

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Озерский технологический институт — филиал НИЯУ МИФИ

Кафедра прикладной математики

Вл. Пономарев

ПРАКТИКУМ

по программированию web-приложений

Учебно-методическое пособие

Часть 3. Введение в XML

2018 г.

УДК 681.3.06
П 56

Вл. Пономарев. Практикум по программированию web-приложений. Учебно-методическое пособие. Часть 3. Введение в XML. Озерск: ОТИ НИЯУ МИФИ, 2018. — 9 с.

В пособии описываются задания практических работ по дисциплине «Современные технологии программирования». Работы третьего семестра изучения дисциплины включают в себя основы программирования web-приложений с использованием языков программирования JavaScript, PHP, XPATH, XSLT и объектной модели документа DOM.

В качестве основного материала при выполнении практических работ пособие предназначено для студентов, обучающихся по направлению подготовки 09.03.01 «Информатика и вычислительная техника» и специальности 09.05.01 «Применение и эксплуатация автоматизированных систем специального назначения».

Рецензенты:

- 1.
- 2.

УТВЕРЖДЕНО
Редакционно-издательским
Советом ОТИ НИЯУ МИФИ

Содержание

Общие цели занятий.....	4
1. Работа XML-01. Введение в XML-документы.....	5
1.1. Тестовый XML-документ	5
1.2. Проверка правильности XML-документа.....	6
1.3. XML DTD	7
1.4. XML Schema	7
1.5. Установка пакета Denwer	7
1.6. Сценарий проверки XML-документа.....	8
2. Работа XML-02. Дерево объектов XML DOM.....	9
2.1. Установка пакета Denwer	9
2.2. Сценарий обхода дерева объектной модели	9

Общие цели занятий

В ходе практических работ изучается использование языков программирования JavaScript и PHP для формирования интерактивных web-страниц и web-приложений в целом.

Целью работ является создание приложения, моделирующего сетевую игру под названием «Королевский квадрат». Весь процесс создания этого приложения условно разбит на три логические части.

При этом первая часть работ в большей степени посвящена разработке алгоритмов игры на языке JavaScript в виде автономной страницы HTML, которая представляет собой законченное приложение.

Во второй части приложение переносится на сервер, и изучается взаимодействие серверной и клиентской частей кода с использованием языка PHP, а также технологий, связанных с представлением и обработкой информации, представленной в виде документа XML.

В третьей части основной задачей является исследование принципов интерактивной связи сервера с клиентом.

На выполнение каждой работы предположительно отводится от 2-х до 4-х академических часов. Успешное усвоение изучаемого материала возможно при условии, что студент самостоятельно находит необходимую техническую информацию, используя сеть Интернет, конспекты лекций и дополнительную литературу.

1. Работа XML-01. Введение в XML-документы

Цели:

- структура XML-документа;
- проверка состоятельности XML-документа.

Задачи:

- составление XML-документа;
- описание синтаксиса XML-документа при помощи DTD;
- описание синтаксиса XML-документа при помощи схем;
- проверка правильности и состоятельности XML-документов и схем.

Опорные документы:

Технологии программирования распределенных приложений. 2015.

Документ расположен на сайте преподавателя.

1.1. Тестовый XML-документ

Разработайте структуру тестового XML-документа в соответствии со своим порядковым номером в группе. Структура должна содержать не менее 3 уровней иерархии, не менее 5 разных тегов, не менее 3 разных типов данных (например: строковый, числовой, дата).

Документ должен иметь кодировку utf-8.

Названия тегов и их атрибутов должны быть русскими.

При помощи этой структуры опишите не менее трех объектов.

Сохраните документ под указанным именем.

Варианты заданий приведены ниже.

Вариант 1. Список книг для продажи books.xml.

Примерные характеристики книги: УДК, название книги, название издательства, множество авторов, год издания, количество страниц.

Вариант 2. Список квартир для продажи apartments.xml.

Примерные характеристики квартиры: общая площадь, количество комнат, размер кухни, размеры комнат, год постройки, признак совмещенного санузла.

Вариант 3. Список автомобилей для продажи autos.xml.

