

#T63

Интеграция и обмен данными. Создание HTTP-сервиса. Создание SOAP-сервиса

Программист 1С: с нуля до профи



Из этой лекции вы узнаете

- Как создать **HTTP-сервис** в 1С
- Что такое **SOAP-протокол** и для чего он нужен
- Как создать **SOAP-сервис (Web-сервис)** в 1С
- В чем разница между **HTTP-сервисом** и **SOAP-сервисом (Web-сервисом)**

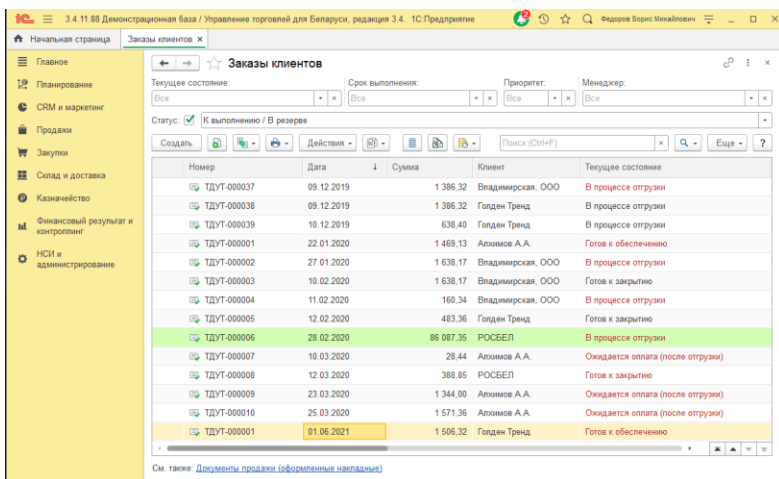
Постановка задачи

Постановка задачи

В рамках интеграции с сайтом интернет-магазина нужно реализовать возможность получать из 1С информацию о товарах и ценах.

На стороне 1С необходимо реализовать программный интерфейс (API) для получения из базы данных о номенклатуре и актуальных ценах.

Информационная база 1С



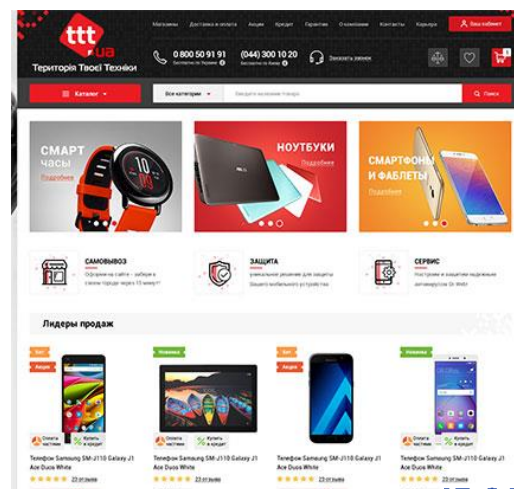
Скриншот интерфейса информационной базы 1С. В центре экрана отображается таблица «Заказы клиентов» с колонками: Номер, Дата, Сумма, Клиент, Текущее состояние. В таблице перечислены различные заказы с датами от 09.12.2019 до 01.06.2021, суммами от 28.44 до 86 087.35 и статусами от «В процессе отгрузки» до «Готов и закрыто».

Номер	Дата	Сумма	Клиент	Текущее состояние
ТДУТ-000037	09.12.2019	1 386.32	Владимирская, ООО	В процессе отгрузки
ТДУТ-000038	09.12.2019	1 386.32	Голден Тренд	В процессе отгрузки
ТДУТ-000039	10.12.2019	638.40	Голден Тренд	В процессе отгрузки
ТДУТ-000001	22.01.2020	1 469.13	Алюмов А.А.	Готов и обеспечено
ТДУТ-000002	27.01.2020	1 638.17	Владимирская, ООО	В процессе отгрузки
ТДУТ-000003	10.02.2020	1 638.17	Владимирская, ООО	Готов и закрыто
ТДУТ-000004	11.02.2020	160.34	Владимирская, ООО	В процессе отгрузки
ТДУТ-000005	12.02.2020	483.36	Голден Тренд	Готов и закрыто
ТДУТ-000006	28.02.2020	86 087.35	РОСБЕЛ	В процессе отгрузки
ТДУТ-000007	10.03.2020	28.44	Алюмов А.А.	Ожидается оплата (после отгрузки)
ТДУТ-000008	12.03.2020	388.85	РОСБЕЛ	Готов и закрыто
ТДУТ-000009	23.03.2020	1 344.00	Алюмов А.А.	Ожидается оплата (после отгрузки)
ТДУТ-000010	25.03.2020	1 571.36	Алюмов А.А.	Ожидается оплата (после отгрузки)
ТДУТ-000001	01.06.2021	1 506.32	Голден Тренд	Готов и обеспечено

