

Os exercícios propostos destinam-se a adquirir competências em Programação Web, com C#. A prática só funciona se, de facto, construir uma solução. Não assuma que o docente lhe disponibilizará uma solução. Como tal, deve, por si, realizar os exercícios propostos e interagir com o docente para avaliar as soluções, por si, desenvolvidas.

#### FICHA 12

1. Num novo projecto (Ficha12), crie uma base de dados com o nome “myDB.mdf” e adicione a tabela “Products”:

```
CREATE TABLE [dbo].[Products]
(
    [idprod] INT NOT NULL PRIMARY KEY IDENTITY,
    [name] NVARCHAR(150) NOT NULL,
    [smallDescription] NVARCHAR(150) NOT NULL,
    [description] NVARCHAR(MAX) NOT NULL,
    [price] DECIMAL(18, 3) NOT NULL,
    [image] NVARCHAR(50) NULL
)
```

2. Utilizando o ficheiro “Tabela-Produtos.xls” fornecido pelo docente, introduza os 6 produtos na tabela criada no ponto anterior.
3. Acrescente ao projeto a pasta “images” e o ficheiro “StyleSheet.css”, ambos facultados pelo docente.
4. Adicione o ficheiro “Produtos.aspx” e construa a seguinte estrutura:



[ASUS -  
FX553VD - i7-  
7700HQ](#)

**Price:**  
1 123,99 €

<<

<

>

>>

5. Adicione um controlo do tipo “**DataList**” à sua página e coloque o código anterior na zona assinalada a amarelo.

```
<asp:DataList ID="DataListProdutos" runat="server"
    RepeatLayout="Table"
    RepeatColumns="2"
    RepeatDirection="Horizontal" >

    <HeaderTemplate>

    </HeaderTemplate>

    <ItemTemplate>
        <div class="product" id="product">...</div>
    </ItemTemplate>

    <FooterTemplate>

    </FooterTemplate>
</asp:DataList>
```

**Nota:** coloque os 4 botões logo após o fecho da tag </asp:DataList>.

6. Crie o evento de *click* no botão “**btnfirst**” (Primeira página):

```
protected void btnfirst_Click(object sender, EventArgs e)
{
    pos = 0;

    //Guarda a nova posição
    this.ViewState["vs"] = pos;

    //Executa o método "databind"
    databind();
}
```

7. Crie o evento de *click* no botão “**btnprevious**” (Página anterior):

```
protected void btnprevious_Click(object sender, EventArgs e)
{
    //Atribui o valor da Viewstate à posição
    pos = (int)this.ViewState["vs"];

    //Retira uma posição
    pos -= 1;

    //Guarda a nova posição
    this.ViewState["vs"] = pos;

    //Executa o método "databind"
    databind();
}
```

8. Crie o evento de *click* no botão “**btnnext**” (Página seguinte):

```
protected void btnnext_Click(object sender, EventArgs e)
{
    //Atribui o valor da Viewstate à posição
    pos = (int)this.ViewState["vs"];

    //Incrementa uma posição
    pos += 1;

    //Guarda a nova posição
    this.ViewState["vs"] = pos;

    //Executa o método "databind"
    databind();
}
```

9. Crie o evento de *click* no botão “**btnlast**” (Última Página):

```
protected void btnlast_Click(object sender, EventArgs e)
{
    //vai para a última posição do objecto PagedDataSource e subtrai 1
    pos = adsource.PageCount - 1;

    //Guarda a nova posição
    this.ViewState["vs"] = pos;

    //Executa o método "databind"
    databind();
}
```

10. Método de DataBinding:

```
public void databind()
{
    //Efectua a query e guarda o valor no SqlDataAdapter
    dadapter = new SqlDataAdapter("select * from Products", connstring);

    //Inicializa os objectos que vão ser utilizados
    dset = new DataSet();
    adsource = new PagedDataSource();

    //Definição do número de produtos por página
    adsource.PageSize = 2;

    //O Dataset é preenchido com o conteúdo do SqlDataAdapter
    dadapter.Fill(dset);

    //O conteúdo do Dataset, é colocado no PagedDataSource
    adsource.DataSource = dset.Tables[0].DefaultView;

    //Flag que indica que o PagedDataSource permite paginação
    adsource.AllowPaging = true;

    //Atribuição da posição actual ao PagedDataSource
    adsource.CurrentPageIndex = pos;

    //Activar e Desactivar botões
    btnfirst.Enabled = !adsource.IsFirstPage;
    btnprevious.Enabled = !adsource.IsFirstPage;
    btnlast.Enabled = !adsource.IsLastPage;
    btnnext.Enabled = !adsource.IsLastPage;

    //Popular a DataList com os dados do PagedDataSource
    DataListProdutos.DataSource = adsource;
    DataListProdutos.DataBind();
}
```

11. Introduza o código abaixo, no evento de **Page\_Load**:

```
protected void Page_Load(object sender, EventArgs e)
{
    if (!IsPostBack)
    {
        //Se for a primeira vez que acede, defini o viewstate["vs"] com o valor 0
        this.ViewState["vs"] = 0;
    }

    //Atribui o valor da Viewstate à posição
    pos = (int)this.ViewState["vs"];

    //Executa o método "databind"
    databind();
}
```

12. Adicione o seguinte código complementar:

```
SqlDataAdapter dadapter; //declaracao do objecto SqlDataAdapter
DataSet dset; //declaracao do objecto DataSet
PagedDataSource adsource; //declaracao do objecto PagedDataSource
int pos; //Posicao da página

//ConnectionString
string connstring = System.Configuration.ConfigurationManager.ConnectionStrings["ConnectionString"].ToString();

protected void Page_Load(object sender, EventArgs e) { ... }
```

13. Execute o projeto e verifique o funcionamento.



14. Altere o código para que seja possível visualizar **4 produtos de cada vez**.
15. Altere o código de forma a que seja possível visualizar a lista de produtos conforme imagem abaixo:

