

Упражнение 7

Създаване на уеб базирано приложение “Виртуална библиотека” за демонстрация на връзка с други приложения, чрез използване на комуникационните технологии NET Remoting и Web Services

ФАЗА I.

Създаване на сайт “Виртуална библиотека”

1. Създайте нова папка **Client_Server_2**. В нея копирайте крайния проект **eLibrary_v3** на приложението от първата част на семестъра и базата данни **biblio.mdf**. От този проект ще се използва само сървърната част и по-точно само класа за отдалечен достъп до данните **DAL**. В този клас трябва да се направят някои промени с оглед на това, че новото приложение ще бъде уеб-базирано и е необходимо да се извличат само част от данните, а не целите таблици с 9000 записа, както беше при настолното приложение.

2. Отворете **eLibraryClient.sln** на проекта eLibraryClient и от сървърната част на приложението, отворете класа DAL.

Добавете:

- функцията **getBooksByCategory**, която е аналогична на съществуващата **getBookByISBN** (може да се копира от нея), но селектира само част от полетата (липсва Description)

```
public DataTable getBooksByCategory(String pCategory)
{
    DataTable dt = new DataTable();
    SqlConnection connection = new SqlConnection();
    SqlCommand cmd = new SqlCommand();

    connection.ConnectionString =
eLibraryServer.Properties.Settings.Default.biblioConnectionString;
    try
    {
        using (connection)
        {
            connection.Open();
            cmd.Connection = connection;
            cmd.CommandText = "Select Authors, Title, ISBN from Titles WHERE BookTypeID='" + pCategory + "'";
            dt.Load(cmd.ExecuteReader());
        }
    }
    catch (SqlException e) { connection.Close(); }
    return dt;
}
```

- функцията **getBookTypes**, която връща само типовете на книгите, необходими ни за зареждане на менюто.

```
public DataTable getBookTypes()
{
    DataTable dt = new DataTable();
    SqlConnection connection = new SqlConnection();
    SqlCommand cmd = new SqlCommand();
```

