формирования имени ресурса
    - 17.2.1.5. Правила формирования условия отбора
    - 17.2.1.6. Параметры запроса
    - 17.2.1.7. Способы получения описания стандартного интерфейса OData
    - 17.2.1.8. Способы получения данных
    - 17.2.1.9. Выполнение функций и действий
    - 17.2.1.10. Ошибочные ситуации
    - 17.2.1.11. Установка значений разделителей
    - 17.2.1.12. Публикация стандартного интерфейса OData

< Назад 0

## 17.2.1. Стандартный интерфейс OData

### 17.2.1.1. Общая информация

Стандартный интерфейс OData в «1С:Предприятии» предназначен для получения доступа к данным системы (например, если прикладное решение стоит на поддержке). Для получения такого доступа не указывается, какие объекты конфигурации будут использоваться таким образом (см. [здесь](#)).

Для доступа к данным используется протокол OData (<http://www.odata.org/>, на английском языке) версии [version-3-0-core-protocol](#), на английском языке). Поддерживаются следующие форматы представления: автоматически генерируется стандартный интерфейс OData, который позволяет читать данные «1С:Предприятие». Прикладное решение на базе «1С:Предприятия» может выступать как клиентом, так и сервером практически никаких дополнительных действий осуществлять не надо (эта возможность предоставления стандартного интерфейса OData, необходимо в прикладном решении реализовать программный слой, который реализует стандартных интерфейсов «1С:Предприятия», например объекта `HTTPСоединение`).

В одной системе могут одновременно существовать и SOAP-сервисы (которые реализуются в «1С:Предприятии» в виде OData и HTTP-сервис (см. [здесь](#)).

Стандартный интерфейс OData можно использовать для следующих задач:

- Интеграция прикладного решения с различными сайтами (например, на базе Microsoft SharePoint);
- Реализация сторонними средствами дополнительной функциональности прикладного решения без использования стандартных средств «1С:Предприятия»;
- Загрузка данных в прикладное решение и выгрузка данных из него;
- Интеграция прикладного решения с корпоративными системами, возможно даже без использования стандартных средств «1С:Предприятия».

# Примеры запросов



# Примеры задач. OData

- ~~Получить~~ всех контрагентов (GET)
- ~~Найти~~ контрагента по ИНН (GET)
- ~~Создать~~ нового контрагента (POST)
- ~~Изменить~~ существующего контрагента (PUT/PATCH)
- ~~Удалить~~ контрагента (DELETE)
- ~~Найти~~ документ (GET)

# Чтение данных

Чтение данных выполняется с помощью GET-запроса

**Пример чтения списка товаров:**

[http://host/base/odata/standard.odata/Catalog\\_Товары](http://host/base/odata/standard.odata/Catalog_Товары)

**Пример чтения одного товара:**

[http://host/base/odata/standard.odata/Catalog\\_Товары\(guid'value'\)](http://host/base/odata/standard.odata/Catalog_Товары(guid'value'))

**Пример чтения набора записей подчиненного регистра сведений:**

[http://host/base/odata/standard.odata/InformationRegister\\_ЦеныПоставщиков\(Recorder\\_Key=guid'value'\)](http://host/base/odata/standard.odata/InformationRegister_ЦеныПоставщиков(Recorder_Key=guid'value'))

**Пример чтения независимого регистра сведений:**

[http://host/base/odata/standard.odata/InformationRegister\\_КурсыВалют/SliceLast?Condition=Валюта/ОсновнаяВалюта\\_Key eq guid'value'](http://host/base/odata/standard.odata/InformationRegister_КурсыВалют/SliceLast?Condition=Валюта/ОсновнаяВалюта_Key eq guid'value)

# Создание объекта

Создание объекта выполняется с помощью POST-запроса

## Пример POST-запроса (формат json):

```
POST http://host/base/odata/standard.odata/Catalog_Товары HTTP/1.1
Accept: application/json
Accept Charset: UTF 8
User Agent: Fiddler
Content Type: application/json
Content Length: 2426
{
  "DeletionMark":false,
  "Parent_Key":"bbb079ae 8c51 11db a9b0 00055d49b45e",
  "IsFolder":false
  "Code":"000000800",
  "Description":"Шлепанцы",
  "Артикул":"SL56X",
  "Поставщик_Key":"086715b0 f348 11db a9c5 00055d49b45e",
  "Вид":"Товар",
  "Штрихкод":null,
  "Описание":"Шлепанцы пляжные"
}
```

# Обновление объекта

Для обновления объекта необходимо выполнить **PUT-/PATCH-**запрос

## Пример PUT-запроса:

```
PUT http://host /base/odata/standard.odata/Document_Документ(guid'19961ec3 3c5f 11e7 8785 50465da19fe4')?$format=json
{
  "Date": "2015 01 01T12:01:07",
  "Организация@odata.bind": "Catalog_Организации(guid'298584b5 92e4 11e3 8bc3 0050568b1678')",
  "Склад@odata.bind": http://host /base/odata/standard.odata/Catalog_Склады(guid'081bc346 2abc 13e4 a1bd 0050568b1688')
  ...
}
```

## Пример PATCH-запроса:

```
PATCH http://host/base/odata/standard.odata/Catalog_Магазины(guid'value')?$format=json HTTP/1.1
Host: host
Connection: keep alive
Accept: application/json
Content Length: 638
{
  "odata.metadata": "http://host/base/odata/standard.odata/$metadata#Catalog_Магазины/@Element",
  "Description": "Новое описание магазина",
  "ТорговыеЗаль@odata.type" : "Collection(StandardODATA.Catalog_Магазины_ТорговыеЗалы_RowType)"
  "ТорговыеЗалы": [
    {
      "LineNumber": "1",
      "Название": "Синий зал",
      "Площадь": 56,
      "ДатаОткрытия": "2015 01 01T00:00:00"
    },
    {
      "LineNumber": "2",
      "Название": "Красный зал",
      "Площадь": 56,
      "ДатаОткрытия": "2015 06 13T11:45:41"
    }
  ]
}
```

# Удаление объекта

Для обновления объекта необходимо выполнить **DELETE**-запрос

## Пример запроса на удаление товара:

```
DELETE http://host/base/odata/standard.odata/Catalog_Товары(guid'value') HTTP/1.1  
User-Agent:Fiddler Everywhere  
Content-Type:application/json
```

**ВАЖНО:** Пометка на удаление не выполняется, объект удаляется непосредственно

# Конец 62-го занятия

Подписывайтесь на нас в социальных сетях 😊



<https://www.youtube.com/channel/UC124tAuP11GlaGpK3YGHzw>



<https://www.facebook.com/ironskillsby/>



[https://www.instagram.com/ironskills\\_/](https://www.instagram.com/ironskills_/)