

## Лабораторная работа №5 Yandex API

### Цель:

Современные поисковые системы предоставляют богатый набор API-функций, реализуемых на базе протокола *http*. Например, ниже представлен список Yandex API, которые на текущий момент доступны пользователю.

<p><a href="#">API Яндекс.Карт</a> Используйте карты Яндекса на своем сайте, создавайте собственные карты, схемы проезда, геоинформационные сервисы.</p>	<p><a href="#">API Яндекс.Директа</a> Разрабатывайте приложения для управления рекламными кампаниями, контроля бюджета, получения отчетов.</p>	<p><a href="#">«Поделиться» в соцсетях</a> Добавьте на сайт или в блог кнопки для быстрой публикации ссылок на ваш контент.</p>
<p><a href="#">Yandex Map Kit</a> Встраивайте Яндекс.Карты в ваши мобильные приложения для платформ Android, iOS и Windows Phone.</p>	<p><a href="#">API Яндекс.Метрики</a> Управляйте счетчиками, анализируйте поведение пользователей и эффективность рекламных кампаний.</p>	<p><a href="#">API Поиска по блогам</a> Организируйте поиск по своему блогу, блогохостингу или форуму. Создайте собственный рейтинг популярных записей.</p>
<p><a href="#">API Яндекс.Вебмастера</a> Создавайте приложения для работы с данными о сайтах из Яндекс.Вебмастера.</p>	<p><a href="#">API Яндекс.Денег</a> Организируйте оплату Яндекс.Деньгами на своем сайте или в приложении.</p>	<p><a href="#">API Яндекс.Диска</a> Храните данные и настройки ваших приложений на Яндекс.Диске и управляйте ими, используя протокол WebDAV.</p>
<p><a href="#">Яндекс.XML</a> Отправляйте запросы к поисковой базе Яндекса и получайте ответы в формате XML.</p>	<p><a href="#">API Яндекс.Маркета</a> Используйте данные Яндекс.Маркета на своем сайте и автоматизируйте управление ставками.</p>	<p><a href="#">API Яндекс.Почты для доменов</a> Подключите ваш домен к Яндекс.Почте для доменов и используйте широкие возможности Яндекс.Почты.</p>
<p><a href="#">API Яндекс.Фоток</a> Создавайте приложения для загрузки и редактирования фотографий, следите за обновлениями в альбомах пользователей.</p>	<p><a href="#">API Яндекс.Услуг</a> Предложите вашим пользователям возможность подбора наиболее выгодных вкладов и кредитов.</p>	<p><a href="#">API Яндекс.Чистый Веб</a> API предоставляет интерфейс к фильтрам Спамобороны Яндекса.</p>
<p><a href="#">API Яндекс.Бара</a> Создайте свой компонент для панели Яндекс.Бара и предложите его миллионам пользователей.</p>	<p><a href="#">Яндекс.Детектор</a> Определяйте модель мобильного устройства по заголовкам HTTP-запросов браузера.</p>	<p><a href="#">API Яндекс.Подписок</a> Создавайте приложения для чтения и работы с RSS-потоками.</p>
<p><a href="#">Виджетная платформа</a> Создавайте виджеты для главной страницы Яндекса и предлагайте их пользователям.</p>	<p><a href="#">Яндекс.Локатор</a> Определяйте местоположение мобильного устройства по сигналам сетей сотовой</p>	<p><a href="#">API Моего Круга</a> Используйте возможности МоегоКруга на вашем сайте и создавайте на их основе свои приложения.</p>

	связи и точек доступа Wi-Fi.	
<a href="#">Яндекс.Спеллер</a> Используйте проверку орфографии в HTML-формах, текстовых редакторах и CMS.	<a href="#">Safe Browsing API</a> Проверяйте ссылки и страницы перед тем, как показывать их своим пользователям.	<a href="#">API Я.ру</a> Создавайте приложения для чтения ленты друзей, ведения блога и размещения контента.
<a href="#">API Перевода</a> Используйте автоматический перевод текстов от Яндекса.		<a href="#">Rich Content API</a> Формируйте превью страниц с помощью данных из контент-системы Яндекс.Поиска.

Рассмотрим пример использования предлагаемых API. Познакомимся с функцией перевода текстов, используя Yandex API.

## Метод translate

Осуществляет перевод текста.

