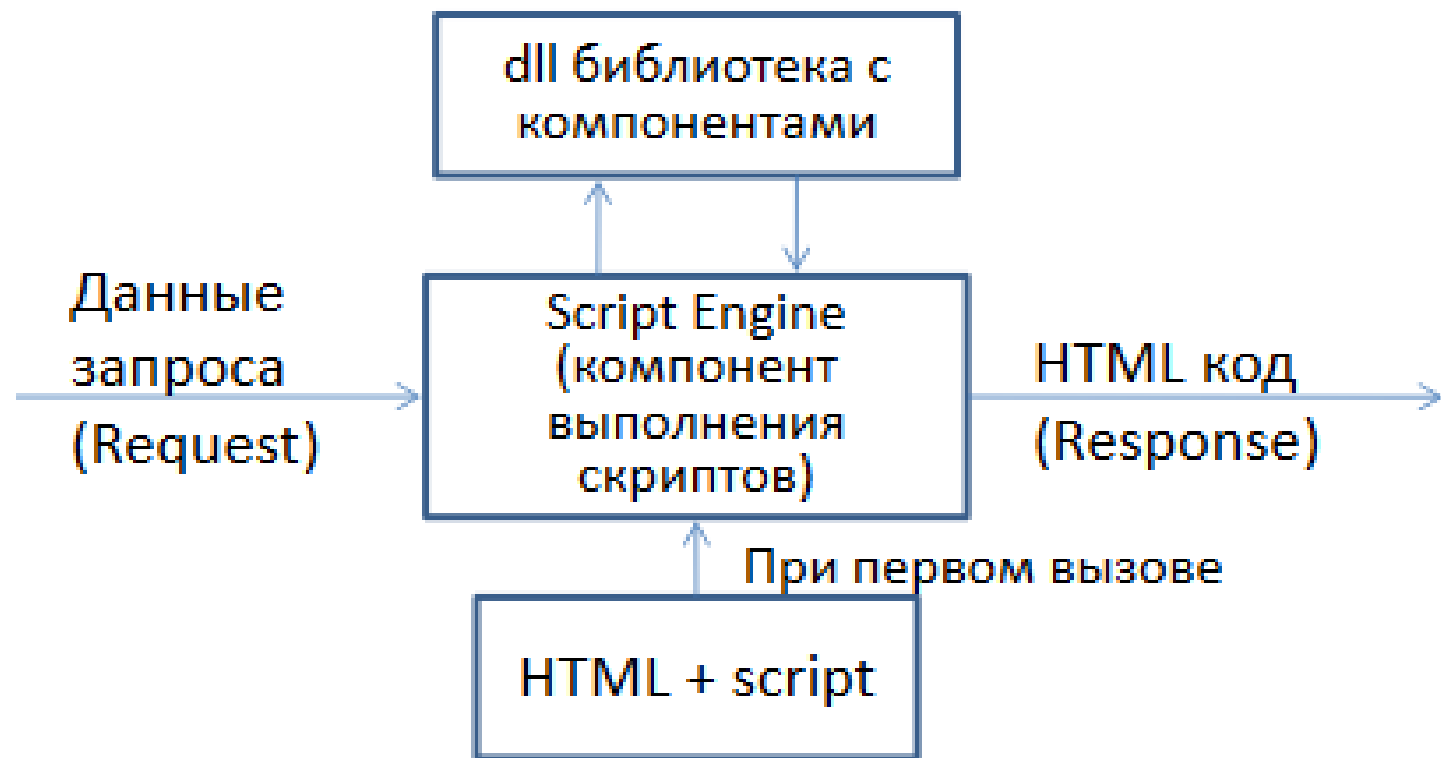


Технология ASP.NET

Приложения на стороне сервера

- ASP.Net
 - Web Forms
 - MVC



Основные идеи ASP.NET

- **Реентерабельные формы** – форма посылает данные на обработку самой себе
- **Сохранение состояния** – между вызовами формы данные (свойства) элементов управления формы сохраняются
- **Объектная модель Web формы** –
 - при обработке на сервере создаются объекты, соответствующие элементам управления
 - на сервер передаются не только данные элементов управления, но и информация о событиях, произошедших на стороне клиента

Типы веб-проектов

- «Web Site» - web-сайты, в состав web-сайта не входят файлы с описанием проекта;
- Проект «ASP.NET Web Application» - Web приложение, в состав которого входят файлы с описанием проекта (*.prj, *.sln и т.п.).

Веб-формы (ASP.NET страницы, веб-страницы)

- Каждое ASP.Net web-приложение включает хотя бы одну web-страницу
- Web-page сохраняется в отдельном файле с расширением *.aspx (это content file или markup file)
- Web page включает:
 - Описание пользовательского интерфейса в виде формы
 - Описание класса производного от класса Page
- Web-page может хранить код обработки сообщений (часть производного класса) в отдельном файле с расширением *.aspx.cs (для языка C#)
- Содержание web-страницы включает:
 - Обычный HTML код
 - Серверные элементы управления

Размещение кода

Описание формы и кода в
одном файле

File.aspx

```
<%@ Page ...>
<%@ Import Namespace="namespace" >
<html>
<script runat="server">
...
</script>
<body>
...
</body>
</html>
```

Описание формы и кода
в разных файлах (code
behind)

File.aspx

```
<% Page CodeFile="xxx.cs" ...>
<html>
<body>
...
<form runat="server">
...
</form>
</body>
</html>
```

File.aspx.cs

```
using System;
using System.Web;
...

public partial class _Default :
    System.Web.UI.Page {
    protected void Page_Load
        (object s, EventArgs e) {
        ...
    }
}
```

Страница Веб-приложения

default.aspx.cs

default.aspx



```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="default.aspx.cs" Inherits="ClientApp._default" %>
```

```
<!DOCTYPE html>
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head runat="server">
```

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
```

```
<title></title>
```

```
</head>
```

```
<body>
```

```
<form id="form1" runat="server">
```

```
<div>
```

```
<asp:Button ID="Button1" runat="server" Text="Button" OnClick="Button1_Click" />
```