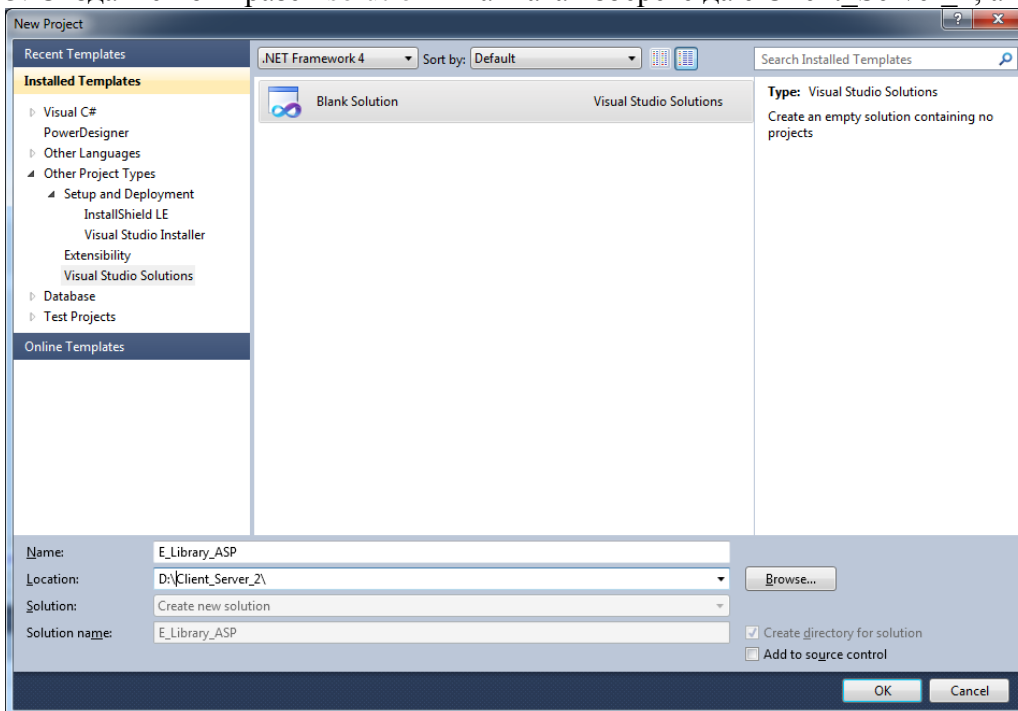
```

connection.ConnectionString = eLibraryServer.Properties.Settings.Default.biblioConnectionString;
try
{
    using (connection)
    {
        connection.Open();
        cmd.Connection = connection;
        cmd.CommandText = "Select * from BookTypes ";
        dt.Load(cmd.ExecuteReader());
    }
}
catch (SqlException e) { connection.Close(); }
return dt;
}

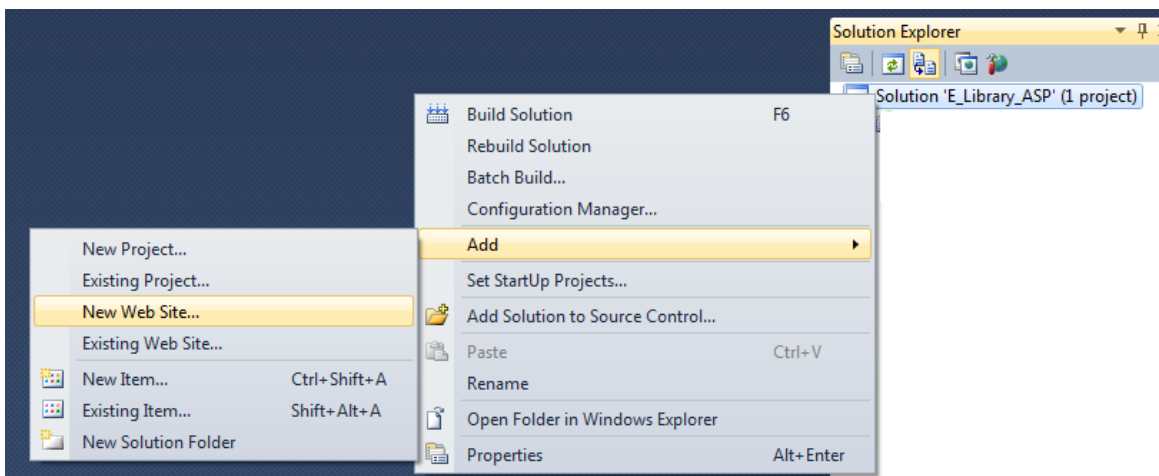
```

Затворете приложението.

3. Създайте нов празен solution – папката изберете да е Client_Server_2, а името E_Library_ASP.



4. Добавете нов празен сайт от типа ASP.Net Empty Web Site. Името му е E_Library_ASP

