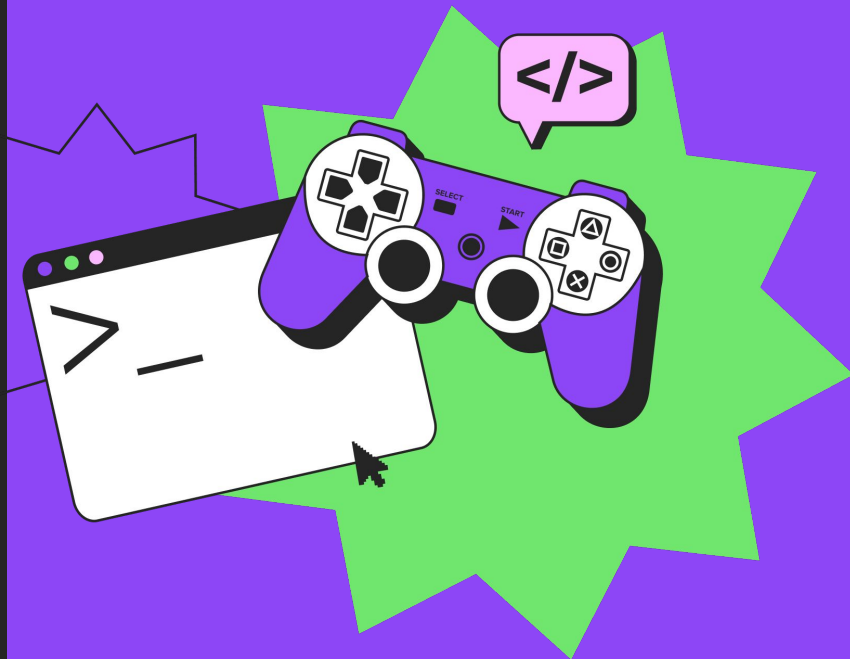


JSON и XML – основные форматы передачи данных

Семинар 4
Знакомство с веб-технологиями





Знакомство с веб-технологиями

1

Лекция 1.
Веб-технологии: вчера,
сегодня, завтра

2

Семинар 1: Веб-
технологии: вчера,
сегодня, завтра

3

Лекция 2.
CSS и HTML – что за звери?

4

Семинар 2.
CSS и HTML – что за звери?

5

Лекция 3. Основы
JavaScript для начинающих
разработчиков

6

Семинар 3. Основы
JavaScript для начинающих
разработчиков

7

Лекция 4. Основы бэкенда:
PHP и MySQL

8

Семинар 4. Основы
бэкенда: PHP и MySQL

9

Лекция 5. JSON и XML

10

Семинар 5. JSON и XML



Что будет на уроке сегодня

- 📌 Получим представление о SOAP и REST.
- 📌 Научимся читать и создавать JSON.
- 📌 Научимся читать и создавать XML.





Вопросы?



Вопросы?



Вопросы?





Викторина



SOAP — это...

1. стандартизированный протокол, который отправляет сообщения с использованием других протоколов, таких как HTTP и др.
2. одновременно и препроцессор гипертекста (HTML) и серверный язык программирования, и скриптовый интерпретируемый язык.
3. не протокол, а архитектурный стиль.



SOAP — это...

1. стандартизированный протокол, который отправляет сообщения с использованием других протоколов, таких как HTTP и др.
2. одновременно и препроцессор гипертекста (HTML) и серверный язык программирования, и скриптовый интерпретируемый язык.
3. не протокол, а архитектурный стиль.



REST — это...

1. не протокол, а архитектурный стиль.
2. стандартизированный протокол, который отправляет сообщения с использованием других протоколов, таких как HTTP и др.
3. одновременно и препроцессор гипертекста (HTML) и серверный язык программирования, и скриптовый интерпретируемый язык.



REST — это...

1. не протокол, а архитектурный стиль.
2. стандартизированный протокол, который отправляет сообщения с использованием других протоколов, таких как HTTP и др.
3. одновременно и препроцессор гипертекста (HTML) и серверный язык программирования, и скриптовый интерпретируемый язык.



XML — это...