```
</div>
```

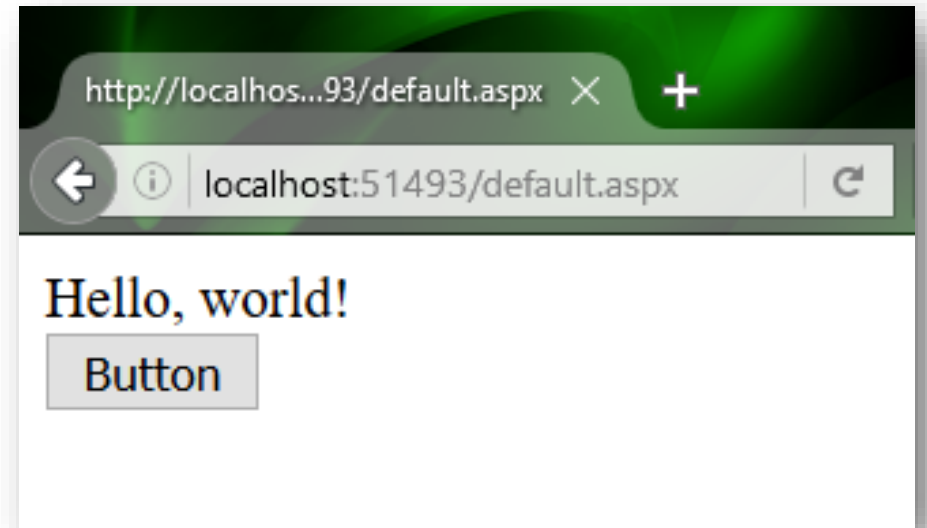
```
</form>
```

```
</body>
```

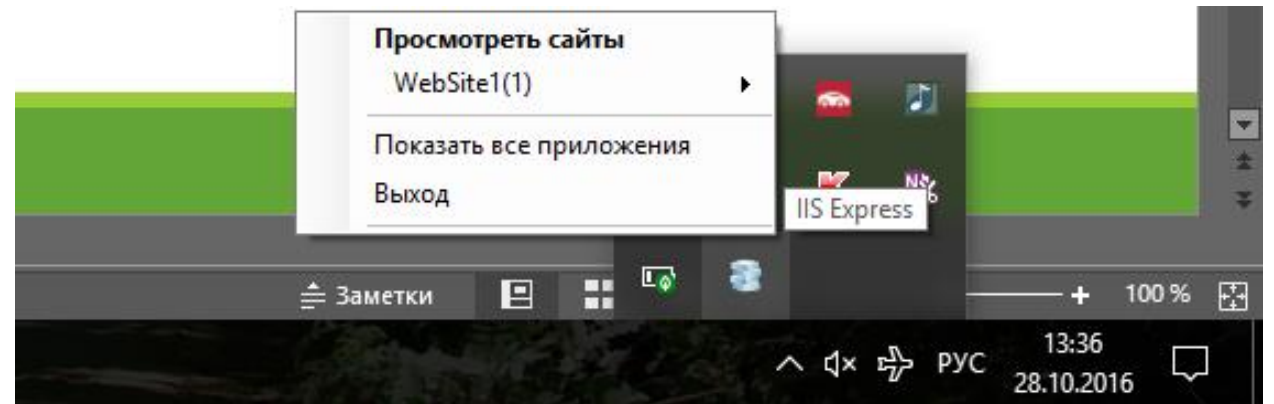
```
</html>
```

Программный код примера

```
using System;
namespace ClientApp
{
    public partial class _default : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
        }
        protected void Button1_Click(object sender, EventArgs e)
        {
            Response.Write("Hello, world!");
        }
    }
}
```



- Для отладки веб-приложений в Visual Studio используется IIS Express, который запускается при запуске приложения на выполнение.



Отладочный веб-сервер

Обработка первого запроса



и
последующих...



Проверка данных

- ASP.NET включает элементы управления, выполняющие проверку введенных данных (валидаторы), которые могут автоматически проверять данные введенные пользователем в других ЭУ и показывать ошибки.
- Специальный метод `IsValid` валидаторов вызываются до обработки любых других событий.
- Валидаторы выполняются не в ответ на какие-либо события. Вместо этого достаточно вызвать метод `Page.IsValid` в любом обработчике события.
- Далее будут подробно рассматриваться ЭУ для выполнения проверки (Validation Controls).

Элемент управления	Назначение
CompareValidator	Проверяет значение в элементе ввода на равенство заданному значению другого элемента ввода или фиксированной константе.
CustomValidator	Позволяет строить пользовательскую функцию проверки достоверности, которая проверяет заданный элемент управления.
RangeValidator	Определяет, находится ли данное значение в пределах предварительно заданного диапазона.
RegularExpressionValidator	Проверяет значение в ассоциированном элементе ввода на соответствие шаблону регулярного выражения.
RequiredFieldValidator	Проверяет заданный элемент ввода на наличие значения (т.е. что он не пуст).
ValidationSummary	Отображает итог по всем ошибкам проверки достоверности страницы в формате простого списка, маркированного списка или одного абзаца. Ошибки могут отображаться встроенным способом и/или во всплывающем окне сообщения.

Элементы управления проверки

- используются для генерации кода JavaScript клиентской стороны (возможно и серверной тоже) в целях проверки достоверности формы

Основные свойства проверочных ЭУ

Член	Назначение
ControlToValidate	Получает или устанавливает элемент управления, подлежащий проверке достоверности.
Display	Получает или устанавливает поведение сообщений об ошибках в элементе управления проверкой достоверности.
EnableClientScript	Получает или устанавливает значение, указывающее, включена ли проверка достоверности клиентской стороны.
ErrorMessage	Получает или устанавливает текст для сообщения об ошибке.
ForeColor	Получает или устанавливает цвет сообщения, отображаемого при отказе проверки достоверности.

Пример использования

Панель элементов

Стандартный

Данные

Проверка

Указатель

CompareValidator

CustomValidator

RangeValidator

RegularExpressionValidat...

RequiredFieldValidator

ValidationSummary

WebTest

Default.aspx.cs

Default.aspx

div

Сообщение об ошибке 1.

Сообщение об ошибке 2.

Фамилия

**

Имя

*

E-Mail

**

Сохранить

[Label1]

Пример использования

```
protected void Button1_Click(object sender, EventArgs e)
{
    if (Page.IsValid)
        Label1.Text = txtFirstName.Text + ":"+txtLastName.Text + ":"+txtEMail.Text;
}

protected void CustomValidator1_ServerValidate(object source, ServerValidateEventArgs args)
{
    args.IsValid = txtFirstName.Text != "Lennon";
}
```


Класс Page

- Web страница также реализуется в виде объекта специального класса Page, из пространства имен System.Web.UI.
- Все классы code-behind явно наследуются от базового класса System.Web.UI.Page. Это означает, что каждая создаваемая web form имеет большое количество функциональности.
- Наследование от класса Page предоставляет следующие очень полезные свойства :
 - Session (ссылка на объект класса Session)
 - Application (ссылка на объект класса Application)
 - Cache (ссылка на объект класса Cache)
 - Request
 - Response
 - Server
 - User
 - Trace
 - Метод FindControl()
 - свойство IsPostBack
 - Controls

Доступные технологии оформления страниц

Шаблоны (MasterPage)

- Позволяют задать шаблон разметки для группы страниц

Темы (Theme)

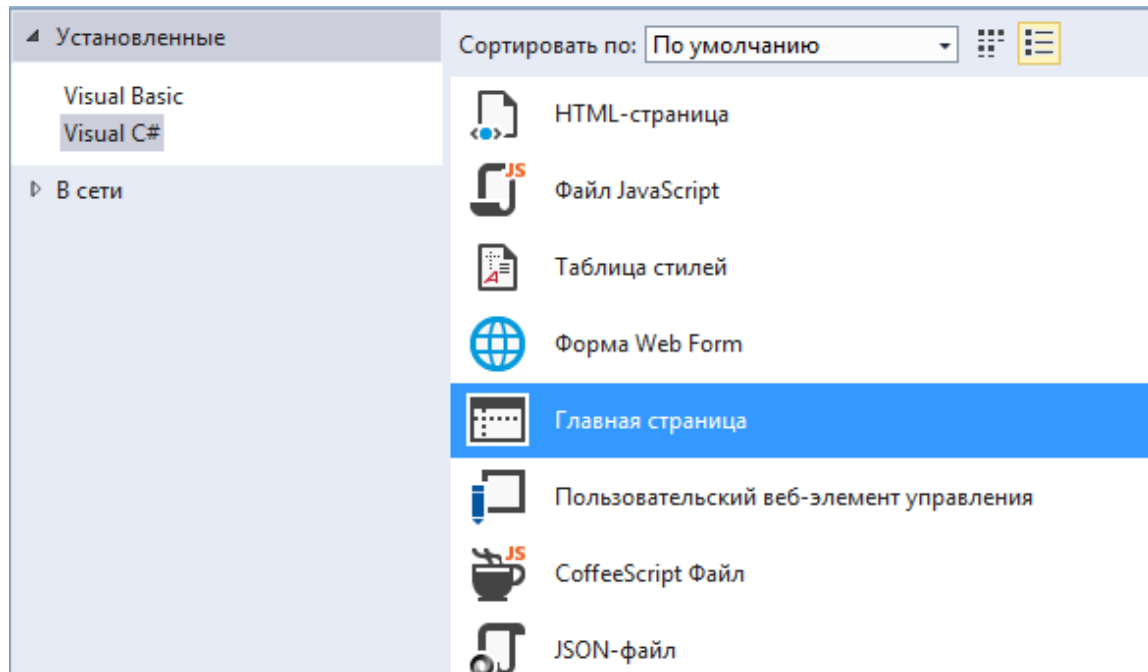
- Определяют набор файлов стилей и изображений

Скины (Skin)

- Задают набор общих визуальных свойств для элементов управления

Использование шаблонов

- Типы страниц:
 - мастер страницы (master pages) – шаблон страницы
 - содержит фиксированные элементы, одинаковые для всех страниц – например, верхний и нижний колонтитулы, панель навигации, панель меню и т.д.
 - может содержать элементы, допустимые для обычной страницы, а также программный код.
 - страницы содержимого (content pages) – каждая страница содержимого ссылается на одну мастер страницу, от которой получает элементы и их расположение.