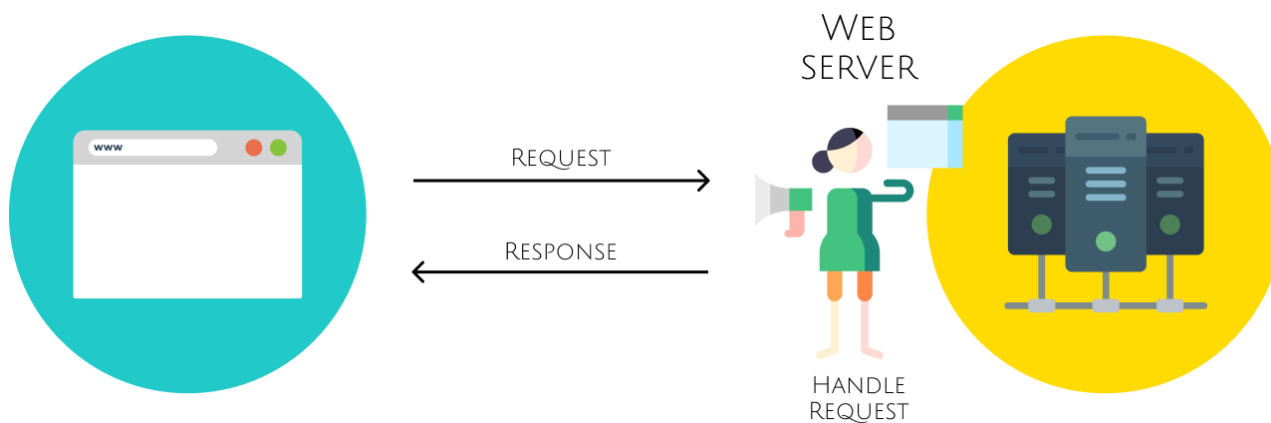
Товары, цены



Сайт интернет-магазина



Разработка HTTP-сервиса



HTTP-сервис. URL

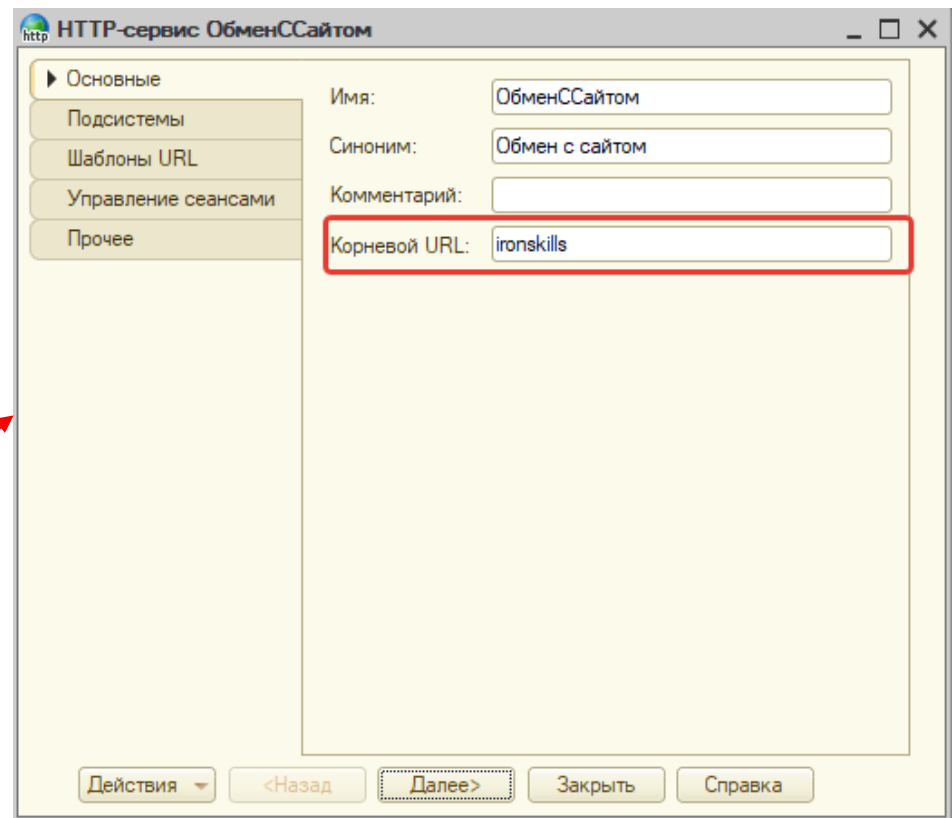
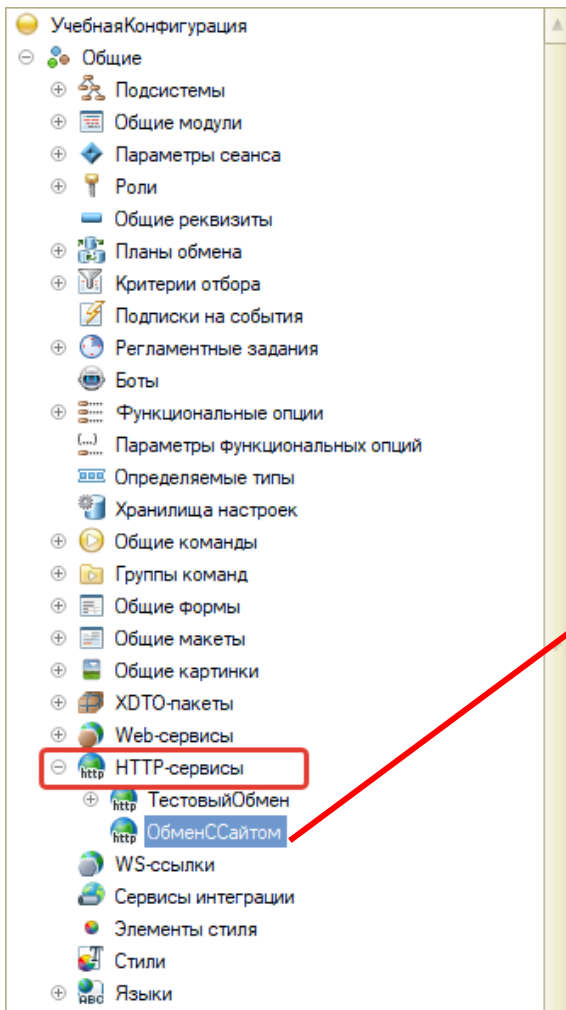
При обращении к HTTP-сервису формируется URL, который выглядит следующим образом:

<http://host/base/hs/корневойURL/относительныйURL>

Составные части:

- **http://host/base/** - URL публикации базы
- **hs** – признак обращения к HTTP-сервису
- **корневойURL** - имя ресурса, которое определяет группу ресурсов, связанных общим смыслом
- **относительныйURL** - определяет ресурс, к которому будет выполняться обращение

Создание HTTP-сервиса

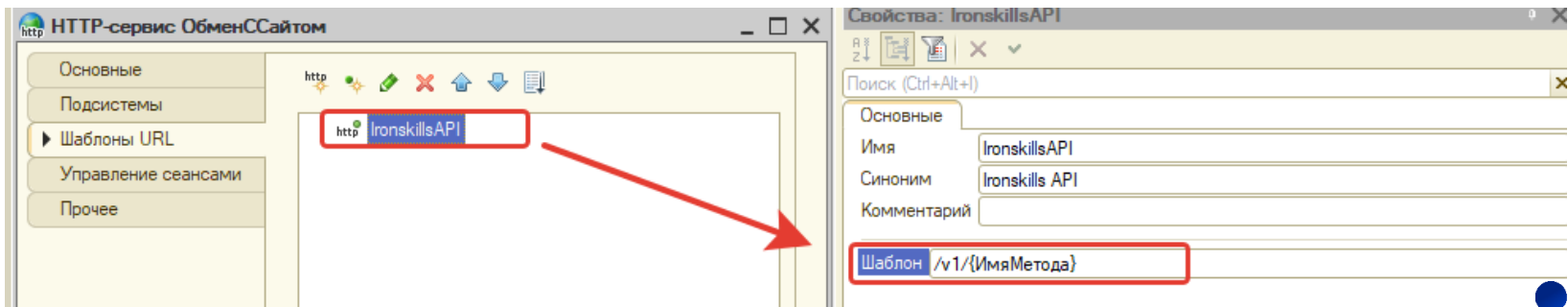


Создание шаблона URL

Шаблон URL используется для задания шаблонов пути, по которым может происходить обращение к сервису.

