

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра «Информационные технологии»

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
ДЛЯ ВЫПОЛНЕНИЯ ЛАБОРАТОРНЫХ РАБОТ
ПО ДИСЦИПЛИНЕ
«ТЕХНОЛОГИИ WEB-ПРОГРАММИРОВАНИЯ»

Ростов-на-Дону
ДГТУ
2018

УДК 004.023, 004.43

Составитель М.В. Ступина

Методические указания для выполнения лабораторных работ по дисциплине «Технологии Web-программирования». – Ростов-на-Дону: Донской гос. техн. ун-т, 2018. – 10 с.

Рассмотрены вопросы разработки веб-страниц, веб-приложений и веб-служб с использованием современных технологий.

Содержат сведения о структуре дисциплины, ее содержании, а также рекомендации по изучению дисциплины.

Предназначены для обучающихся направления 44.03.04 Профессиональное обучение (по отраслям) для очной и заочной форм обучения.

УДК 004.023, 004.43

Печатается по решению редакционно-издательского совета
Донского государственного технического университета

Ответственный за выпуск зав. кафедрой «Информационные технологии»,
д-р техн. наук, профессор Б.В. Соболев

В печать 26.12. 2018 г.
Формат 60×84/16. Объем 0,6 усл.п.л.
Тираж 50 экз. Заказ № 2027.

Издательский центр ДГТУ
Адрес университета и полиграфического предприятия:
344000, г. Ростов-на-Дону, пл. Гагарина, 1

©Донской государственный
технический университет, 2018

Лабораторная работа №1 IP-АДРЕСАЦИЯ

Цель работы: Изучение структуры ip-адреса, основами протокола http и утилитами диагностики сетевых соединений.

Форма отчета: Демонстрация выполненного задания преподавателю.

Методические указания:

1. Повторить раздел лекционного материала по данной тематике.
2. Изучить возможности улитит.
3. Продемонстрировать пример работы преподавателю.

Задание 1.

1) С помощью утилиты *ipconfig* (запускается в командной строке командой *ipconfig*) определите IP-адрес и маску подсети для своего компьютера.

2) Определите класс подсети, в которой находится ваш компьютер без использования маски подсети и по маске подсети.

3) Определите адрес подсети, в которой находится ваш компьютер, с использованием функции “Логическое И” над IP-адресом и маской подсети. Следует иметь в виду, что операция “Логическое И” должна производиться с двоичным представлением операндов.

Задание 2.

С помощью утилиты *ping* (запускается в командной строке командой *ping*) проверьте доступность хостов, минимальное, среднее и максимальное время приема-передачи ICMP пакетов до них. Можно рассмотреть хосты, например в следующей последовательности:

- сервер вашего непосредственного провайдера или сервера вашей подсети;
- какой-либо сервер вашего региона;
- веб-сервер интернет-университета информационных технологий: www.intuit.ru;
- веб-сервер университета в кембридже: www.cam.ac.uk;
- веб-сервер университета в калифорнии: www.ucla.edu;
- веб-сервер университета в токио: www.u-tokyo.ac.jp;
- веб-сервер компании майкрософт: www.microsoft.com.

Задание 3.

С помощью утилиты *tracert* (запускается в командной строке командой *tracert*) определите маршруты следования и время прохождения пакетов до хостов, приведенных в задании 2.

Задание 4.

1. С помощью утилиты *netstat* (запускается в командной строке командой *netstat*) посмотрите активные текущие сетевые подключения и их состояние на вашем компьютере.

2. Запустите несколько экземпляров веб-браузера, загрузив в них веб-страницы с разных веб-серверов. Посмотрите с помощью *netstat*, какие новые сетевые подключения появились в списке.

3. Закрывайте браузеры и с помощью *netstat* проверяйте изменение списка сетевых подключений.

Задание 5.

1. Запустите сеанс *telnet* (запускается в командной строке командой *telnet*). При этом появится подсказка *Microsoft Telnet>*. С полным списком команд можно ознакомиться с помощью команды *help*.

2. Разрешите режим отображения вводимых с клавиатуры символов с помощью команды *set localecho*.

3. В соответствии с протоколом HTTP необходимо установить соединение с веб-сервером. Для этого с помощью команды *open* устанавливается соединение, например: *open www.yandex.ru 80*.