1. расширяемый язык разметки. Используется для хранения и передачи данных.
2. стандартизированный протокол, который отправляет сообщения с использованием других протоколов, таких как HTTP и др.
3. одновременно и препроцессор гипертекста (HTML) и серверный язык программирования, и скриптовый интерпретируемый язык.



XML — это...

1. расширяемый язык разметки. Используется для хранения и передачи данных.
2. стандартизированный протокол, который отправляет сообщения с использованием других протоколов, таких как HTTP и др.
3. одновременно и препроцессор гипертекста (HTML) и серверный язык программирования, и скриптовый интерпретируемый язык.



Будет ли данный XML код well formed?

```
1 <req>
2 <query>мск сухонская 11</query>
3 <count>7
4 </req>
5 </count>
```

1. да
2. нет
3. в зависимости от того, какой стиль API.



Будет ли данный XML код well formed?

```
1 <req>
2 <query>мск сухонская 11</query>
3 <count>7
4 </req>
5 </count>
```

1. да
2. **нет**
3. в зависимости от того, какой стиль API.



JSON — это...

1. текстовый формат обмена данными, основанный на JavaScript.
2. расширяемый язык разметки. Используется для хранения и передачи данных.
3. стандартизированный протокол, который отправляет сообщения с использованием других протоколов, таких как HTTP и др.



JSON — это...

1. **текстовый формат обмена данными, основанный на JavaScript.**
2. расширяемый язык разметки. Используется для хранения и передачи данных.
3. стандартизированный протокол, который отправляет сообщения с использованием других протоколов, таких как HTTP и др.



Будет ли данный JSON код well formed?

```
1 {  
2   "query": "мск сухонская 11"  
3   "count": 7  
4 }
```

1. нет
2. да
3. в зависимости от того, какой стиль API.



Будет ли данный JSON код well formed?

```
1 {  
2   "query": "мск сухонская 11"  
3   "count": 7  
4 }
```

1. **нет**
2. да
3. в зависимости от того, какой стиль API.



Вопросы?



Вопросы?



Вопросы?





Практика



Задание 1

Задача: с помощью онлайн-сервиса <https://tools.seo-zona.ru/postget.html>

- Отправьте **GET** запрос на ресурс <https://nalog.ru>.
- Отправьте **POST** запрос (имя: text, значение: котики) на ресурс <https://yandex.ru/search/>.

Объясните разницу между методами и полученными результатами.



10 минут



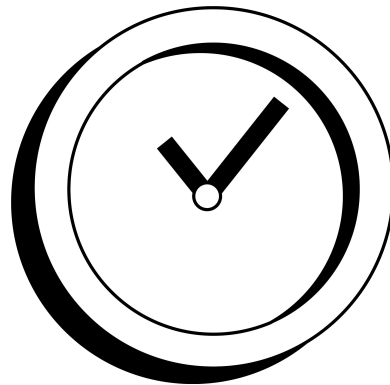
Задание 1

Задача: с помощью онлайн-сервиса <https://tools.seo-zona.ru/postget.html>

- Отправьте **GET** запрос на ресурс <https://nalog.ru>.
- Отправьте **POST** запрос (имя: text, значение: котики) на ресурс <https://yandex.ru/search/>.

Объясните разницу между методами и полученными результатами.

<<10:00-





Задание 1

Куда (адрес) и какие запросы:

Метод: Цель:

Способ кодирования: Повторять каждые: секунд

Имя:

Значение:

Куда (адрес) и какие запросы:

Метод: Цель:

Способ кодирования: Повторять каждые: секунд

Имя:

Значение:



Задание 2

Задача: создать XML-документ на основе каталога книг для сайта, который содержит следующие данные:

Название	Автор	Количество страниц	Дата издания	Цена, руб
Камил видит руками	Томаш Малковски	146	2019	863
Вероятности и неприятности	Сергей Самойленко	134	2022	1055
Когда жизнь сбивает с ног	Расс Хэррис	80	2021	945
Цветы в темноте	Сестра Данг Нгием	124	2020	1027
Стратегии перемен	Джон Коттер	102	2022	1418



Для выполнения задания и демонстрации результата используйте онлайн-редактор <https://jsonformatter.org/xml-formatter>

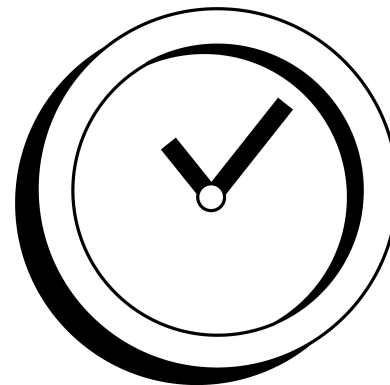


Задание 2

Задача: создать XML-документ на основе каталога книг для сайта, который содержит следующие данные:

Название	Автор	Количество страниц	Дата издания	Цена, руб
Камил видит руками	Томаш Малковски	146	2019	863
Вероятности и неприятности	Сергей Самойленко	134	2022	1055
Когда жизнь сбивает с ног	Расс Хэррис	80	2021	945
Цветы в темноте	Сестра Данг Нгием	124	2020	1027
Стратегии перемен	Джон Коттер	102	2022	1418

<<15:00-



Для выполнения задания и демонстрации результата используйте онлайн-редактор <https://jsonformatter.org/xml-formatter>



Задание 2

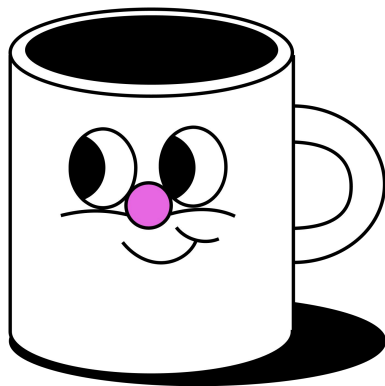
Задача: создать XML-документ на основе каталога книг для сайта.

Пример, на вашем экране →

```
1 <?xml version="1.0"?>
2 <CATALOG>
3   <BOOK>
4     <TITLE>Камил видит руками</TITLE>
5     <AUTHOR>Томаш Малковски</AUTHOR>
6     <PAGES>146</PAGES>
7     <YEAR>2019</YEAR>
8     <PRICE>863 руб.</PRICE>
9   </BOOK>
10  <BOOK>
11    <TITLE>Вероятности и неприятности</TITLE>
12    <AUTHOR>Сергей Самойленко</AUTHOR>
13    <PAGES>134</PAGES>
14    <YEAR>2022</YEAR>
15    <PRICE>1055 руб.</PRICE>
16  </BOOK>
17  <BOOK>
18    <TITLE>Когда жизнь сбивает с ног</TITLE>
19    <AUTHOR>Расс Хэррис</AUTHOR>
20    <PAGES>80</PAGES>
21    <YEAR>2021</YEAR>
22    <PRICE>945 руб.</PRICE>
23  </BOOK>
24  <BOOK>
25    <TITLE>Цветы в темноте</TITLE>
26    <AUTHOR>Сестра Данг Нгием</AUTHOR>
27    <PAGES>124</PAGES>
28    <YEAR>2020</YEAR>
29    <PRICE>1027 руб.</PRICE>
30  </BOOK>
31  <BOOK>
32    <TITLE>Стратегии перемен</TITLE>
33    <AUTHOR>Джон Коттер</AUTHOR>
34    <PAGES>102</PAGES>
35    <YEAR>2022</YEAR>
36    <PRICE>1418 руб.</PRICE>
37  </BOOK>
38 </CATALOG>
```



Перерыв



<<5:00->>



Задание 3

Задача: создать JSON-документ на основе каталога продукции для интернет-магазина, который содержит следующие данные:

Название товара	Вес упаковки, гр	Товарная группа	Цена, руб	Номер склада
Яблоки	1000	Фрукты	150	1
Картошка ранняя	5000	Овощи	320	2
Бананы	1000	Фрукты	99	1
Баунти мини	100	Кондитерские изделия	180	3

[Файл с таблицей](#) (*.txt)

Для выполнения задания и демонстрации результата используйте онлайн-редактор <https://jsonformatter.org/json-parser>