Примерные характеристики автомобиля: марка, год выпуска, мощность двигателя, количество ведущих колес, количество дверей, признак наличия ботового компьютера.

Примерные характеристики

Вариант 4. Учебное расписание группы shed.xml.

Примерные характеристики расписания: дата, номер пары, предмет, преподаватель, аудитория, время начала пары.

Вариант 5. Описание номеров отеля hotels.xml.

Примерные характеристики номера отеля: цена, класс обслуживания (А, Б или В), число комнат, наличие санузла, мини-бара, холодильника.

Вариант 6. Список компьютеров для продажи comps.xml.

Примерные характеристики компьютера: марка производителя, цена, марка процессора, тактовая частота процессора, объем ОЗУ, объем ЖД, название операционной системы, ее версия.

Вариант 7. Список учебных курсов studies.xml.

Примерные характеристики учебного курса: название, количество лекций, количество практик, дата начала, множество преподавателей.

Вариант 8. Описание стрелкового оружия guns.xml.

Примерные характеристики стрелкового оружия: марка, калибр, вес, объем обоймы, признак автоматического оружия.

Копии полученного документа должны использоваться для выполнения отдельных частей задания.

1.2. Проверка правильности XML-документа

После создания тестового документа он должен быть проверен на правильность. Используйте для этого один из сайтов, делающих такую проверку, например, сайт <http://www.utilities-online.info/xsdvalidation/>.

Текст документа скопируйте и вставьте в левое окно (рисунок 1), нажмите кнопку Check XML Well Formed.

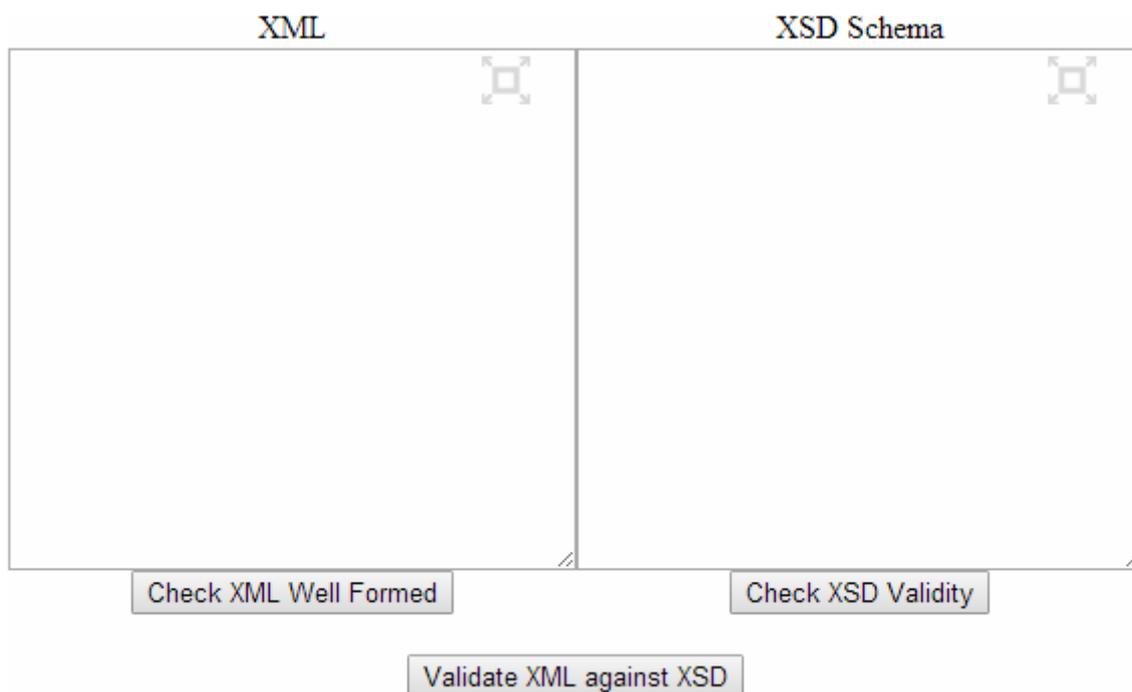


Рисунок 1 — Сайт для проверки XML-документов

Если XML-документ правильный, то в появившемся сообщении вы увидите слова «well formed».