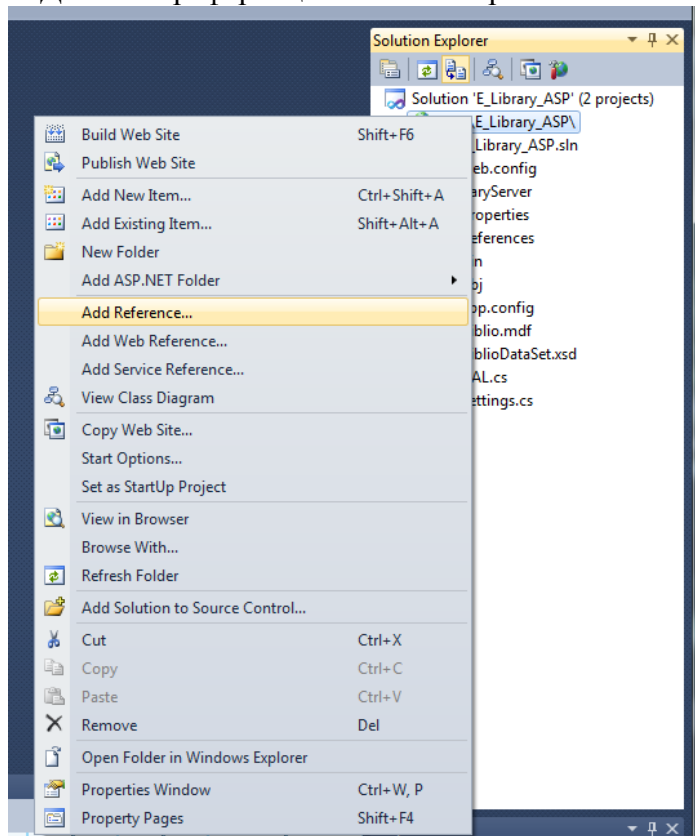


ДОБАВЯНЕ НА eLibraryServer

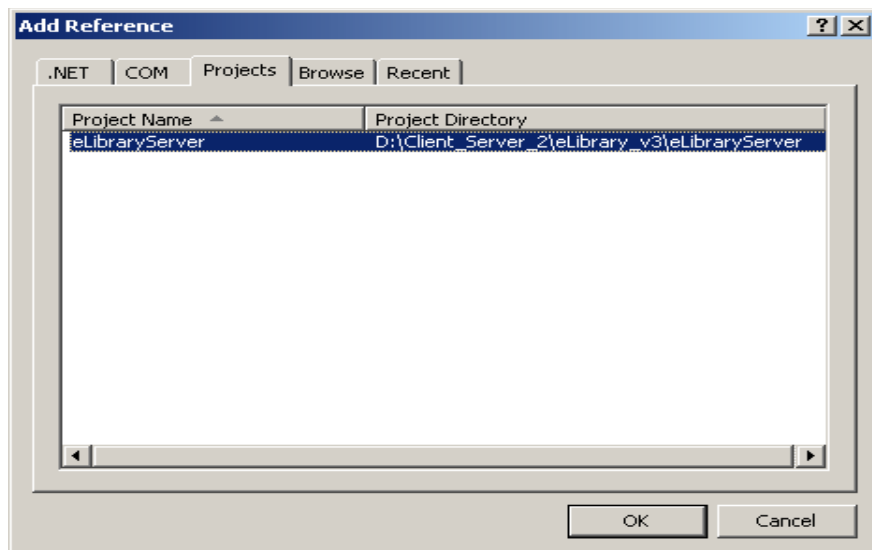
1. Към така създадения (празен засега) сайт, добавете съществуващия проект eLibraryServer от solution eLibrary_v3 чрез менюто:

File->Add->Existing Project

2. Добавете референция към този проект.



Изберете прозореца Projects



СЪЗДАВАНЕ НА САЙТА “ВИРТУАЛНА БИБЛИОТЕКА”