15 минут



Задание 3

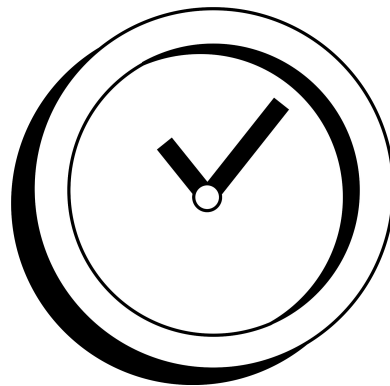
Задача: создать JSON-документ на основе каталога продукции для интернет-магазина, который содержит следующие данные:

Название товара	Вес упаковки, гр	Товарная группа	Цена, руб	Номер склада
Яблоки	1000	Фрукты	150	1
Картошка ранняя	5000	Овощи	320	2
Бананы	1000	Фрукты	99	1
Баунти мини	100	Кондитерские изделия	180	3

[Файл с таблицей](#) (*.txt)

Для выполнения задания и демонстрации результата используйте онлайн-редактор <https://jsonformatter.org/json-parser>

<<15:00-





Задание 3

Задача: создать JSON-документ на основе каталога продукции для интернет-магазина, который содержит следующие данные:

[Пример, на вашем экране →](#)

```
1 {"catalog": [  
2   {  
3     "name": "Яблоки",  
4     "weight": 1000,  
5     "group": "Фрукты",  
6     "price": 150,  
7     "number_store": 1  
8   },  
9   {  
10    "name": "Картошка ранняя",  
11    "weight": 5000,  
12    "group": "Овощи",  
13    "price": 320,  
14    "number_store": 2  
15  },  
16  {  
17    "name": "Бананы",  
18    "weight": 1000,  
19    "group": "Фрукты",  
20    "price": 99,  
21    "number_store": 1  
22  },  
23  {  
24    "name": "Баунти мини",  
25    "weight": 100,  
26    "group": "Кондитерские изделия",  
27    "price": 180,  
28    "number_store": 3  
29  }  
30 ]  
31 }
```



Задание 4

Задача: проверить на Well formed

XML

```
1 <?xml version="1.0"?>
2 <doRegister>
3   <email>olga@gmail.com</email>
4   <name>Ольга</name>
5   <password>1
6 </doRegister>
7 </password>
8 <doRegister>
9   <email>ivan@gmail.com</email>
10  <name>Иван</name>
11  <password>222</password>
12 </doRegister>
```

JSON

```
1 {
2   "query": "Виктор Иван"
3   "count": 7
4   "parts": ("NAME", "SURNAME")
5 }
```

5 минут



Задание 4

Задача: проверить на Well formed

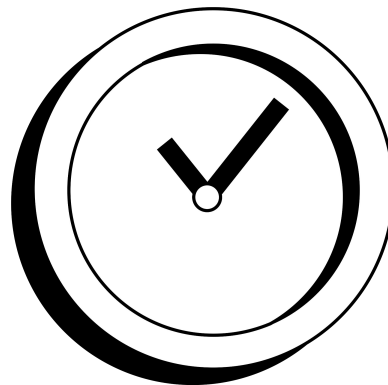
XML

```
1 <?xml version="1.0"?>
2 <doRegister>
3   <email>olga@gmail.com</email>
4   <name>Ольга</name>
5   <password>1
6 </doRegister>
7 </password>
8 <doRegister>
9   <email>ivan@gmail.com</email>
10  <name>Иван</name>
11  <password>222</password>
12 </doRegister>
```

JSON

```
1 {
2   "query": "Виктор Иван"
3   "count": 7
4   "parts": ("NAME", "SURNAME")
5 }
```

<<05:00-





Задание 4

Задача: проверить на Well formed

XML

```
1 <?xml version="1.0"?>
2 <doRegister>
3   <email>olga@gmail.com</email>
4   <name>Ольга</name>
5   <password>1
6 </doRegister>
7 </password>
8 <doRegister>
9   <email>ivan@gmail.com</email>
10  <name>Иван</name>
11  <password>222</password>
12 </doRegister>
```

```
1 <?xml version="1.0"?>
2 <users>
3   <doRegister>
4     <email>olga@gmail.com</email>
5     <name>Ольга</name>
6     <password>1</password>
7   </doRegister>
8   <doRegister>
9     <email>ivan@gmail.com</email>
10    <name>Иван</name>
11    <password>222</password>
12  </doRegister>
13 </users>
```