```
MainPage.master  X
<%@ Master Language="C#" AutoEventWireup="true" CodeFile="MainPage.master.cs" Inherits="MainPage" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
<title></title>
<asp:ContentPlaceHolder id="head" runat="server">
</asp:ContentPlaceHolder>
</head>
<body>
<form id="form1" runat="server">
<div>
<asp:ContentPlaceHolder id="ContentPlaceHolder1" runat="server">

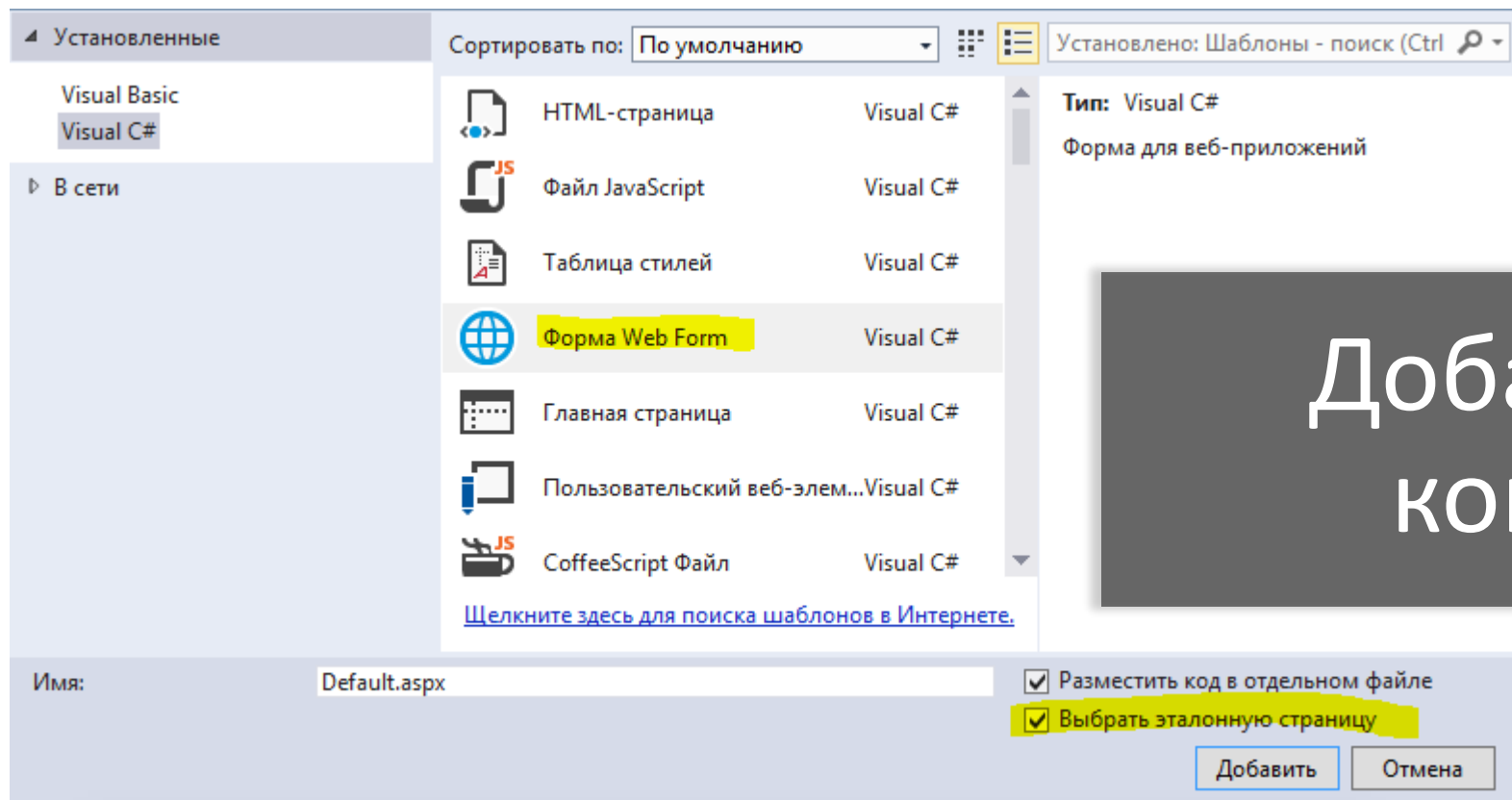
</asp:ContentPlaceHolder>
</div>
</form>
</body>
</html>
```

Веб-страницы начинаются с директивы *Page*, а *мастер страница* – с директивы *Master*

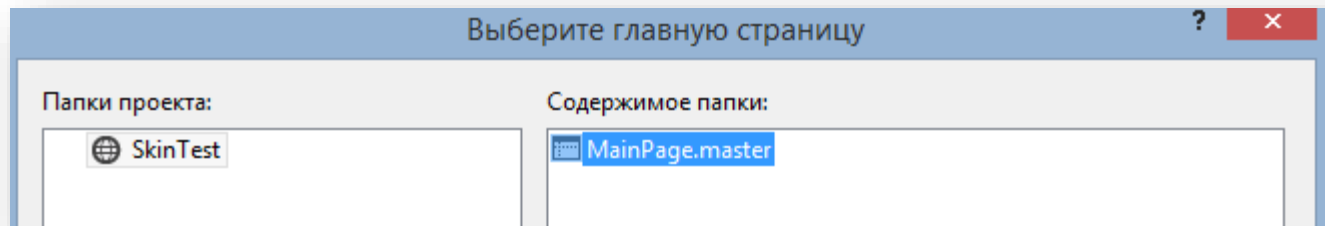
Создание шаблона