Если этого не произошло, то нужно попытаться понять в сообщении, что неверно, исправить документ в окне, добиться его правильности, после чего скопировать исправленный документ из окна в свой файл.

1.3. XML DTD

Сохраните копию документа под именем `name-dtd.xml`, где `name` — название вашего XML-документа. Далее работайте с этой копией.

Опишите синтаксис документа в секции DOCTYPE.

Для справки используйте раздел «Секция DOCTYPE» опорного документа, а также ссылку на сайт <http://www.w3schools.com/schema/>.

После создания нового документа также убедитесь в его правильности так, как описано выше.

1.4. XML Schema

Сохраните копию документа `name-schema.xml`, где `name` — название вашего тестового XML-документа.

В этой копии укажите, что документ имеет схему. Для справки используйте раздел «XML-схемы» опорного документа и листинг 39. Название XML-документа схемы этого документа — `name.xsd`.

По окончании редактирования копии проверьте новый XML-документ на правильность.

Создайте новый XML-документ с названием `name.xsd`.

Опишите в этом документе схему своего тестового документа любым способом из описанных в разделе «XML-схемы» опорного документа.

После создания схемы скопируйте ее текст и вставьте в правое окно, после чего нажмите кнопку `Check XSD Validity` (рисунок 1). Если сообщение сайта указывает на ошибки, исправьте их в окне сайта, добейтесь правильности схемы, после чего скопируйте текст из окна сайта в свой документ.

После того, как документ с указанием схемы и схема будут правильными, нажмите кнопку `Validate XML against XSD`. Добейтесь того, чтобы документ соответствовал своей схеме.

1.5. Установка пакета Denwer

Скачайте с сайта преподавателя и установите пакет Denwer.

Виртуальный диск для пакета Denwer выберите Q.

Каталог для установки выберите `C:\WWW`.

Файл сценария назовите `valid.php`. Разместите файл сценария в каталоге `C:\WWW\home\localhost\www\xml`.

Первоначальное содержание файла следующее:

```
<?  
echo "Проверка XML-документа." ;  
>
```

Запустите Denwer при помощи ярлыка Start Denwer на рабочем столе, а в отсутствие ярлыка при помощи программы Run.exe, расположенной в каталоге C:\WWW\denwer\.

Важно! Перед уходом из аудитории остановите Denwer при помощи ярлыка Stop Denwer на рабочем столе, а в отсутствие ярлыка при помощи программы Stop.exe, расположенной в каталоге C:\WWW\denwer\.

Откройте браузер и введите адрес localhost/xml/valid.php.

Если все правильно, вы должны увидеть в браузере строку, заданную в команде echo.

1.6. Сценарий проверки XML-документа

Составьте сценарий php, проверяющий состоятельность документа.

Используйте для справки листинг 50 опорного документа, раздел «Загрузка и сохранение XML-документов».

2. Работа XML-02. Дерево объектов XML DOM

Цели:

- изучение объектной модели документа XML.

Задачи:

- чтение документа XML;
- построение объектной модели документа XML;
- вывод узлов дерева в браузер.

Опорный документ:

Технологии программирования распределенных приложений. 2015.

Документ расположен на сайте преподавателя.

2.1. Установка пакета Denwer

Проверьте, есть ли на компьютере пакет Denwer. Если нет, установите пакет так, как описано в предыдущей работе.

2.2. Сценарий обхода дерева объектной модели

Изучите раздел «Обход дерева объектной модели» опорного документа, листинг 53.

Если в документе нет ни одного внутреннего компонента, добавьте компонент в секцию DOCTYPE и вставьте хотя бы в один тег.

Если в документе нет ни одного комментария, добавьте в него один или несколько комментариев.

Создайте сценарий dom.php и опишите в нем процедуру обхода дерева объектной модели XML-документа.

Сценарий разместите в каталоге C:\WWW\home\localhost\www\dom.

Запустите Denwer, и введите в строку адреса localhost/dom/dom.php.