Шаблон пути может содержать следующие символы:

- Любые символы, допустимые в идентификаторах "1С:Предприятия",
- Символ "/",
- Символы "{", "}" с непустым текстом между ними,
- Символ "*".



Создание метода

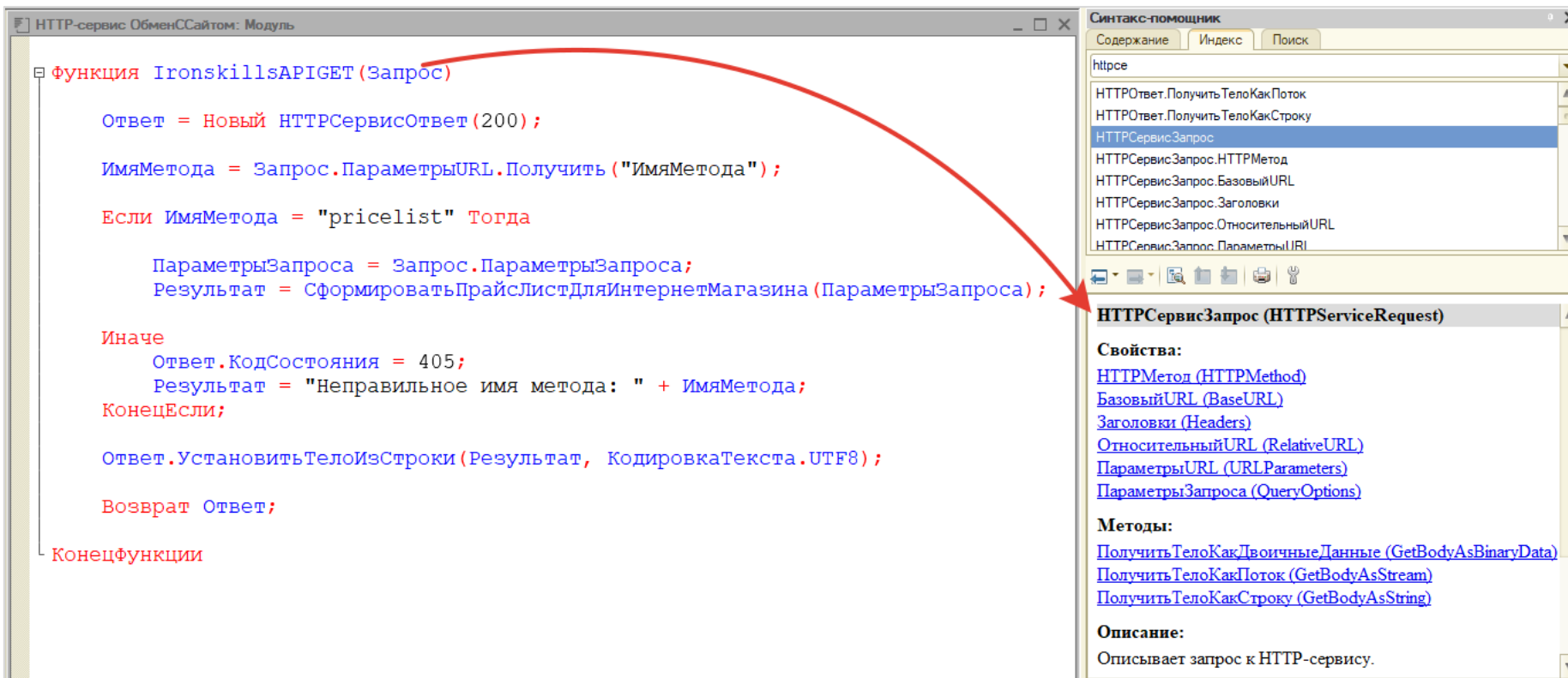
The screenshot displays the Ironskills API management interface. The main window, titled "HTTP-сервис ОбменССайтом", shows a list of endpoints under "http://IronskillsAPI". A "GET" method is highlighted with a red box. To the right, the "Свойства: GET" (Properties: GET) dialog is open, showing the following details:

- Имя: GET
- Синоним: GET
- Комментарий: (empty)
- HTTP-метод: GET
- Обработчик: IronskillsAPIGET

A red arrow points from the "Обработчик" field in the properties dialog to the code editor below. The code editor shows the following code:

```
Функция IronskillsAPIGET (Запрос)  
    Ответ = Новый HTTPСервисОтвет (200) ;  
    Возврат Ответ ;  
Конецфункции
```

Создание метода



The image shows a development environment with two windows. The main window, titled "HTTP-сервис ОбменССайтом: Модуль", contains the following code for a function named "IronskillsAPIGET":

```
Функция IronskillsAPIGET(Запрос)  
  
    Ответ = Новый HTTPСервисОтвет(200);  
  
    ИмяМетода = Запрос.ПараметрыURL.Получить("ИмяМетода");  
  
    Если ИмяМетода = "pricelist" Тогда  
  
        ПараметрыЗапроса = Запрос.ПараметрыЗапроса;  
        Результат = СформироватьПрайсЛистДляИнтернетМагазина(ПараметрыЗапроса);  
  
    Иначе  
        Ответ.КодСостояния = 405;  
        Результат = "Неправильное имя метода: " + ИмяМетода;  
    КонечЕсли;  
  
    Ответ.УстановитьТелоИзСтроки(Результат, КодировкаТекста.UTF8);  
  
    Возврат Ответ;  
  
КонечФункции
```