JSON

```
1 {
2   "query": "Виктор Иван"
3   "count": 7
4   "parts": ("NAME", "SURNAME")
5 }
```

```
1 {
2   "query": "Виктор Иван",
3   "count": 7,
4   "parts": ["NAME", "SURNAME"]
5 }
```




Задание 5

Задача: создать XML-документ на основе списка учащихся, который содержит следующие данные:

ФИО	Дата рождения	Класс	Адрес прописки
Карпова Анна Александровна	2008	8А	г. Москва, Ленинский проспект, д. 2, кв. 277
Мальшкин Михаил Иванович	2014	2Б	г. Москва, Ленградское шоссе, д. 10, кв. 56
Андреева Алла Петровна	2015	1Г	г. Москва, Дмитровское шоссе, д. 107, кв. 2

[Файл с таблицей](#) (*.txt)

Для выполнения задания и демонстрации результата используйте онлайн-редактор <https://jsonformatter.org/xml-formatter>

15 минут



Задание 5

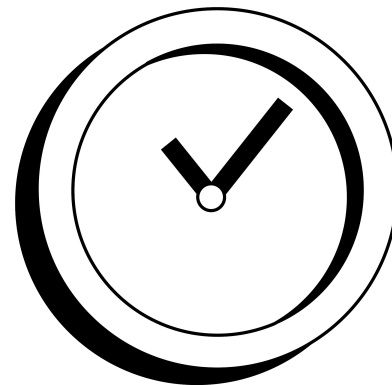
Задача: создать XML-документ на основе списка учащихся, который содержит следующие данные:

ФИО	Дата рождения	Класс	Адрес прописки
Карпова Анна Александровна	2008	8А	г. Москва, Ленинский проспект, д. 2, кв. 277
Мальшкин Михаил Иванович	2014	2Б	г. Москва, Ленградское шоссе, д. 10, кв. 56
Андреева Алла Петровна	2015	1Г	г. Москва, Дмитровское шоссе, д. 107, кв. 2

[Файл с таблицей](#) (*.txt)

Для выполнения задания и демонстрации результата используйте онлайн-редактор <https://jsonformatter.org/xml-formatter>

<<15:00-





Задание 5

Задача: создать XML-документ на основе списка учащихся, который содержит следующие данные

Пример, на вашем экране →

```
1 <?xml version="1.0"?>
2 <School>
3   <Student>
4     <Name>
5       <FirstName>Карпова</FirstName>
6       <Patronymic>Анна</Patronymic>
7       <LastName>Александровна</LastName>
8     </Name>
9     <Birthdate>2008</Birthdate>
10    <ClassNo>8А</ClassNo>
11    <Address>
12      <City>Москва</City>
13      <Street>Ленинский проспект</Street>
14      <House>2</House>
15      <Apartment>277</Apartment>
16    </Address>
17  </Student>
18  <Student>
19    <Name>
20      <FirstName>Малышкин</FirstName>
21      <Patronymic>Михаил</Patronymic>
22      <LastName>Иванович</LastName>
23    </Name>
24    <Birthdate>2014</Birthdate>
25    <ClassNo>2Б</ClassNo>
26    <Address>
27      <City>Москва</City>
28      <Street>Ленинский проспект</Street>
29      <House>10</House>
30      <Apartment>56</Apartment>
31    </Address>
32  </Student>
33  <Student>
34    <Name>
35      <FirstName>Андреева</FirstName>
36      <Patronymic>Алла</Patronymic>
37      <LastName>Петровна</LastName>
38    </Name>
39    <Birthdate>2015</Birthdate>
40    <ClassNo>1Г</ClassNo>
41    <Address>
42      <City>Москва</City>
43      <Street>Дмитровское шоссе</Street>
44      <House>107</House>
45      <Apartment>2</Apartment>
46    </Address>
47  </Student>
48 </School>
```