4. Сформируйте клиентский запрос. Как минимум он должен содержать строку состояния, например: *GET HTTP://WWW.YANDEX.RU/INDEX.HTML HTTP/1.0*

5. Изучите полученный ответ сервера. Обратите внимание на код ответа в строке состояния ответа веб-сервера в строке состояния и поля заголовка ответа. Если ответ сервера очень большой (в первую очередь из-за размера документа в теле ответа), то содержимое ответа сервера в окне интерпретатора командной строки обрезается с начала. В этом случае рекомендуется для просмотра заголовка вместо метода *GET* использовать метод *HEAD*.

Лабораторная работа №2 СЕТЕВОЕ ВЗАИМОДЕЙСТВИЕ В .NET

Цель работы: Изучение работы с сетью и выполнение низкоуровневых операций в .NET.

Форма отчета: Демонстрация выполненного задания преподавателю.

Методические указания:

1. Повторить раздел лекционного материала по данной тематике.
2. Изучить возможности классов.
3. Продемонстрировать пример работы преподавателю.

Задание 1.

Разберите на составляющие части (адрес, порт, протокол и т.д.) любой URL адрес и выполните обратное преобразование:

- создайте объект класса *uribuilder* для парсинга;
- выведите на консоль отдельно все компоненты переданного URL адреса;
- создайте объект класса *uribuilder* для построения адреса по отдельным элементам;
- выведите на консоль весь полученный адрес целиком.

Задание 2.

Получите DNS-имя вашего компьютера и определите его IP адрес:

- создайте строку с именем *dns*, содержащую dns имя вашего компьютера (используйте метод *gethostname()*);
- создайте строку с именем *ip*, содержащую ip адрес вашего компьютера (используйте свойство *addresslist* класса *iphostentry*);
- выведите на консоль полученные результаты.

Задание 3.

Реализуйте приложение, проверяющее сеть посредством использования классов *Ping*, *DNS*, *IPHostEntry*:

- создайте объект класса *ping* – отправитель;
- создайте объект класса *pingoptions* для передачи параметров;
- создайте строку, которую будете передавать и задайте *int* переменную – максимальное время (в миллисекундах) на ожидание ответа;
- создайте объект класса *pingreply* – ответ получателя;
- в случае, если статус ответа успешен, выведите на консоль его параметры.

Лабораторная работа №3 РАБОТА С XML

Цель работы: Изучение возможностей расширенного языка разметки XML.
Форма отчета: Демонстрация выполненного задания преподавателю.

Методические указания:

1. Повторить раздел лекционного материала по данной тематике.
2. Создать необходимые файлы-исходники.
3. Продемонстрировать пример работы преподавателю.

Задание 1.

1. Создать XML-документ на основе исходных файлов. Разработать для данного документа DTD-определение и XML-схему.

```
<gradeReport id="120851">
  <date>10-06-2008</date>
  <subject>Computer Science Fundamentals</subject>
  <examiner>prof.Litvinov</examiner>
  <gradeList>
    <gradeRecord id="1">
      <student>Ivanov</student>
      <grade>4</grade>
    </gradeRecord>
    <gradeRecord id="2">
      <student>Petrov</student>
      <grade>3</grade>
    </gradeRecord>
    <gradeRecord id="3">
      <student>Sidorov</student>
      <grade>5</grade>
    </gradeRecord>
  </gradeList>
</gradeReport>
```

2. Проверить на валидность с использованием сервиса <http://www.w3.org/2001/03/webdata/xsv>.

3. Создать JS-скрипт файл загрузки дерева XML-документа и обеспечить навигацию по элементам с использованием отношений между ними. Выполнить изменение произвольного узла.

4. Реализовать XSL-преобразование документа для корректного отображения в браузере. Постройте для него XSL файл, необходимый для XSLT преобразования исходного XML документа в HTML страницу, содержащую заголовок ведомости и таблицу оценок со столбцами: *номер, студент, оценка*.

Лабораторная работа №4 JS

Цель работы: Изучение возможностей навигации по дереву документа, использования регулярных выражений.

Форма отчета: Демонстрация выполненного задания преподавателю.

Методические указания:

1. Повторить раздел лекционного материала по данной тематике.
2. Создать необходимые файлы-исходники.
3. Продемонстрировать пример работы преподавателю.

Задание 1.