The right window, titled "Синтаксис-помощник", shows a search for "httpse" and lists several related items. The item "HTTPСервисЗапрос" is selected, and its details are shown below:

HTTPСервисЗапрос (HTTPServiceRequest)

Свойства:

- [HTTPМетод \(HTTPMethod\)](#)
- [БазовыйURL \(BaseURL\)](#)
- [Заголовки \(Headers\)](#)
- [ОтносительныйURL \(RelativeURL\)](#)
- [ПараметрыURL \(URLParameters\)](#)
- [ПараметрыЗапроса \(QueryOptions\)](#)

Методы:

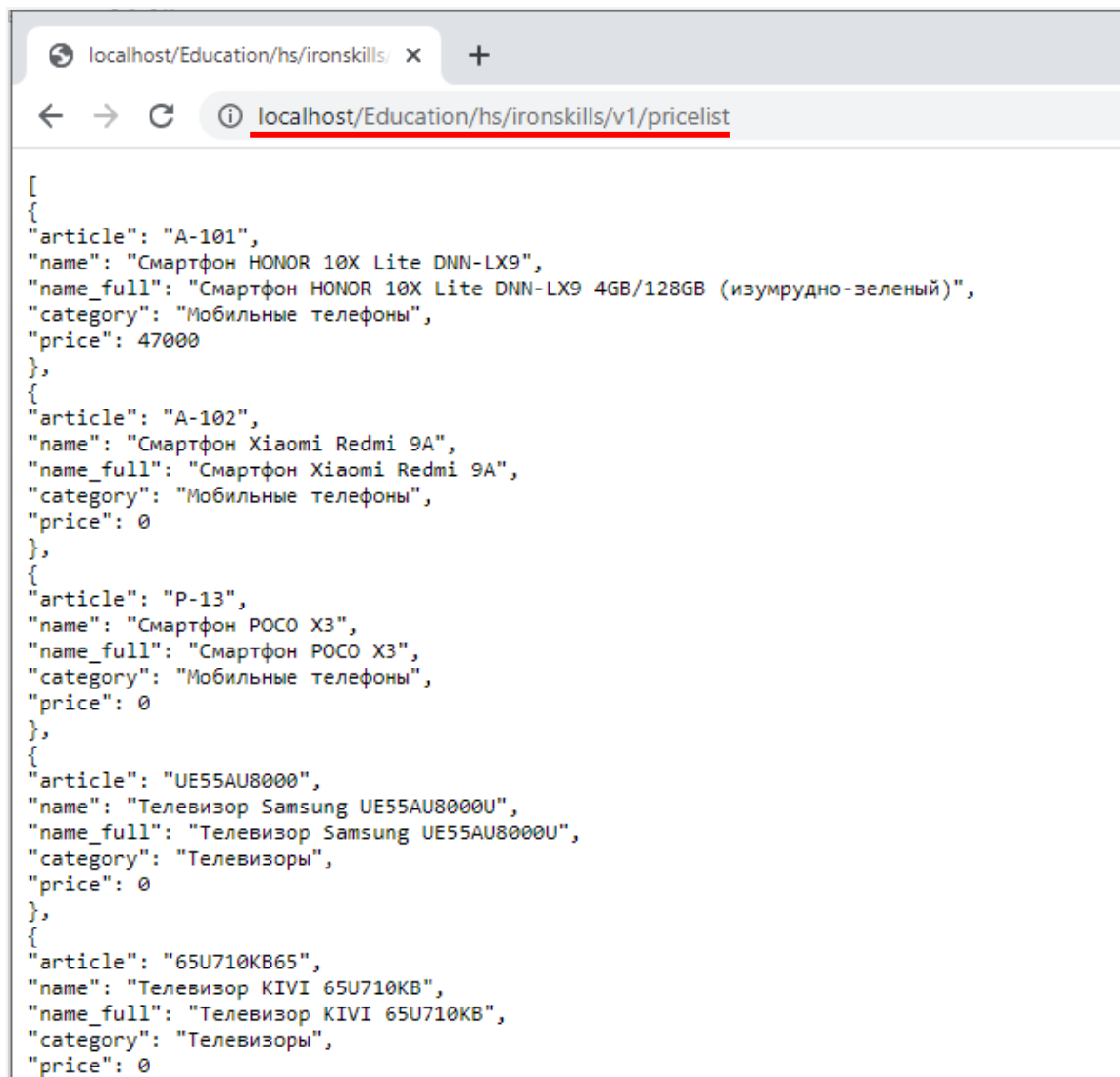
- [ПолучитьТелоКакДвоичныеДанные \(GetBodyAsBinaryData\)](#)
- [ПолучитьТелоКакПоток \(GetBodyAsStream\)](#)
- [ПолучитьТелоКакСтроку \(GetBodyAsString\)](#)

Описание:

Описывает запрос к HTTP-сервису.

A red arrow points from the "ИмяМетода" variable in the code to the "HTTPСервисЗапрос" entry in the syntax assistant.

Тест HTTP-сервиса



```
[
  {
    "article": "A-101",
    "name": "Смартфон HONOR 10X Lite DNN-LX9",
    "name_full": "Смартфон HONOR 10X Lite DNN-LX9 4GB/128GB (изумрудно-зеленый)",
    "category": "Мобильные телефоны",
    "price": 47000
  },
  {
    "article": "A-102",
    "name": "Смартфон Xiaomi Redmi 9A",
    "name_full": "Смартфон Xiaomi Redmi 9A",
    "category": "Мобильные телефоны",
    "price": 0
  },
  {
    "article": "P-13",
    "name": "Смартфон POCO X3",
    "name_full": "Смартфон POCO X3",
    "category": "Мобильные телефоны",
    "price": 0
  },
  {
    "article": "UE55AU8000",
    "name": "Телевизор Samsung UE55AU8000U",
    "name_full": "Телевизор Samsung UE55AU8000U",
    "category": "Телевизоры",
    "price": 0
  },
  {
    "article": "65U710KB65",
    "name": "Телевизор KIVI 65U710KB",
    "name_full": "Телевизор KIVI 65U710KB",
    "category": "Телевизоры",
    "price": 0
  }
]
```

Создание документации

The screenshot shows the Postman application interface. On the left sidebar, under 'My Workspace', the collection 'IRONSKILLS - Обмен с сайтом' is selected and highlighted with a red box. A red arrow points from this collection to the 'Documentation' tab on the right. The 'Documentation' tab is currently active, showing a description and a 'Basic Auth' authorization method. Below the authorization method, there are input fields for 'Username' (containing 'Администратор') and 'Password'. A warning message is displayed: 'Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment, we recommend using variables. Learn more about variables'. At the bottom of the documentation panel, there is a link: 'View complete collection documentation ->'. The top of the interface shows the Postman logo, menu items (File, Edit, View, Help), and navigation tabs (Home, Workspaces, API Network, Reports, Explore). The bottom of the interface shows utility icons for Cookies, Capture requests, Bootcamp, Runner, Trash, and Help.