```
<form id="form1" runat="server">
<div>
    <h1>Заголовок шаблона</h1>
    <asp:ContentPlaceHolder id="ContentPlaceHolder1" runat="server">
    </asp:ContentPlaceHolder>
    <h1>Окончание шаблона</h1>
</div>
</form>
.
```

Редактирование шаблона



Добавление контента



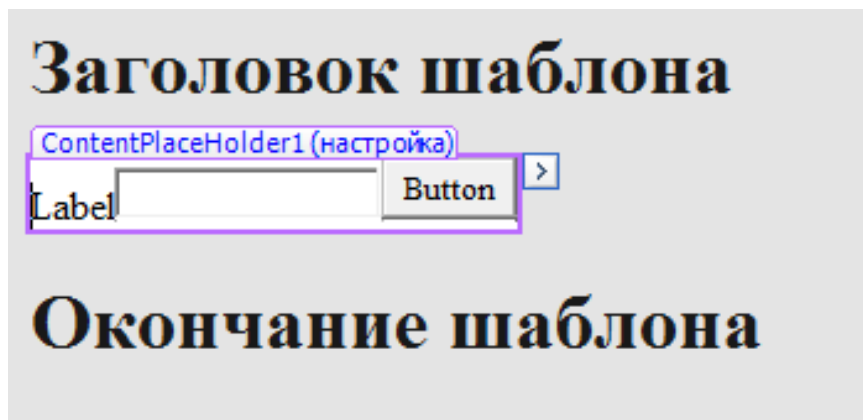
```
<%@ Page Title="" Language="C#" MasterPageFile="~/MainPage.master" AutoEventWireup="true" CodeFile="Default.aspx.cs"
    <asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
    </asp:Content>
    <asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
    </asp:Content>
```

Применение шаблонов

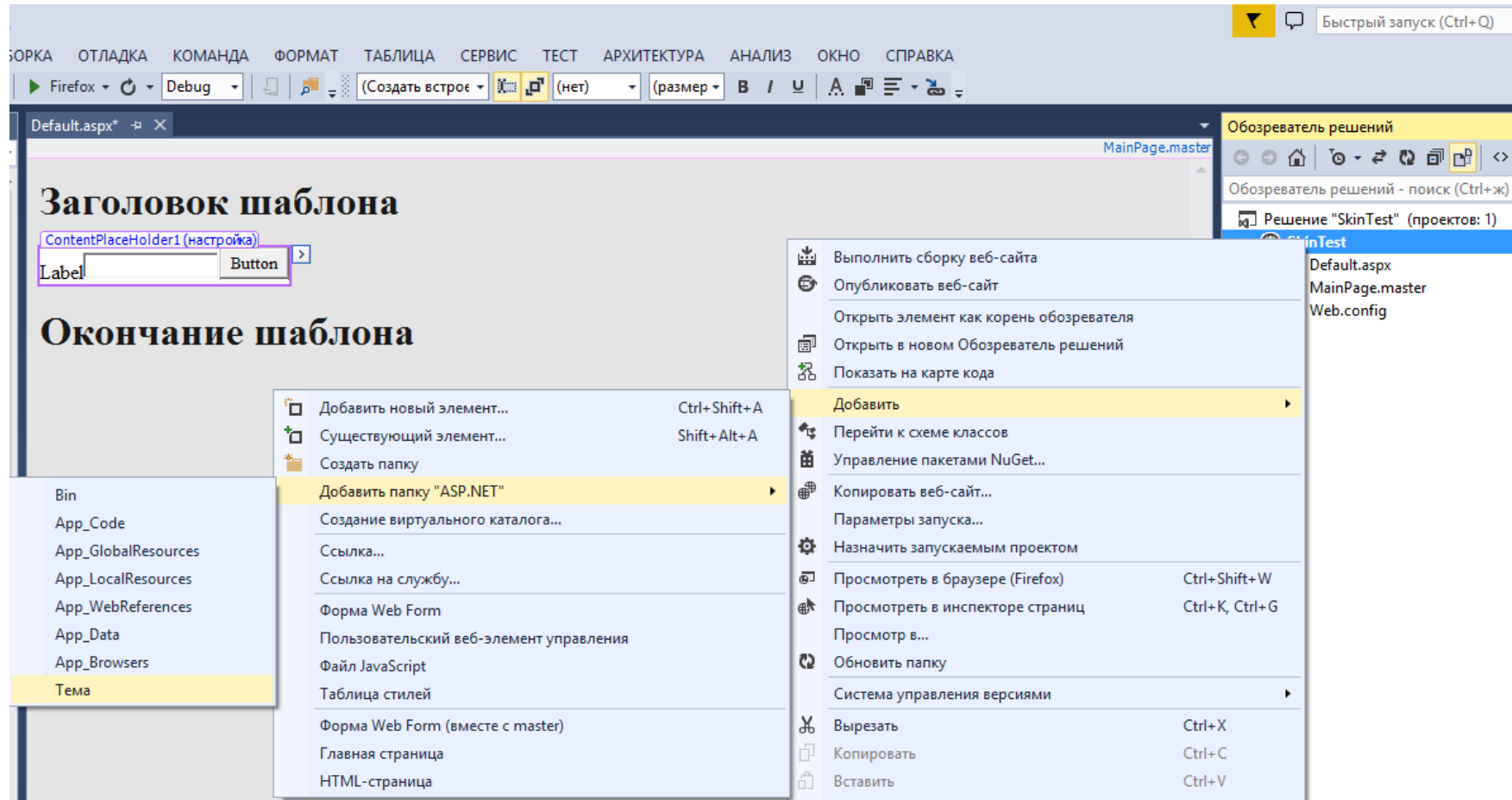
КОД СТРАНИЦЫ