### Описание

Translation `translate(string lang, string text[], string format);`

Страница для демонстрации работы метода: <http://translate.yandex.net/api/v1/tr/?op=translate>

### Входные параметры

Входные параметры могут передаваться либо с помощью HTTP GET-запроса, либо с помощью HTTP POST-запроса, где параметры передаются в body HTTP-запроса.

### Пример запроса:

XML-интерфейс:

<http://translate.yandex.net/api/v1/tr/translate?lang=en-ru&text=To+be,+or+not+to+be%3F&text=That+is+the+question.>

JSON-интерфейс:

<http://translate.yandex.net/api/v1/tr.json/translate?lang=en-ru&text=To+be,+or+not+to+be%3F&text=That+is+the+question.>

JSONP-интерфейс (для функции myCallback):

<http://translate.yandex.net/api/v1/tr.json/translate?lang=en-ru&text=To+be,+or+not+to+be%3F&text=That+is+the+question.&callback=myCallback>

Ниже приводится полный список входных параметров.

Параметр	Тип	Описание
<b>Обязательные</b>		
lang	string	Направление перевода (например, "en-ru" или "ru"). Задается в виде: <ol style="list-style-type: none"> <li>1. Пары кодов языков, перечисленных через дефис. Например, "en-ru" задает перевод с английского на русский.</li> <li>2. Одиночного кода языка. Например, "ru" задает перевод на русский язык. В этом случае язык исходного текста определяется автоматически.</li> </ol>
text	string[]	Текст, который требуется перевести. Ограничения: <ul style="list-style-type: none"> <li>• Для POST-запросов максимальный размер передаваемого текста составляет 10000 символов.</li> <li>• В GET-запросах ограничивается не размер передаваемого текста, а размер всей строки запроса, которая кроме текста может</li> </ul>

		содержать и другие параметры. Максимальный размер строки запроса - 10Кб.
<b>Необязательные</b>		
format	string	Формат текста. Возможные значения: <ul style="list-style-type: none"> <li>• plain - текст без разметки (значение по умолчанию);</li> <li>• html - текст в формате HTML.</li> </ul>

### Возвращает

В XML-интерфейсе возвращает структуру Translation, содержащую переведенный текст.

Например:

```
<?xml version="1.0" encoding="utf-8"?>
<Translation code="200" lang="en-ru">
  <text>Быть или не быть?</text>
  <text>Вот в чем вопрос.</text>
</Translation>
```

Элементы XML-схемы ответа:

- Translation — корневой элемент, содержит элементы text (по одному для каждого фрагмента). Атрибут code содержит код возврата (см. таблицу), атрибут lang - код языка фрагмента, определенный сервисом.
- text — массив строк с переведенным текстом (для XML-интерфейса — последовательность элементов <text>). Количество элементов в массиве совпадает с числом параметров text в запросе.

### Коды возврата

Код	Значение	Описание
ERR_OK	200	Операция выполнена успешно.
ERR_TEXT_TOO_LONG	413	Превышен максимальный размер текста.
ERR_UNPROCESSABLE_TEXT	422	Текст не может быть переведен.
ERR_LANG_NOT_SUPPORTED	501	Заданное направление перевода не поддерживается.

В JSON-интерфейсе вместо XML-элементов возвращаются JavaScript-объекты с теми же именами и семантикой:

```
{
  "code": 200,
  "lang": "en-ru",
  "text": [
    "Быть или не быть?",
    "Вот в чем вопрос."
  ]
}
```

В JSONP-интерфейсе те же самые JavaScript-объекты возвращаются в callback-функции (например, myCallback):

```
myCallback({
  "code": 200,
  "lang": "en-ru",
  "text": [
    "Быть или не быть?",
    "Вот в чем вопрос."
  ]
})
```

```
<html xmlns="http://www.w3.org/1999/xhtml" >
<head>
  <title>Untitled Page</title>
<SCRIPT type="text/javascript">
```

```

function GoTranslate() {
    var lang_ = document.formsearch.lang_.value;
    var trans_words = document.formsearch.trans_words.value;

    var FullSearchUrl = "http://translate.yandex.net/api/v1/tr.json/translate?lang="+
        lang_ +
        "&text=" + trans_words +
        "&callback=myCallback";
//var FullSearchUrl =
"http://translate.yandex.net/api/v1/tr/translate?lang="+lang_+"&text="+trans_words;

    location.href = FullSearchUrl;

}

</SCRIPT>

</head>
<body>

<FORM NAME="formsearch" action="">
    Перевод:
    <br>
    язык оригинала-перевода: <INPUT TYPE="text" NAME="lang_" SIZE="40">
    <br>
    текст для перевода: <INPUT TYPE="text" NAME="trans_words" SIZE="40">
    <br>
    <INPUT TYPE="button" VALUE="Искать" onClick="GoTranslate()">

</FORM>
</body>
</html>