Создание документации

IRONSKILLS - Обмен с сайтом Fork | 0 View Collection Publish

Release Tag CURRENT Language cURL

IRONSKILLS - Обмен с сайтом

Make things easier for your teammates with a complete collection description.

Authorization Basic Auth

Username

GET Прайс-лист Open Request →

Make things easier for your teammates with a complete request description.

Authorization Basic Auth


Username

Query Params

category

JUMP TO

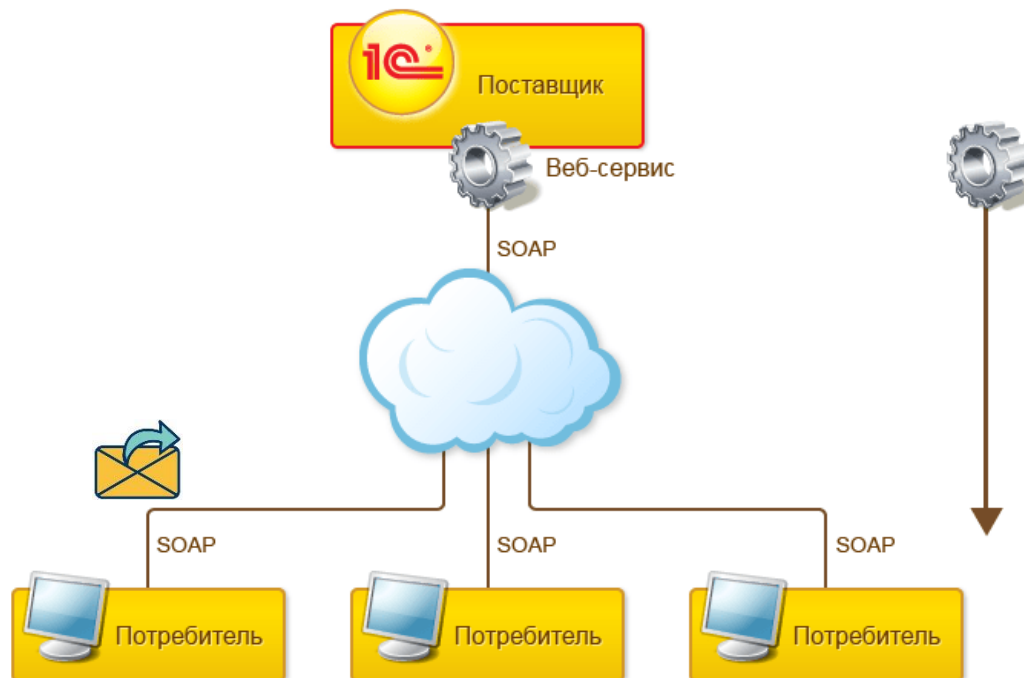
- Introduction
- GET Прайс-лист



Разработка Web-сервиса (SOAP-сервиса)

Протокол SOAP

SOAP (Simple Object Access Protocol) – протокол обмена структурированными сообщениями в распределённой вычислительной среде.

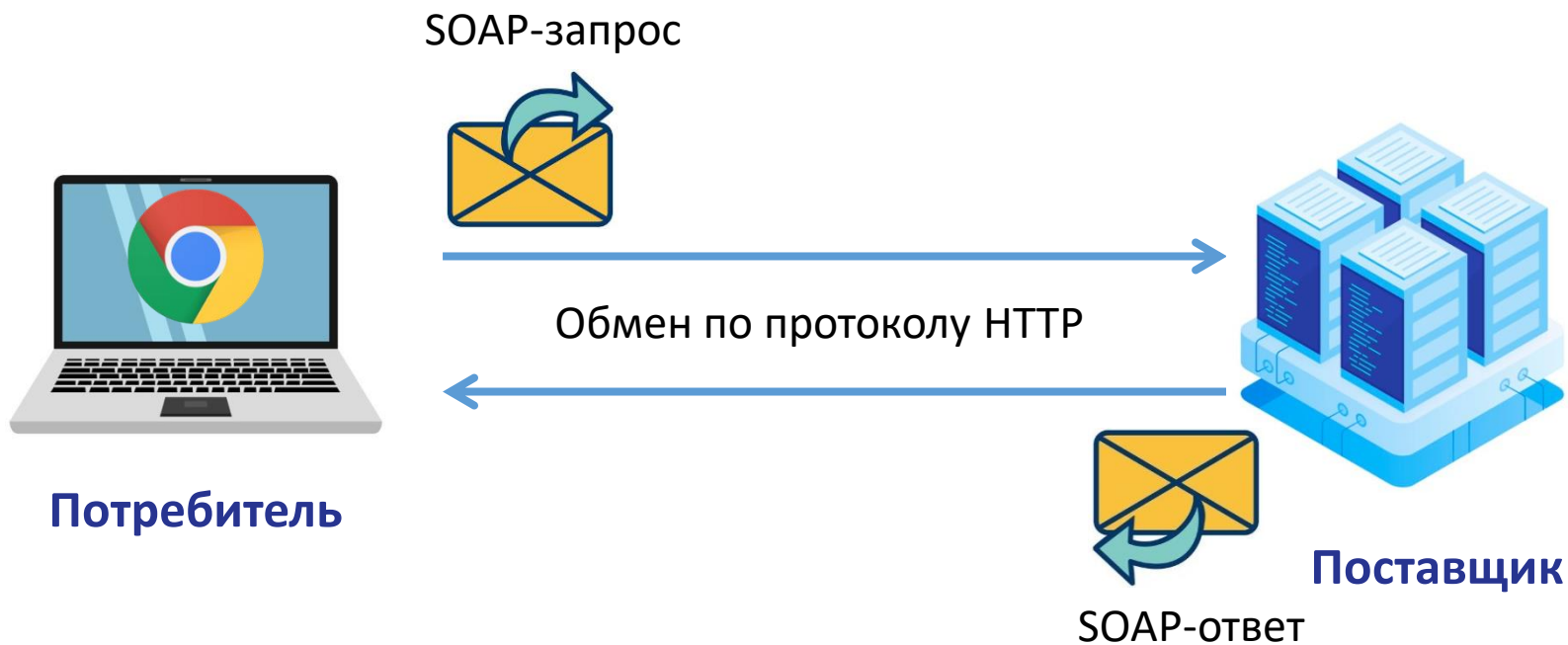


Протокол SOAP

Сообщение в SOAP – это XML-документ, состоящий из следующих элементов:

- **Envelope.** Корневой обязательный элемент. Определяет начало и окончание сообщения.
- **Header.** Необязательный элемент — заголовок. Содержит элементы, необходимые для обработки самого сообщения. Например, идентификатор сессии.
- **Body.** Основной элемент, содержит основную информацию сообщения. Обязательный.
- **Fault.** Элемент, содержащий информацию об ошибках, возникающих в процессе обработки сообщения. Необязательный.

Как это работает?



Примеры запроса и ответа в SOAP

Пример SOAP запроса:

```
1 <?xml version="1.0"?>
2 <soap:Envelope xmlns:soap="https://www.w3.org/2003/05/soap-envelope/"
3     soap:encodingStyle="http://www.w3.org/2003/05/soap-encoding/">
4     <soap:Body>
5         <m:GetPrice xmlns:m="https://www.w3schools.com/prices">
6             <m:Item>Apples</m:Item>
7         </m:GetPrice>
8     </soap:Body>
9 </soap:Envelope>
```

Пример SOAP ответа:

```
1 <?xml version="1.0"?>
2 <soap:Envelope xmlns:soap="https://www.w3.org/2003/05/soap-envelope/"
3     soap:encodingStyle="http://www.w3.org/2003/05/soap-encoding/">
4     <soap:Body>
5         <m:GetPriceResponse xmlns:m="https://www.w3schools.com/prices">
6             <m:Price>1.90</m:Price>
7         </m:GetPriceResponse>
8     </soap:Body>
9 </soap:Envelope>
```

Web-сервис. URL

При обращении к HTTP-сервису формируется URL, который выглядит следующим образом:

<http://host/base/ws/ИмяWebСервиса>

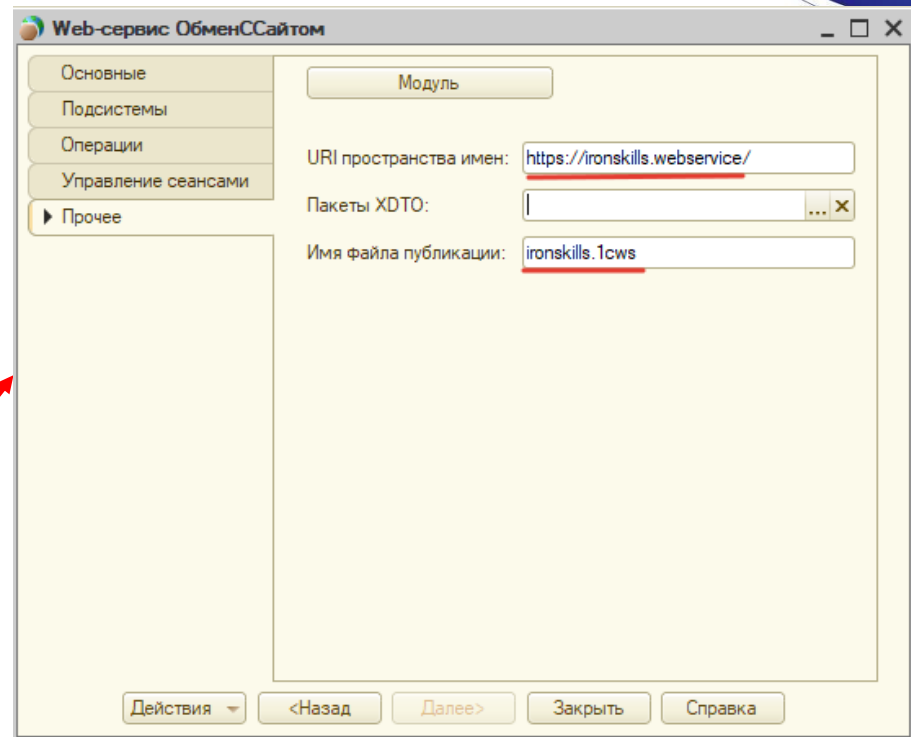
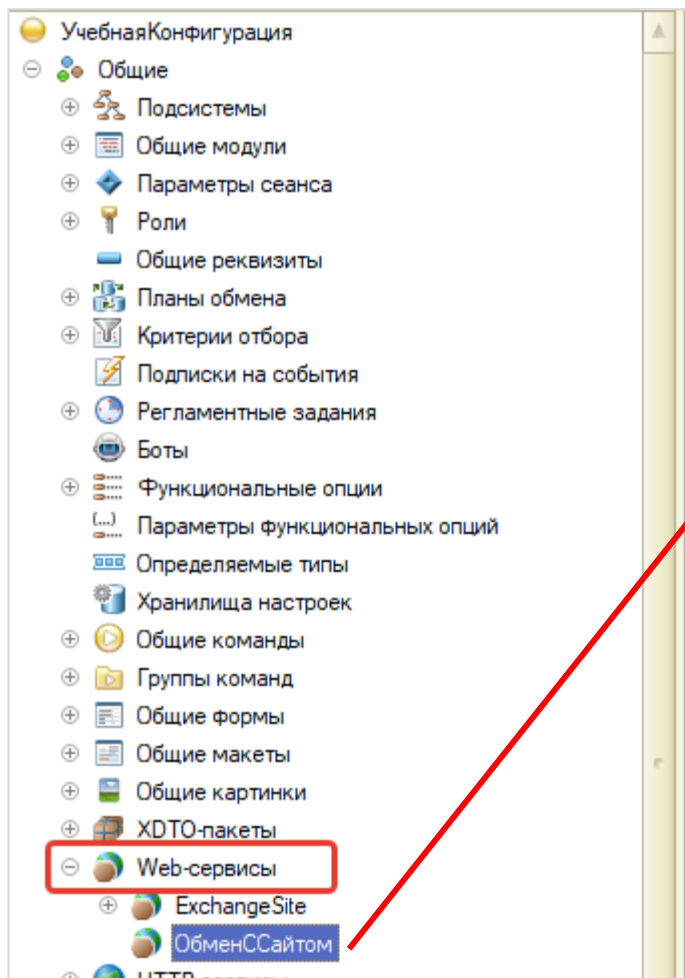
или