```
<%@ Page Title="" Language="C#" MasterPageFile="~/MainPage.master" AutoEventWireup="true" (
    <asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
    </asp:Content>
    <asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
        <asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>
        <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
        <asp:Button ID="Button1" runat="server" Text="Button" />
    </asp:Content>
```

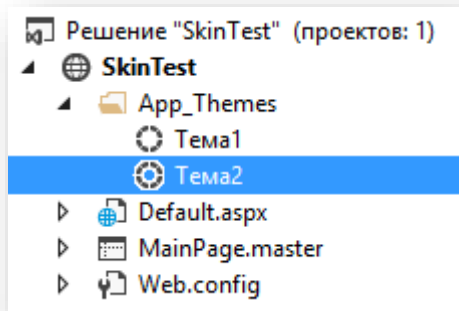
ДИЗАЙН СТРАНИЦЫ



Создание папки тем

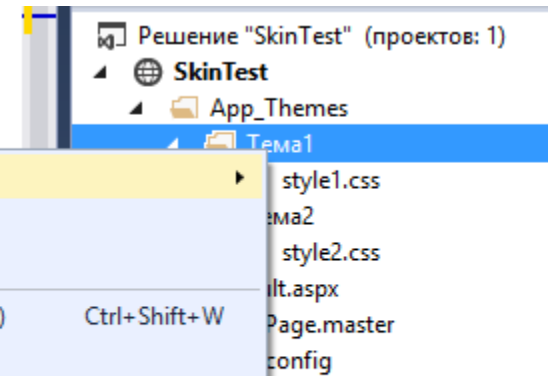
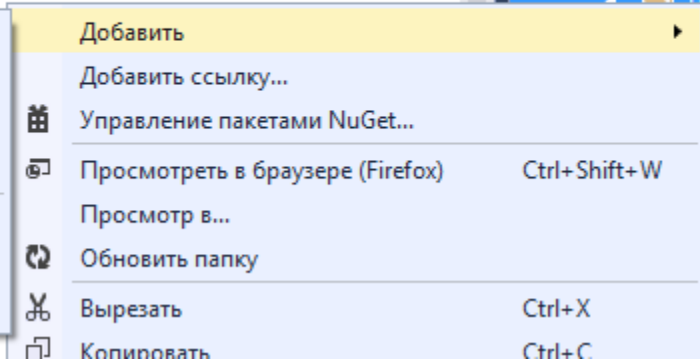
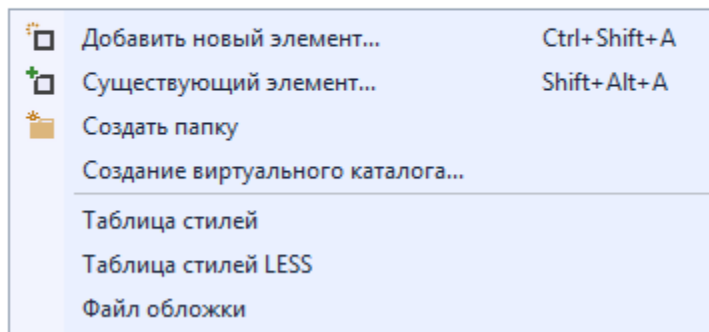


Добавление тем

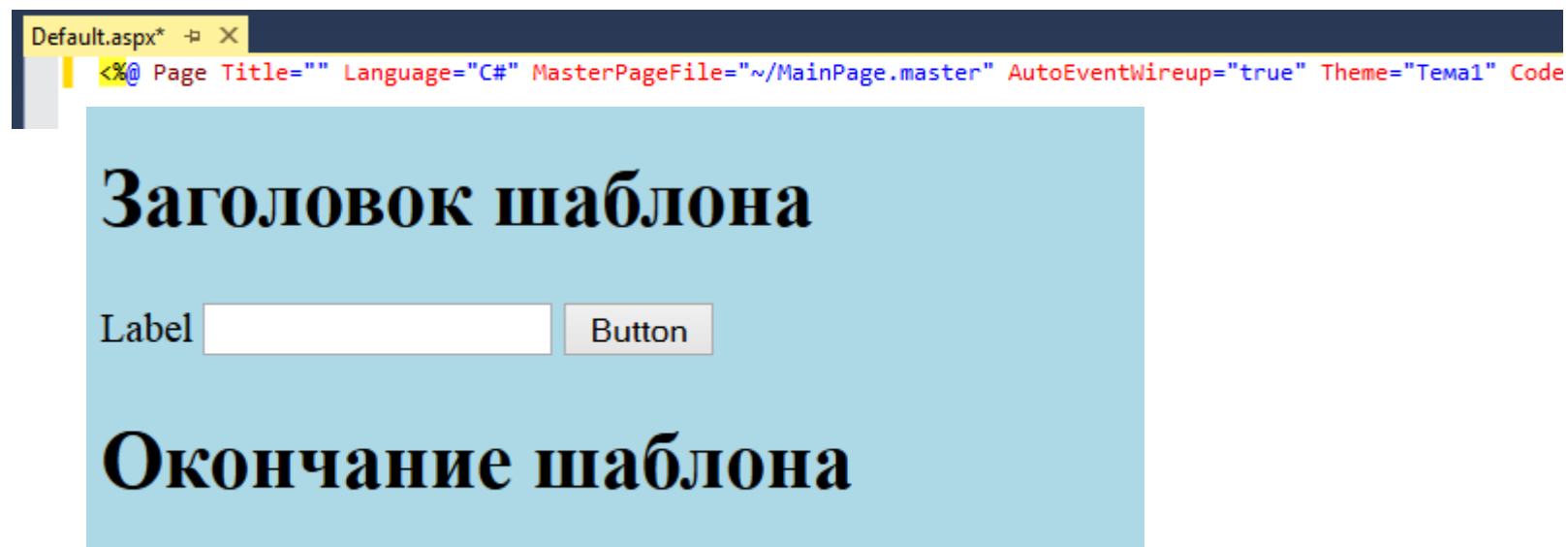


Контент для папки ТЕМА:

- CSS файлы
- Изображения
- Файлы обложки («скины»)



Применение темы



```
<!DOCTYPE html>

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

</title>
<link href="App Themes/%D0%A2%D0%B5%D0%BC%D0%B01/style1.css" type="text/css" rel="stylesheet" /></head>
<body>
    <form method="post" action="." id="form1">
    <div class="aspNetHidden">
    <input type="hidden" name="__VIEWSTATE" id="__VIEWSTATE" value="DbRjwZLZwo7BA1tFb9m4SHSvNSze1Odu7pm9JeZH"
    </div>
```

Файл обложки

```
Skin1.skin X Default.aspx
<asp:Button runat="server"
  BackColor="#9933FF"
  BorderColor="#003366"
  ForeColor="White" />
```

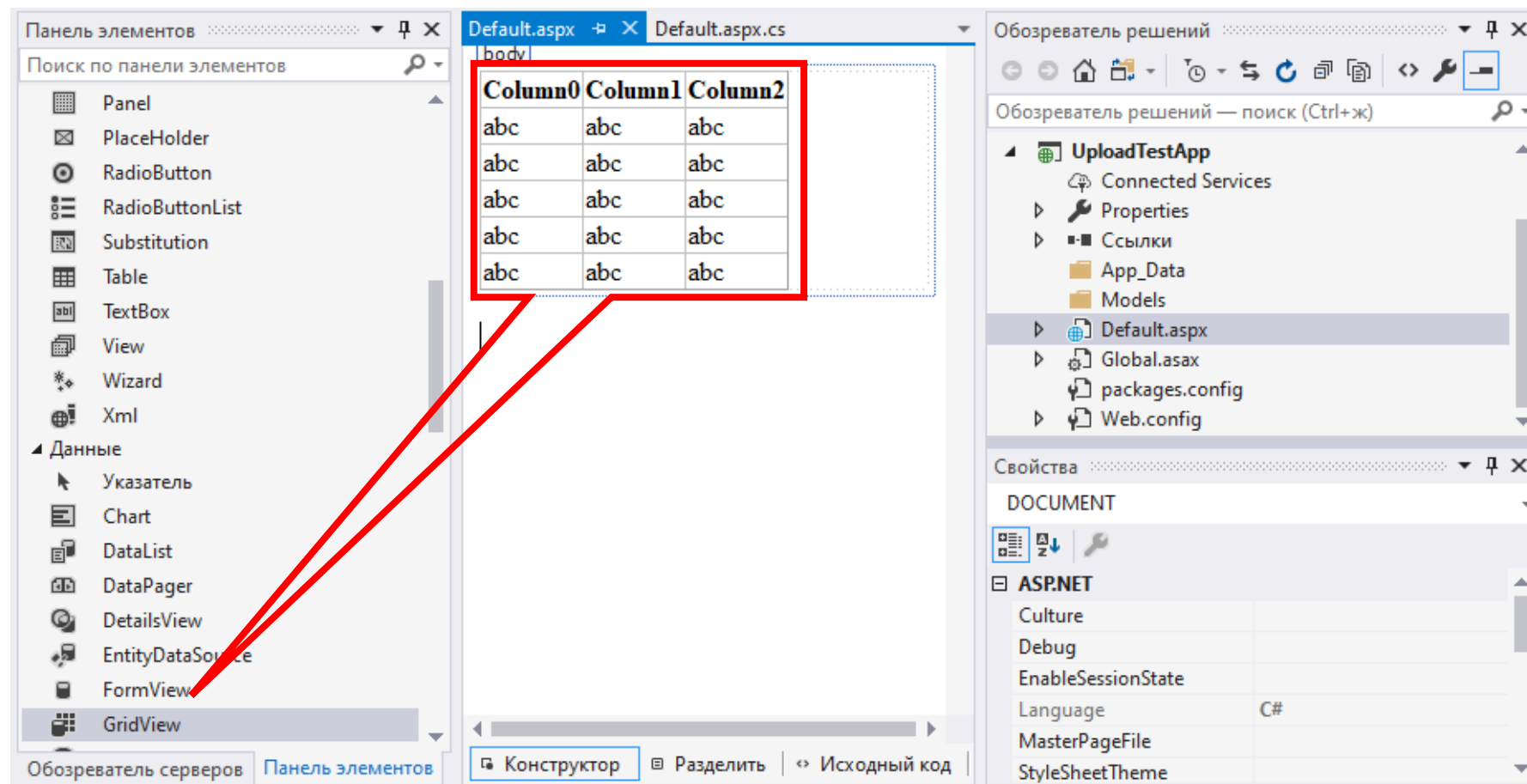
```
<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
  <asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>
  <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
  <asp:Button ID="Button1" runat="server" Text="Button" Skinid="id1"/>
</asp:Content>
```

Заголовок шаблона

Label Button

Окончание шаблона

Доступ к данным



Пример кода