```

**Задание 1:** Создать приведенный пример скрипта в составе HTML-документа. Проверить работу, проверить варианты перевода с немецкого на русский, с французского на русский, с русского на английский.

### Индивидуальное задание 1:

**Вариант 1:** Разобрать функционал API поиска по блогам, начиная с адреса: <http://api.yandex.ru/blogs/doc/search/reference/get-parameters-text.xml>

Разработать страницу по аналогии с примером задания 1 для ввода следующих параметров поиска по блогам:

- искомый текст
- Область поиска (с проверкой ввода одного из вариантов: **popular, blog, community, personal, comments, forum, all**)
- Пол, возраст и место жительства

**Вариант 2:** Разобрать функционал API поиска по блогам, начиная с адреса: <http://api.yandex.ru/blogs/doc/search/reference/get-parameters-text.xml>

Разработать страницу по аналогии с примером задания 1 для ввода следующих параметров поиска по блогам:

- искомый текст
- Область поиска (с проверкой ввода одного из вариантов: **popular, blog, community, personal, comments, forum, all**)
- дата начала и конца публикации

**Вариант 3:** Разобрать функционал API поиска по блогам, начиная с адреса: <http://api.yandex.ru/blogs/doc/search/reference/get-parameters-text.xml>  
Разработать страницу по аналогии с примером задания 1 для ввода следующих параметров поиска по блогам:

- искомый текст
- Область поиска (с проверкой ввода одного из вариантов: **popular, blog, community, personal, comments, forum, all**)
- ССЫЛКИ

**Вариант 4:** Разобрать функционал API поиска по блогам, начиная с адреса: <http://api.yandex.ru/blogs/doc/search/reference/get-parameters-text.xml>  
Разработать страницу по аналогии с примером задания 1 для ввода следующих параметров поиска по блогам:

- искомый текст
- Область поиска (с проверкой ввода одного из вариантов: **popular, blog, community, personal, comments, forum, all**)
- Записи и комментарии одного блоггера

## Пример работы с Yandex API Maps

```
<!DOCTYPE html>

<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title>Примеры. Размещение карты на странице.</title>
  <!--
    Подключаем API карт
    Параметры:
      - load=package.standard - основные компоненты;
      - lang=ru-RU - язык русский.
  -->
  <script src="http://api-maps.yandex.ru/2.0/?load=package.standard&lang=ru-RU"
type="text/javascript"></script>

<script type="text/javascript">
  var myMap;

  // Дождёмся загрузки API и готовности DOM.
  ymaps.ready(init);

  function init() {
    // Создание экземпляра карты и его привязка к контейнеру с
    // заданным id ("map").
    myMap = new ymaps.Map('map', {
      // При инициализации карты обязательно нужно указать
      // её центр и коэффициент масштабирования.
      center: [55.76, 37.64], // Москва
      zoom: 10
    });

    document.getElementById('destroyButton').onclick = function () {
      // Для уничтожения используется метод destroy.
      myMap.destroy();
    };
  }
}
```

```

</script>
</head>

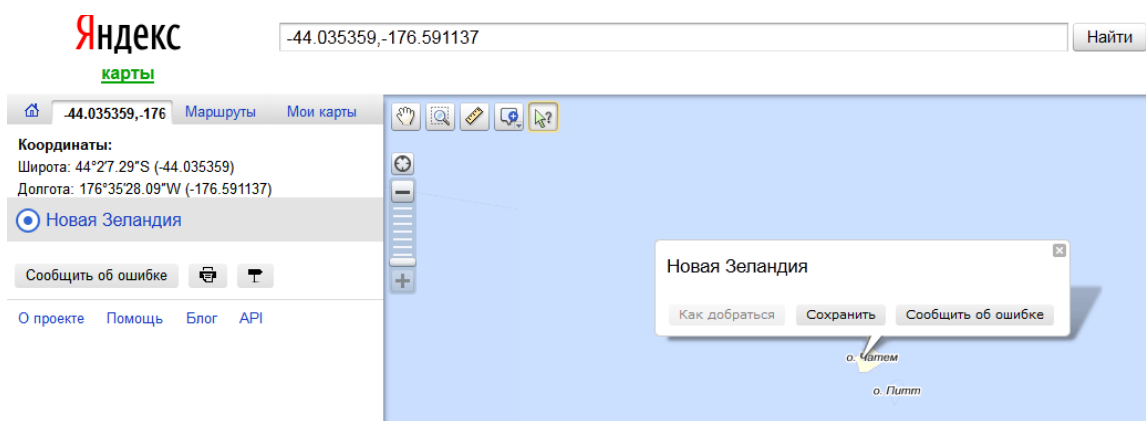
<body>
  <div id="map" style="width:400px; height:300px"></div>
  <input type="button" id="destroyButton" value="Удалить карту"/>
</body>

</html>