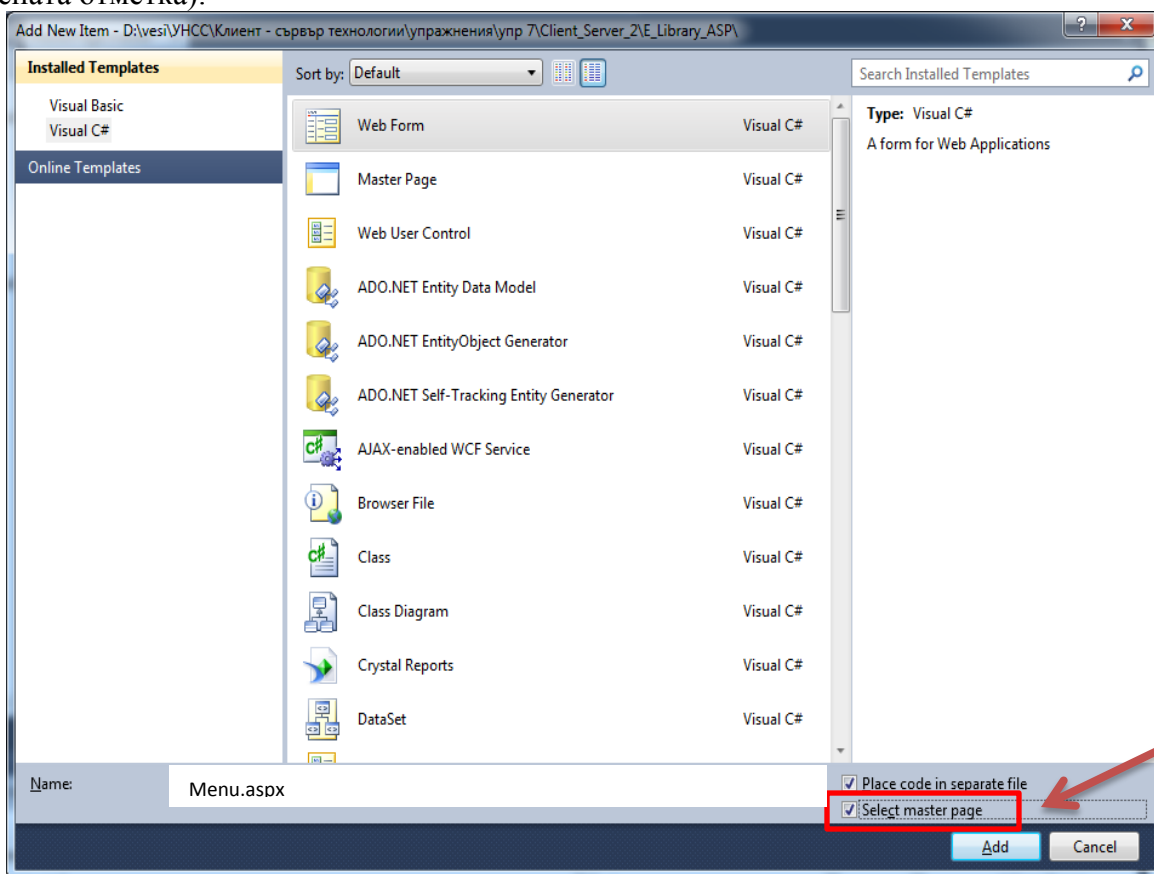
Първо създайте проект на сайта, който трябва да изглежда приблизително както настолното приложение от първата част от семестъра.

Да се създаде Master страница, която да организира интерфейса на приложението. Нека страницата е разделена на 3 части- 2 реда, на 2-рия ред 2 колони.

На първия ред изпишете „Виртуална библиотека“ в подходящ контрол и оформете по ваша преценка.

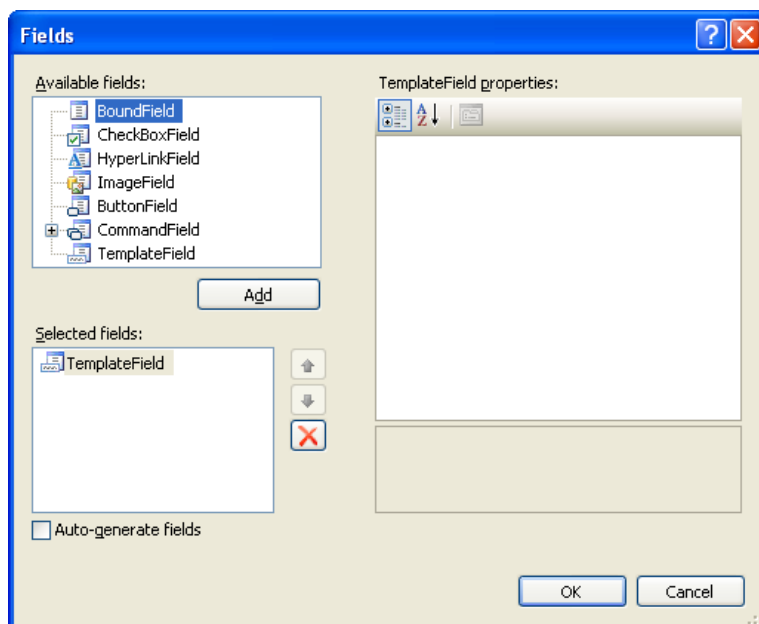
На втория ред и в двете части поставете контроли от типа **ContentPlaceHolder**.