```
protected void Page_Load(object sender, EventArgs e)
{
    string path = @"Data Source=DESKTOP-CGL72NT;Initial Catalog=Sal
    string query = "SELECT * FROM Product";
    DataSet ds = new DataSet();
    using (SqlConnection con = new SqlConnection(path))
    {
        SqlDataAdapter da = new SqlDataAdapter(query, con);
        da.Fill(ds, "Product");
        GridView1.DataSource = ds.Tables["Product"];
        GridView1.DataBind();
    }
}
```

Column0	Column1	Column2
abc	abc	abc
abc	abc	abc
abc	abc	abc
abc	abc	abc
abc	abc	abc

Задачи GridView

Автоформат...

Выборить источник данных: (Нет)

Автоформат

Выберите схему:

- Снять форматирование
- Цветной
- Классический
- Простой
- Профессиональный
- Осень
- Океаника
- Бастр
- Аспидный сланец
- Песок и небо
- Дождливый день
- Заснеженная сосна
- Сирень в тумане
- Черный и синий 1
- Черный и синий 2
- Поле клевера
- Яблоневый сад
- Мокко

Образец:

Column0	Column1	Column2
abc	abc	abc
abc	abc	abc
abc	abc	abc
abc	abc	abc
abc	abc	abc

OK Отмена Применить

localhost:53384/Default.aspx

ProductId	ProductName	Price
1	Валенки	2500,0000
2	Галоши	850,0000
4	Сандалии	3500,0000

Форматирование без CSS



localhost:53384/Default.aspx



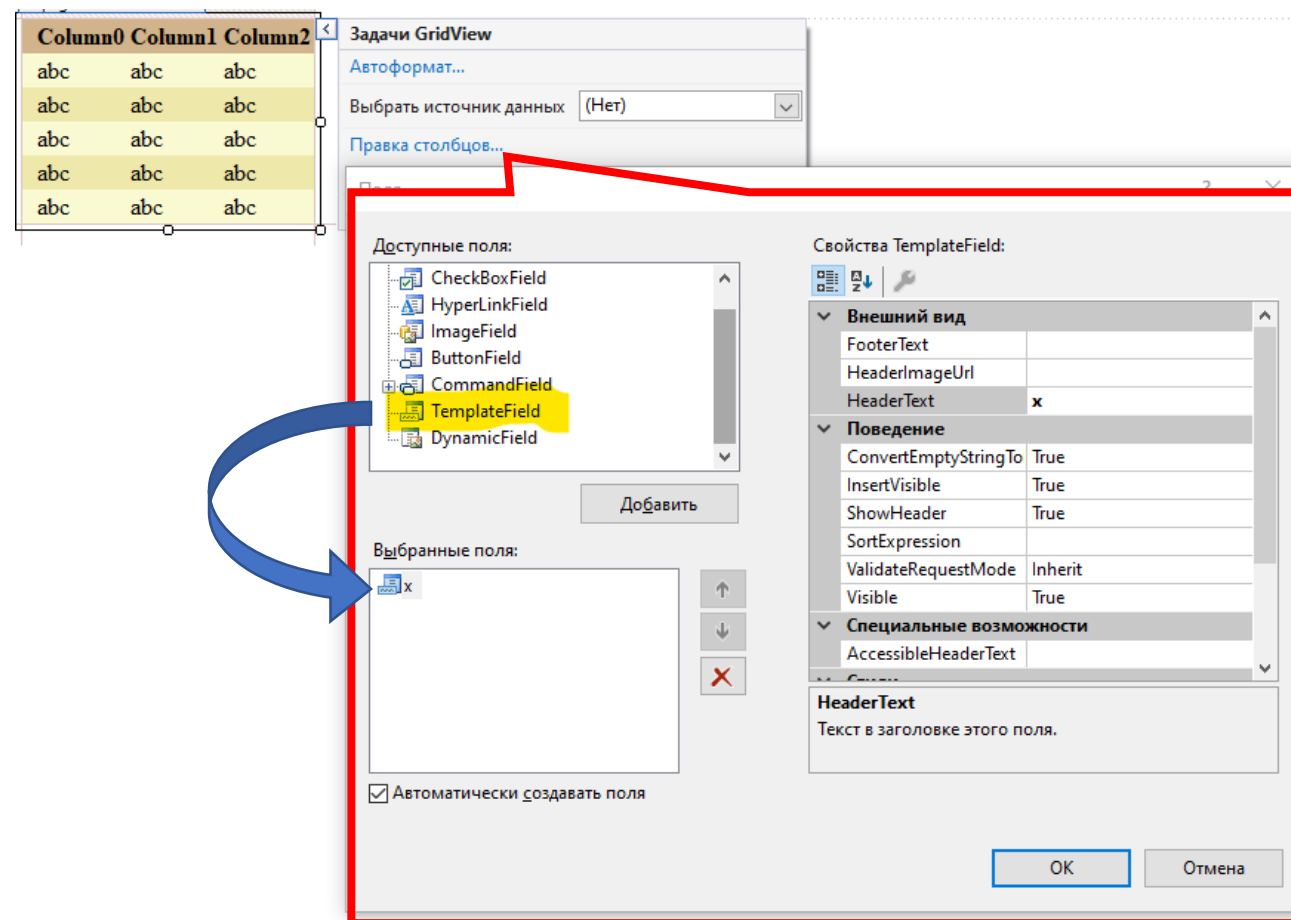
localhost:53384...



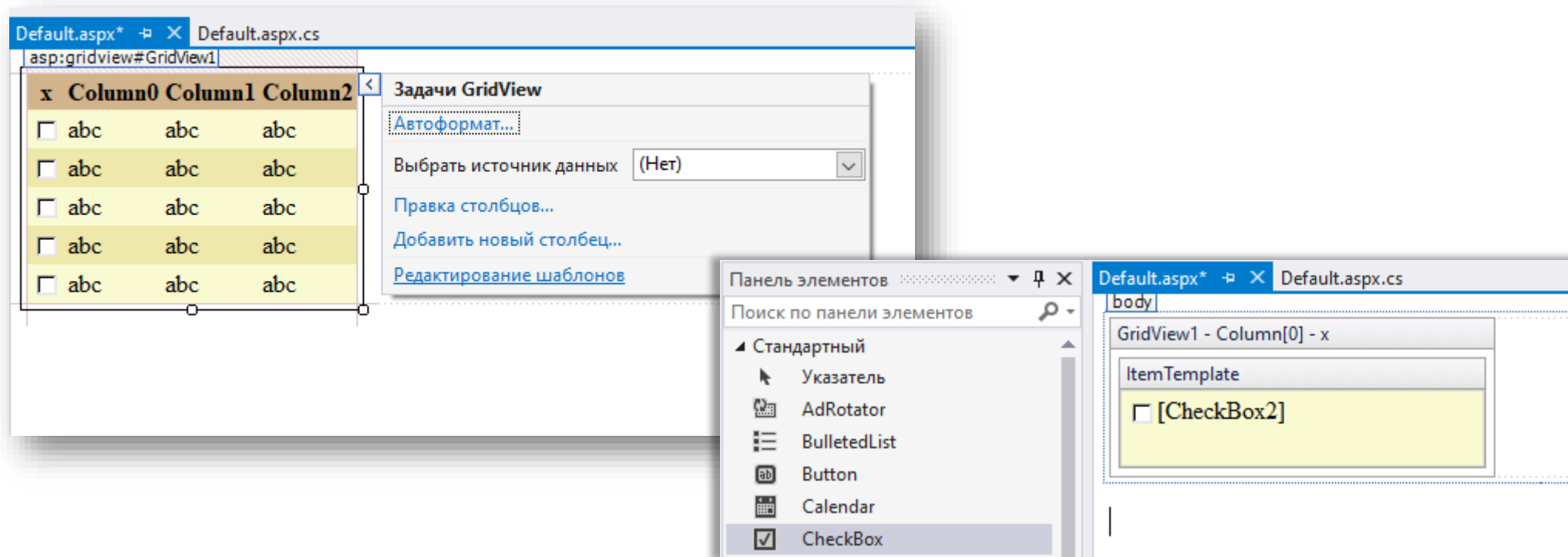
ProductId	ProductName	Price
1	Валенки	2500,0000
2	Галоши	850,0000
4	Сандалии	3500,0000

ИТОГ:

Новый столбец...



Редактирование шаблона





localhost:53384/Default.aspx



localhost:53384...



x	ProductId	ProductName	Price
<input checked="" type="checkbox"/>	1	Валенки	2500,0000
<input type="checkbox"/>	2	Галоши	850,0000
<input checked="" type="checkbox"/>	4	Сандалии	3500,0000

ИТОГ

SOMEE.COM

➔ Free .Net Hosting

- Free ASP.Net web hosting
- 150MB storage, 5GB transfer
- ASP.Net 4.6/4.5/4.0/3.5/2.0
- 15MB MSSQL 2012/2014/2016
- Free third level domain
- FTP access

[Learn More](#)

\$0.00

➔ MS SQL Hosting

- MS SQL 2016/2014/2012/2008R2
- 500MB data, 750MB log size
- 1500MB backup storage
- Auto backup every 12 hours
- Manual backup/restore/attach
- Remote database access

[Learn More](#)

\$7.85

➔ Windows hosting

- 1000MB premium storage
- 50GB premium transfer
- ASP / ASP.Net 2.0-4.6 / PHP
- Dedicated application pool
- 100MB MSSQL 2016/14/12/08
- 99.9% Uptime guarantee

[Learn More](#)

\$7.95

➔ Virtual Server

- Hosted on VMWare ESXi
- 2 CPU at 2000MHz
- 2048MB RAM 60GB SSD
- KVM full access
- Windows license fee included
- 24/7/365 Support

[Learn More](#)

\$17.49

Создание сайта

User menu

User (assist)

Create support ticket