<http://host/base/ws/АдресWebСервиса>

Составные части:

- **<http://host/base/>** - URL публикации базы
- **ws** – признак обращения к Web-сервису
- **ИмяWebСервиса** - имя Web-сервиса (имя объекта конфигурации)
- **АдресWebСервиса** – альтернативное имя для доступа к Web-сервису (свойство «Имя файла публикации» объекта «Web-сервис»)

Создание Web-сервиса



Создание метода Web-сервиса

The screenshot illustrates the process of creating a web service method. It shows three main components:

- Main Window (Web-сервис ОбменССайтом):** A tree view on the left shows the project structure. The 'pricelist' method is selected and highlighted in blue. A red arrow points from this selection to the properties window.
- Properties Window (Свойства: pricelist):** This window displays the configuration for the selected method. The 'Имя' (Name) is 'pricelist', and the 'Синоним' (Synonym) is 'Pricelist'. The 'Тип возвращаемого значения' (Return type) is set to 'string (http://www.w3.org/2001/XMLSchema...)'.
- Code Editor (Web-сервис ОбменССайтом: Модуль):** The code editor shows the skeleton of the 'pricelist()' function. A red arrow points from the 'Имя процедуры' (Procedure name) field in the properties window to the function name in the code editor.

```
Функция pricelist()  
    // Вставить содержимое обработчика.  
Концефункции
```

Параметры метода Web-сервиса

The image shows a screenshot of a web service configuration tool. The main window is titled "Web-сервис ОбменСайтом". On the left, there is a sidebar with menu items: "Основные", "Подсистемы", "Операции", "Управление сеансами", and "Прочее". The main area displays a tree view with "pricelist" expanded to show a parameter "category". A red arrow points from the "category" parameter in the tree view to the "Свойства: category" properties window on the right. Another red arrow points from the "category" parameter in the tree view to the code editor at the bottom.

The "Свойства: category" window shows the following properties:

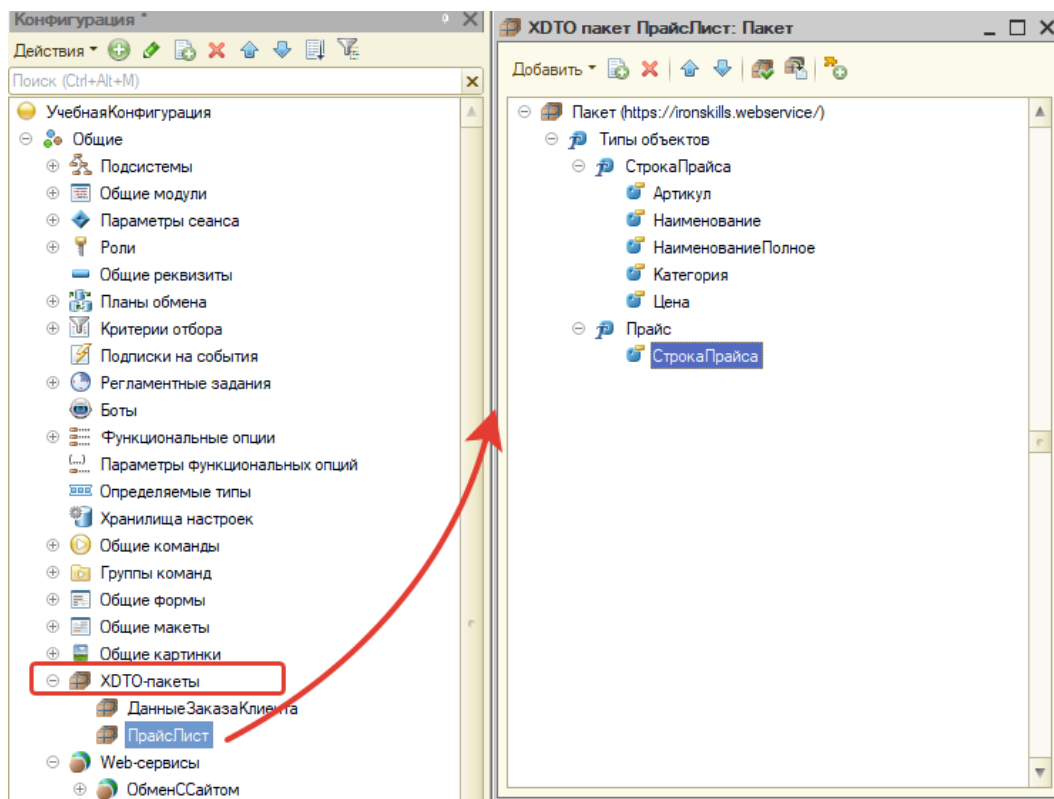
- Имя: category
- Синоним: Category
- Комментарий: (empty)
- Тип значения: string (http://www.w3.org/2001/XMLSchema)...
- Возможно пустое значение:
- Направление передачи: Входной

The code editor at the bottom shows the following code:

```
Функция pricelist(category)
    // Вставить содержимое обработчика.
Конецфункции
```

Создание XDTO-пакета

Пакет XDTO содержит описание некоторого множества типов, принадлежащих одному пространству имен – пространству имен пакета