Постройте таблицу случайно выбранных цветов. Цвет ячейки таблицы задается с помощью атрибута `bgcolor`. Цвет ячейки описывается в рамках трехкомпонентной модели RGB, например: `<td bgcolor="#c0a145">`. Для генерации каждой компоненты можно использовать генератор случайных чисел с помощью методов объекта `Math` и преобразование в шестнадцатиричный формат:

```
color = Math.round(255.0*Math.random());  
r = color.toString(16).
```

Результирующий цвет образуется путем конкатенации компонентов:

```
color = r + g + b;
```

Задание 2.

Используя технологию DOM обеспечьте вывод разработанного в лабораторной работе (предыдущей) документа в режиме для чтения и в режиме редактирования некоторых полей.

Задание 3.

Постройте регулярное выражение, описывающее шаблон для проверки номера телефона, и внесите все необходимые изменения и дополнения в функцию `CheckData()`.

Лабораторная работа №5 РАЗРАБОТКА ASP.NET ВЕБ-ПРИЛОЖЕНИЙ

Цель работы: Ознакомление с основными этапами разработки веб-приложений на основе ASP.NET в среде Microsoft Visual Studio.NET. Изучение структуры проекта ASP.NET Web Application. Знакомство со средствами автоматизации разработки веб-приложений в ASP.NET в виде серверных элементов управления WebForm. Изучение элементов WebForm, предназначенных для отображения на веб-странице данных, получаемых из источников данных.

Форма отчета: Демонстрация выполненного задания преподавателю.

Методические указания:

1. Повторить раздел лекционного материала по данной тематике.
2. Создать необходимые файлы-исходники.
3. Продемонстрировать пример работы преподавателю.

Задание 1.

Создайте ASP.NET приложение, выводящее список всех *переменных окружения* веб-сервера.

Для этого потребуется вставка следующего кода:

```
<%  
// метод AllKeys формирует список всех ключей массива  
string[] rkeys = Request.ServerVariables.AllKeys;  
  
// создание переменной типа StringBuilder (т.е. изменяемой строки)  
StringBuilder output = new StringBuilder();  
  
// перебор всех элементов массива rkeys и формирование списка значений  
foreach (string rkey in rkeys)  
{  
    output.Append(rkey + "=" + Request.ServerVariables[rkey] + "<br>");  
}  
// формирования ответа сервера  
Response.Write(output);  
%>
```

Задание 2.

Создайте веб-приложение *ASP.NET*, поддерживающее ввод на странице международного телефонного номера пользователем и использующее серверный элемент управления `<asp:RegularExpressionValidator>`.

Задание №3.

Создайте веб-приложение *ASP.NET*, отображающее на веб-странице содержимое таблицы *Exam*, содержащей поля: *идентификатор записи, фамилия студента, название дисциплины, оценка*. При разработке используйте элемент `<asp:GridView>` или `<asp:DataList>`.

Лабораторная работа №6 XSLT-преобразования .NET

Цель работы: Знакомство с методами форматирования и преобразования XML-документов на основе XSLT-преобразований.NET.

Форма отчета: Демонстрация выполненного задания преподавателю.

Методические указания:

1. Повторить раздел лекционного материала по данной тематике.
2. Создать необходимые файлы-исходники.
3. Продемонстрировать пример работы преподавателю.

Задание 1.

Реализуйте и проверьте работоспособность следующего кода на примерах к лабораторной работе.

```
using System;  
using System.IO;  
using System.Xml;  
using System.Xml.XPath;  
using System.Xml.Xsl;
```

```

public class Sample
{
    private const String filename = "";
    private const String stylesheet = "";

    public static void Main()
    {
        XsltTransform xslt = new XsltTransform();
        xslt.Load(stylesheet);
        XPathDocument xpathdocument = new
        XPathDocument(filename);
        XmlTextWriter writer = new XmlTextWriter(Console.Out);
        writer.Formatting=Formatting.Indented;

        xslt.Transform(xpathdocument, null, writer, null);
    }
}

```

Задание 2.

Реализуйте и проверьте работоспособность следующего кода на примерах к лабораторной работе.

```

// Open books.xml as an XPathDocument.
XPathDocument doc = new XPathDocument("books.xml");
// Create a writer for writing the transformed file.
XmlWriter writer = XmlWriter.Create("books.html");
// Create and load the transform with script execution enabled.
XsltCompiledTransform transform = new XsltCompiledTransform();
XsltSettings settings = new XsltSettings();
settings.EnableScript = true;
transform.Load("transform.xsl", settings, null);
// Execute the transformation.
transform.Transform(doc, writer);

```

Задание 3.