Support tickets

Checkout

Account

Password change

Addresses

Payment methods

Invoices

Billing options

Active packages

Managed products

Virtual servers

Websites

MS SQL

Databases

Logins

Email

Add new mail domain

Create website

Hosting plan: Hosting plan "Freebie"

⚠

Although your hosting plan supports global domains you still need to provide initial default domain name which is hosted within our zone. You will be able to add additional domains later in control panel.

Site name (Subdomain):

Zone name:

Operating system:

ASP.Net version:

Site title:

Site description:

i

Your site will have default domain names as 'Sub domain'. 'Zone name' and www.'Sub domain'. 'Zone name'. Also, if supported by hosting plan, you can use your own domain names (registred with domain name registrar). DNS record will be created instantly, but because of DNS replication delay it may take up to 24 hours for your site to be visible to all users on the Internet. You can try to access your site right after it's created. If it's not working retry it every 30 minutes.

The process may take up to several minutes. Please be patient.

Завершение создания сайта

Create website

You website was created successfully!



You can upload content of your web site with either FTP or built in file manager.

If you'll use FTP – DO NOT upload files into a root directory. Upload your web content into subdirectory with a name of "www." + SiteName. For your convinience you can also use protected directory (named: "protected." + SiteName) to store files that you don't want to be directly accessable to users (ex: Database, configuration files..)

Depending on a location of your web site you may or may not see sub directories of your other web sites in the root directory - this is normal.



Because of DNS replication delay it may take up to 24 hours for your site to be visible to all users on the Internet. You can try to access your site right after it's created. If it's not working retry it every 30 minutes.

[Manage website](#)

Управление сайтом

User menu

User (assist)

Create support ticket

Support tickets

Checkout

Account

Password change

Addresses

Payment methods

Invoices

Billing options

Active packages

Managed products

Virtual servers

Websites

assist.somee.com

Applications

Website properties

Site name(s): **assist.somee.com**
(<http://assist.somee.com>; <http://www.assist.somee.com>;)

Supplied package ID: SPID788894
Managed product ID: MPID2890878

FTP addresses: (<ftp://assist.somee.com>; <ftp://www.assist.somee.com>;)