Описание алгоритма метода Web-сервиса

Функция pricelist(category)

```
// Получить типы объектов XDTO
СтрокаПрайсаТип = фабрикаXDTO.Тип("https://ironskills.webservice/", "СтрокаПрайса");
ПрайсТип = фабрикаXDTO.Тип("https://ironskills.webservice/", "Прайс");

// Получаем вид номенклатуры
ВидНоменклатуры = Неопределено;
Если НЕ ПустаяСтрока(category) Тогда
    ВидНоменклатуры = Справочники.ВидыНоменклатуры.НайтиПоНаименованию(category, Истина);
КонецЕсли;

//получение данных (номенклатура и цены)
РезультатЗапроса = ПолучитьДанныеДляПрайсЛиста(ВидНоменклатуры);
Выборка = РезультатЗапроса.Выбрать();

//формирование ответа Web-сервиса
Прайс = фабрикаXDTO.Создать(ПрайсТип);

Пока Выборка.Следующий() Цикл

    СтрокаПрайса = фабрикаXDTO.Создать(СтрокаПрайсаТип);
    СтрокаПрайса.Артикул = Выборка.Артикул;
    СтрокаПрайса.Наименование = Выборка.Наименование;
    СтрокаПрайса.НаименованиеПолное = Выборка.НаименованиеПолное;
    СтрокаПрайса.Категория = СокрЛП(Выборка.ВидНоменклатуры);
    СтрокаПрайса.Цена = Выборка.Цена;

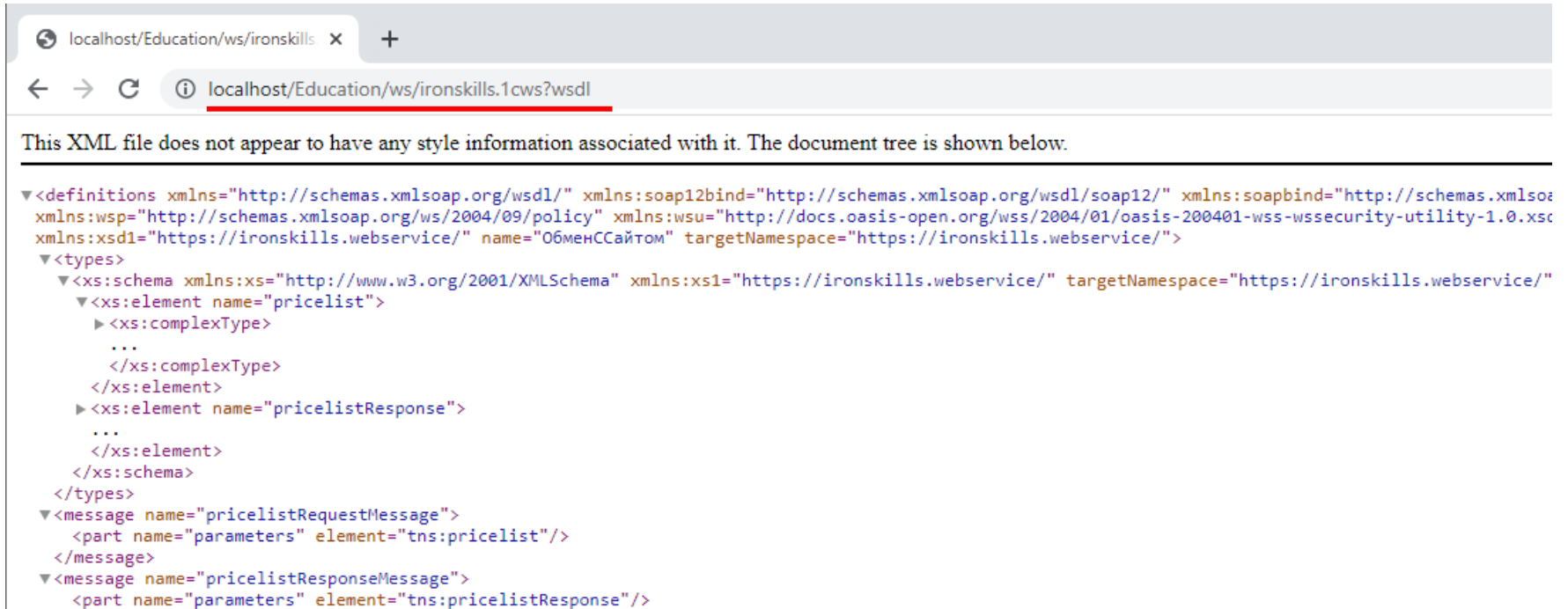
    Прайс.СтрокаПрайса.Добавить(СтрокаПрайса);

КонецЦикла;

Возврат Прайс;
```

Конецфункции

Тест Web-сервиса



localhost/Education/ws/ironskills x +

localhost/Education/ws/ironskills.1cws?wsdl

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<definitions xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:soap12bind="http://schemas.xmlsoap.org/wsdl/soap12/" xmlns:soapbind="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" xmlns:xsd1="https://ironskills.webservice/" name="ОбменССайтом" targetNamespace="https://ironskills.webservice/">
  <types>
    <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:xsl="https://ironskills.webservice/" targetNamespace="https://ironskills.webservice/">
      <xs:element name="pricelist">
        <xs:complexType>
          ...
        </xs:complexType>
      </xs:element>
      <xs:element name="pricelistResponse">
        ...
      </xs:element>
    </xs:schema>
  </types>
  <message name="pricelistRequestMessage">
    <part name="parameters" element="tns:pricelist"/>
  </message>
  <message name="pricelistResponseMessage">
    <part name="parameters" element="tns:pricelistResponse"/>
  </message>
</definitions>
```

Как протестировать методы SOAP-сервиса

- 1) Использовать специальные программы
 - **SoapUI**
- 2) Использовать расширения браузера
 - **Boomerang – SOAP & REST Client**
- 3) Написать внешнюю обработку на платформе 1С

Пример обращения к методу Web-сервиса

```
//получение описания веб-сервиса (WSDL)
АдресОписанияВебСервиса = "http://localhost/Education/ws/ironskills.1cws?wsdl";

Пользователь = "web";
Пароль       = "";

Определение = Новый WSOпределения(АдресОписанияВебСервиса, Пользователь, Пароль);

//создание прокси (соединения) для обращения к веб-сервису
URIПространстваИмен = "https://ironskills.webservice/";
ИмяВебСервиса      = "ОбменССайтом";

Прокси = Новый WSPрокси(Определение, URIПространстваИмен, ИмяВебСервиса, ИмяВебСервиса + "Soap");
Прокси.Пользователь = Пользователь;
Прокси.Пароль       = Пароль;

//вызов метода веб-сервиса
ПараметрКатегория = "";

Прайс = Прокси.pricelist(ПараметрКатегория);
```

Конец 63-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_/)