Реализуйте и проверьте работоспособность следующего кода на примерах к лабораторной работе.

```

// Create a reader to read books.xml
XmlReader reader = XmlReader.Create("books.xml");
// Create a writer for writing the transformed file.
XmlWriter writer = XmlWriter.Create("books.html");
// Create and load the transform with script execution enabled.
XsltCompiledTransform transform = new XsltCompiledTransform();
XsltSettings settings = new XsltSettings();
settings.EnableScript = true;
transform.Load("transform.xsl", settings, null);
// Execute the transformation.
transform.Transform(reader, writer);

```

Задание 4.

Реализуйте и проверьте работоспособность следующего кода на примерах к лабораторной работе.


```
// Create and load the transform with script execution enabled.
XslCompiledTransform transform = new XslCompiledTransform();
XsltSettings settings = new XsltSettings();
settings.EnableScript = true;
transform.Load("transform.xsl", settings, null);
// Execute the transformation.
transform.Transform("books.xml", "books.html");
```

Задание 5.

1. Преобразуйте программы, созданные в заданиях 1,2,3,4 для работы с командной строки:
 - первый параметр кс – (имя) файл который подлежит преобразованию;
 - второй параметр кс – (имя) файл таблицы преобразования;
 - третий параметр кс – (имя) файла результата.
2. Создайте таблицу преобразования для формирования html документа на основе xml документа из файла purchaseorders.xml.
3. Продемонстрируйте работу приложений для данных файлов.
4. Проверьте правильность результатов преобразования с помощью браузера.

Лабораторная работа №7 РАЗРАБОТКА ВЕБ-СЛУЖБЫ ASP.NET

Цель работы: Изучение создания веб-служб ASP.NET в среде Microsoft Visual Studio.
Форма отчета: Демонстрация выполненного задания преподавателю.

Методические указания:

1. Повторить раздел лекционного материала по данной тематике.
2. Создать необходимые файлы-исходники.
3. Продемонстрировать пример работы преподавателю.

Задание 1.

Создайте ASP.NET веб-службу, поддерживающую три метода, возвращающие соответственно *название* текущего дня *недели*, *номер* текущего дня в *месяце*, *номер* текущего дня в *году*.

Для получения необходимых данных на стороне сервера можно использовать свойства и методы:

```
Label1.Text = DateTime.Now.DayOfWeek.ToString();
Label1.Text = DateTime.Now.Day.ToString();
Label1.Text = DateTime.Now.DayOfYear.ToString();
```

Задание 2.

Создайте ASP.NET прокси-сборку на языке C# для веб-службы, поддерживающей три метода, возвращающие соответственно *название* текущего дня *недели*, *номер* текущего дня в *месяце*, *номер* текущего дня в *году*.

Выбор метода веб-службы должен быть реализован через *форму* с элементом типа *ListBox* или *RadioButton*.

Лабораторная работа №8 WCF WEB

Цель работы: Изучение возможностей программного фреймворка WCF.
Форма отчета: Демонстрация выполненного задания преподавателю.

Методические указания:

1. Повторить раздел лекционного материала по данной тематике.
2. Создать необходимые файлы-исходники.
3. Продемонстрировать пример работы преподавателю.

Задание 1.

1. создайте WEB сервис, реализующий интерфейс простого калькулятора:

```
public interface ICalculator
{
    [OperationContract]
    double Add(double n1, double n2);
    [OperationContract]
    double Subtract(double n1, double n2);
    [OperationContract]
    double Multiply(double n1, double n2);
    [OperationContract]
    double Divide(double n1, double n2);
}
```

2. реализуйте данные методы в службе;
3. создайте клиентское приложение для тестирования работы созданной web службы;
4. создайте web приложение для демонстрации функциональности полученной web службы.

ЛИТЕРАТУРА

1. Конспект лекций;
2. <https://www.intuit.ru/studies/courses/63/63/info>
3. <https://www.intuit.ru/studies/courses/485/341/info>
4. <https://www.intuit.ru/studies/courses/69/69/info>