1. Създайте начална страница “Menu.aspx”, която е свързана с масър страницата (поставете дадената отметка):

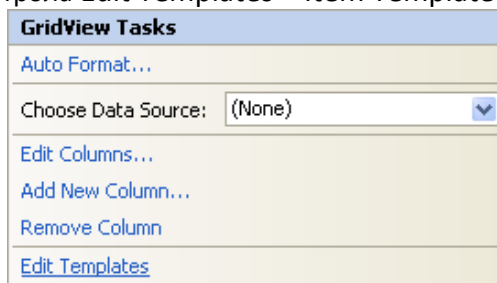


В нея, в лявата част поставете следните контроли :

2. Етикет, в който да се въведе „меню и категории”.
3. Поставете контрол GridView. От GridView изберете Edit Columns, след това добавете TemplateFields и изключете Auto-generate fields.



4. Изберете от smart тага на контрола Edit Templates > Item Template



5. Поставете контрол LinkButton.

6. Изберете Link Button-а в команден режим. Добавете свойството текст и въведете командите:

```
<asp:LinkButton ID="LinkButton1"
    runat="server"
    Text='<%# Eval("BookTypeName")%>'
    OnCommand="Get_Category"
    CommandName='<%#Eval("BookTypeID")%>'>
</asp:LinkButton>
```

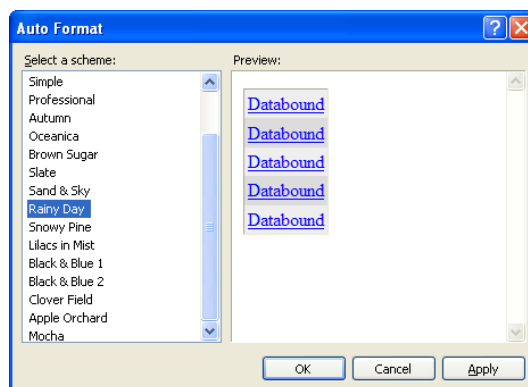
7. Най-отдолу на менюто трябва да има един етикет, където ще се извежда съобщение, в случай, че не може да се установи връзка със сървъра.

```
<asp:Label ID="LabelErr" runat="server" Text="" ></asp:Label>
```

8. Изберете на GridView характеристиката ShowHeader да е False.



9. От smart тага на GridView – то, Auto Format изберете подходящ формат.



10. Добавете код в **Menu.aspx.cs**

```
protected void Page_Load(object sender, EventArgs e)
{
    DataTable bookTypes;
    eLibraryServer.DAL oDAL= new eLibraryServer.DAL();

    try
    {
        bookTypes = oDAL.getBookTypes();
        GridView1.DataSource = bookTypes.DefaultView;
        GridView1.DataBind();
    }
    catch (Exception ex){ LabelErr.Text = "No link to the Data Server!";}
}

public void Get_Category(Object Src, CommandEventArgs Args)
{
    Response.Redirect("Menu.aspx?Category=" + Args.CommandName);
}
```

При зареждане на менюто се създава един нов обект oDAL, на класа eLibraryServer.DAL, който се намира в сървърното приложение. Запълването на менюто става през този клас, който комуникира с базата данни.

При избор на елемент от менюто, управлението се предава на дясната част на страницата **Menu.aspx**, заедно с избраната категория, като се използват аргументите на функцията.

Тествайте менюто.