```

**Задание 2:** Создать приведенный пример скрипта в составе HTML-документа. Проверить работу.

## Примеры. Добавление прямоугольника на карту



```

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>Примеры. Добавление прямоугольника на карту.</title>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>

  <!--
    Подключаем API карт 2.x
    Параметры:
    - load=package.full - полная сборка;
    - lang=ru-RU - язык русский.
  -->
  <script src="http://api-maps.yandex.ru/2.0/?load=package.full&lang=ru-RU"
    type="text/javascript"></script>
  <script type="text/javascript">
    // Как только будет загружен API и готов DOM, выполняем инициализацию
    ymaps.ready(init);

    function init() {
      var myMap = new ymaps.Map('map', {
        center: [-44.03, -176.64],
        zoom: 6
      }),
      // Создаем первый прямоугольник
      myRect = new ymaps.Rectangle([
      // Задаем координаты диагональных углов прямоугольника
        [-43.76, -176.60],
        [-43.99, -176.69]
      ]),

      // Альтернативный способ создания прямоугольника:
      // создаем геообъект с типом геометрии "прямоугольник".
      myGeoObject = new ymaps.GeoObject({

```

```

        // Задаем модель геообъекта.
        // Модель = геометрия + свойства геообъекта.
        // Геометрия = тип геометрии + координаты геообъекта.
        geometry: {
            // Тип геометрии - прямоугольник
            type: "Rectangle",
            // Координаты
            coordinates: [
                [-43.73, -176.46],
                [-43.99, -176.53]
            ]
        }
    });

    myMap.geoObjects.add(myRect)
        .add(myGeoObject);
}
</script>
</head>

<body>
<h2>Добавление прямоугольника на карту</h2>

<div id="map" style="width:600px;height:400px"></div>
</body>

</html>

```

**Задание 3:** Выполните пример, посмотрите работу. Далее просмотрите примеры работы с картами по адресу: <http://api.yandex.ru/maps/doc/jsapi/2.x/examples/#basic>

### Индивидуальное задание 2:

**Вариант 1:** Разобрать функционал API карты по примерам. Создать код, который отображал бы карты в масштабе 8 с центром карты на «Бразилия, Рио-де-Жанейро». И рисовал бы круг вокруг города.

**Вариант 2:** Разобрать функционал API карты по примерам. Создать код, который отображал бы карты в масштабе 10 с центром карты на «Парагвай, Асунсьон». И рисовал бы шестигранник вокруг города.

**Вариант 3:** Разобрать функционал API карты по примерам. Создать код, который отображал бы карты в масштабе 8 с центром карты на «Намибия, Виндхук». И рисовал бы квадрат вокруг города.

**Вариант 4:** Разобрать функционал API карты по примерам. Создать код, который отображал бы карты в масштабе 8 с центром карты на «Юкон-Коюкек, Аляска, Соединенные Штаты Америки». И рисовал бы треугольник вокруг города.

## Отчет по лабораторной работе

В соответствии со структурой заготовки отчета и примером оформления оформить в отчете все задания, выполняемые в ходе лабораторной работы, а также индивидуальные задания по вариантам. Файл с отчетом называть по шаблону: **Фамилия\_лаб\_раб\_номер**.

Отчет предоставляется в электронном виде либо лично преподавателю, либо на электронную почту для проверки. Также по результатам лабораторной работы на следующем за ней занятии проводится выборочный опрос по командам языка.