FTP username: **assist**
FTP credentials (Username and password) are the same as your Username and password.

IP address: **66.85.73.147 (Shared)**

Hosting plan: **Hosting plan "Freebie"**
Registration date: **10/24/2017 8:17:06 AM**

Создание БД

SOME.com User home Store System help Ask a question Search help Logo

User menu

- Create support ticket
- Support tickets
- Checkout
- Account
 - Password change
 - Addresses
 - Payment methods
 - Invoices
 - Billing options
- Active packages
- Managed products
 - Virtual servers
 - Websites
 - MS SQL
 - Databases
 - Logins

MS SQL Databases

Product ID	Package	MS SQL plan	Database name
MPID2890880	Free hosting package (Supplied package ID: SPID788894)	MS SQL Plan "Novice"	Database is not created Create

Create empty MS SQL database

MS SQL plan: MS SQL Plan "Novice"

Database name (7-30 symbols):

Zone name:

MS SQL login:

MS SQL Server version:



Warning! Remember when choosing MS SQL Server version that they are not backward compatible! So, if, at your development environment you have server with lower MS SQL version than you'll choose here, you won't be able to move the database from here back to your development environment without upgrading your MS SQL server!

The process may take up to several minutes. Please be patient.









Управление БД

MS SQL Database properties

FTP credentials (Username and password) are the same as your Username and password.

Data file size, Max/Used
/Usage: **15MB / 2.19MB / 14.58%**

Log file size, Max/Used
/Usage: **20MB / 1.00MB / 5.00%**

- Database related links:
-  [New SQL query](#)
 -  [New SQL query from file](#)
 -  [Backup file manager](#)
 -  [Restore database](#)
 -  [Backup database](#)
 -  [Attach database](#)
 -  [Get data and log files](#)
 -  [Delete database](#)

Upload and attach

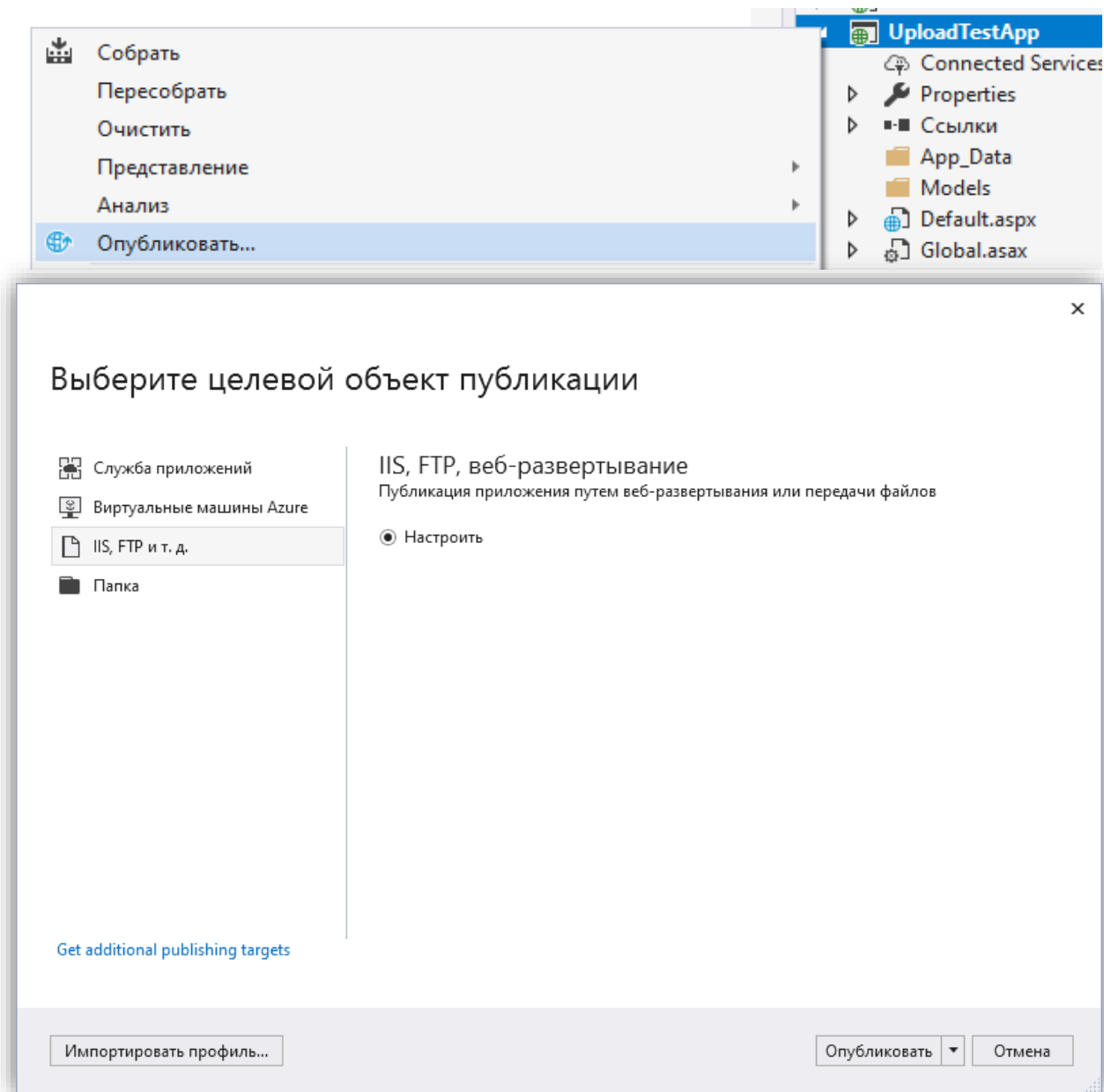
MS SQL data file (*.mdf): Файл не выбран.

MS SQL log file (if exists): Файл не выбран.

Total size of both files cannot exceed 50MB! If it's more than use a 'Backup file manager'.

The process may take up to several minutes. Please be patient.

Публикация проекта



Настройка подключения

Опубликовать

Опубликовать

Подключение

Параметры

CustomProfile *

Способ публикации: FTP

Сервер: ftp://assist.somee.com

Путь к сайту: www.assist.somee.com


☒ Пассивный режим

Имя пользователя: assist

Пароль:

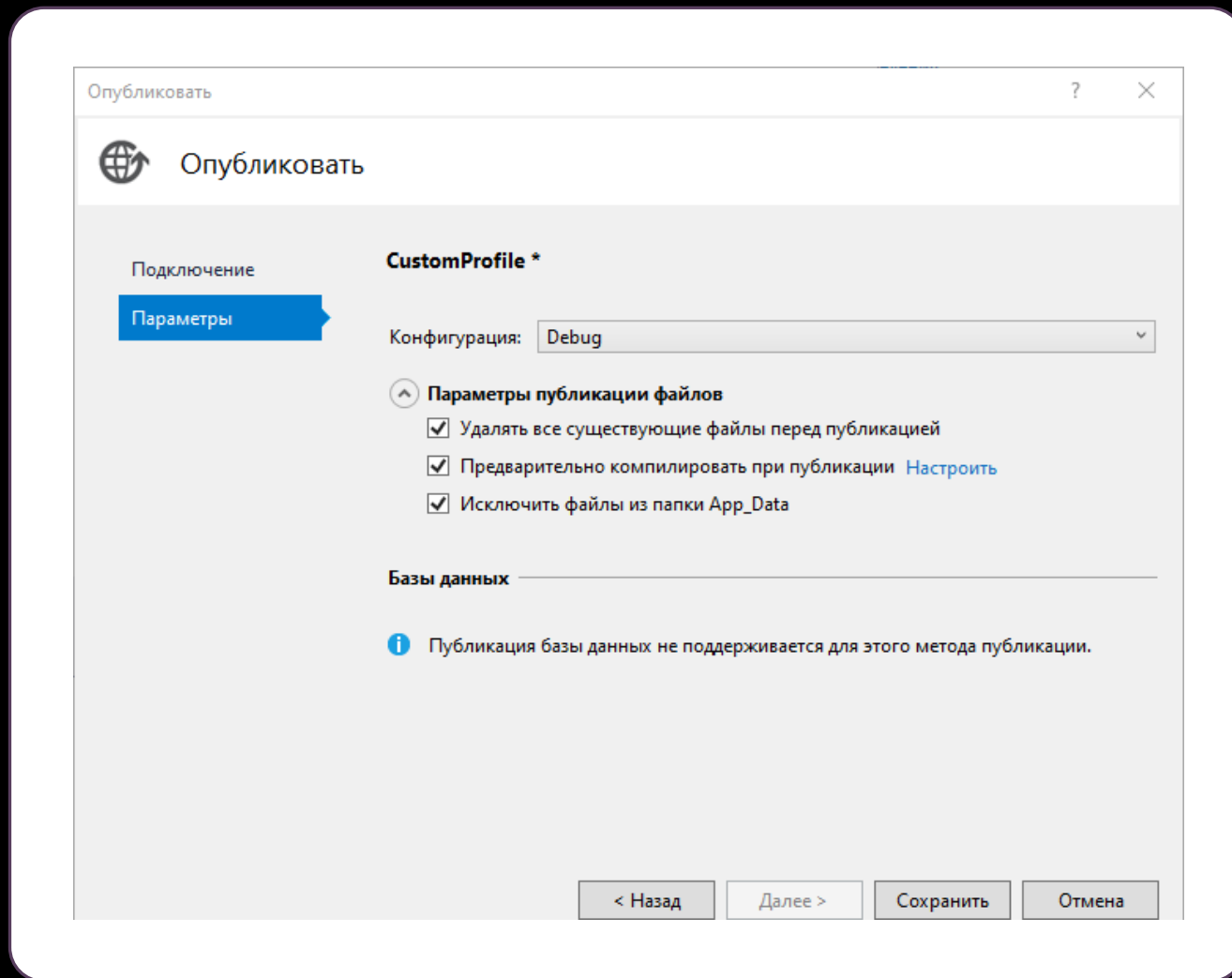
☒ Сохранить пароль

Целевой URL-адрес: ftp://assist.somee.com

Проверить подключение 

< Назад Далее > Сохранить Отмена

Параметры публикации...



Публикация...