Задание 6

Задача: создать JSON-документ на основе данных о пользователе для сайта, который содержит следующие данные:

Имя	Фамилия	Возраст	Увлечения	Женат/ Замужем	Дети
Иван	Иванов	37	рыбалка, охота, футбол	да	Мария, Николай
Ольга	Фролова	21	путешествия, вышивание	нет	

[Файл с таблицей](#) (*.txt)

Для выполнения задания и демонстрации результата используйте онлайн-редактор <https://jsonformatter.org/json-parser>

15 минут



Задание

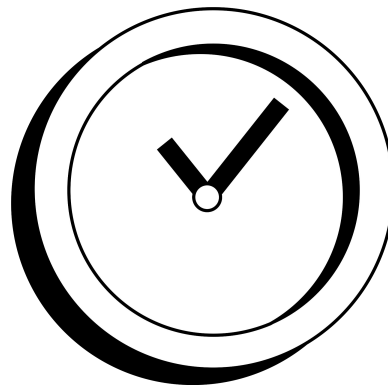
Задача: создать JSON-документ на основе данных о пользователе для сайта, который содержит следующие данные:

Имя	Фамилия	Возраст	Увлечения	Женат/ Замужем	Дети
Иван	Иванов	37	рыбалка, охота, футбол	да	Мария, Николай
Ольга	Фролова	21	путешествия, вышивание	нет	

[Файл с таблицей](#) (*.txt)

Для выполнения задания и демонстрации результата используйте онлайн-редактор <https://jsonformatter.org/json-parser>

<<15:00-





Задание 5

Задача: создать JSON-документ на основе данных о пользователе для сайта, который содержит следующие данные:

Пример, на вашем экране →

```
1 {"users": [  
2   {  
3     "first_name": "Иван",  
4     "last_name": "Иванов",  
5     "age": 37,  
6     "hobbies": [  
7       "рыбалка",  
8       "охота",  
9       "футбол"  
10    ]},  
11    "married": true,  
12    "children": [  
13      "Мария",  
14      "Николай"  
15    ]  
16  },  
17  {  
18    "first_name": "Ольга",  
19    "last_name": "Фролова",  
20    "age": 21,  
21    "hobbies": [  
22      "путешествия",  
23      "вышивание"  
24    ]},  
25    "married": false,  
26    "children": null  
27  }  
28 ]  
29 }
```



Вопросы?



Вопросы?



Вопросы?





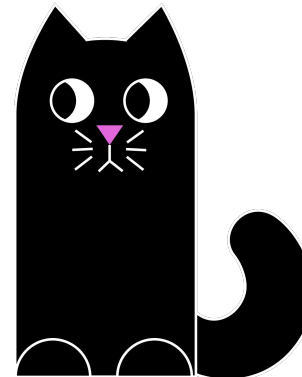
Домашнее задание



Домашнее задание

Задача №1: проверить XML на Well formed

```
1 <req>
2   <surname>Иванов</surname>
3
4   <name>Иван</name>
5
6   <patronymic>Иванович</patronymic>
7
8   <birthdate>01.01.1990</birthdate>
9
10  <birthplace>Москва</birthplace>
11
12  <phone>8 926 766 48 48</phone>
13 </req>
```

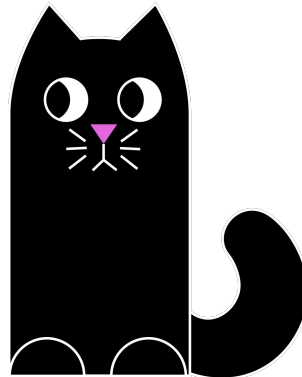




Домашнее задание

Задача №2: проверить JSON на Well formed

```
1 {  
2   "surname": "Иванов"  
3  
4   "name": "Иван"  
5  
6   "patronymic": "Иванович"  
7  
8   "birthdate": "01.01.1990"  
9  
10  "birthplace": "Москва"  
11  
12  "phone": "8 926 766 48 48"  
13 }
```





Спасибо за внимание!