



## Ficha de Trabalho N.º 5

### ASP.NET - WebMatrix

1. Iniciar o WebMatrix
2. Abrir o web site WebPagesMovie.
3. Clicar em Files para abrir o ambiente de trabalho Files.
4. Criar uma nova página com o nome testform.cshtml.
5. Na secção <body> acrescentar o código seguinte:

```
<form method="post">  
  <input type="text" name="name" value="" />  
  <br/>  
  <input type="submit" name="submit" value="Submit" />  
</form>
```

- a. O elemento <form> contém elementos a serem submetidos. O atributo method indica ao browser como tratar os dados inseridos pelo utilizador. Definimos post para atualizar os dados no servidor web e get para “trazer” dados do servidor.
  - b. Elementos <input> permitem a interação do utilizador com o servidor. Têm a seguinte sintaxe:  
<input type="type" name="name">
    - i. Text box: <input type="text">
    - ii. Check box: <input type="checkbox">
    - iii. Radio button: <input type="radio">
    - iv. Button: <input type="button">
    - v. Submit button: <input type="submit">
6. Guardar as alterações e executar a página no browser.
  7. Abrir o ficheiro Filmes.cshtml.

8. Após a tag </h1> inserir o seguinte código:

```
<form method="get">
  <div>
    <label for="searchGenre">Género a pesquisar:</label>
    <input type="text" name="searchGenre" value="" />
    <input type="Submit" value="Pesquisar Género" /><br/>
    (Deixar em branco para pesquisar todos os filmes.)<br/>
  </div>
</form>
```

9. Guardar as alterações e executar a página no browser.

10. Digitar um valor na textbox, por exemplo Ação.

11. Como o atributo method do elemento <form> foi definido para get, o valor digitado faz agora parte da URL da página:

<http://localhost:32122/Filmes.cshtml?searchGenre=Ação>

- É este valor que utilizamos para fazer a pesquisa na base de dados.
- O código Request.QueryString permite obter uma lista de todos os elementos submetidos no form. Para obter um determinado elemento da lista, especifica-se o nome do elemento que queremos:

```
var termoPesquisa = Request.QueryString["searchgenre"];
```

a variável termoPesquisa fica com o valor "Ação".

12. Vamos usar o valor guardado na variável termoPesquisa numa query SQL para seleccionar dados de uma tabela.

- Para obter todos os filmes da base de dados usamos a seguinte query: SELECT \* FROM Filmes
- Para obter os filmes de um determinado género, por exemplo Ação, usamos a seguinte query: SELECT \* FROM Filmes WHERE Género = 'Ação'
- Queremos fazer uma query que funcione para qualquer Género que o utilizador especifique:

```
SELECT * FROM Filmes WHERE Género = @0
```

O carater @ seguindo do número 0 (zero) especifica um parâmetro na query. Uma query pode conter vários parâmetros que serão especificados por @0, @1, @2, etc.

Para elaborar a query e passar-lhe o valor que o utilizador digita na caixa de texto do formulário utilizamos o código seguinte:

```
termoPesquisa = Request.QueryString["searchgenre"]
```

```
selectCommand = "SELCET * FROM Filmes WHERE Género = @0";
```

```
selectData = db.Query(selectComand, termoPesquisa);
```

13. Atualizar o código da página Filmes.cshtml para que fique igual a este:

```
@{
    var db = Database.Open("WebPagesMovies") ;
    var selectCommand = "SELECT * FROM Filmes";
    var termoPesquisa = "";

    if(!Request.QueryString["searchGenre"].IsEmpty() ) {
        selectCommand = "SELECT * FROM Filmes WHERE Género = @0";
        termoPesquisa = Request.QueryString["searchGenre"];
    }

    var selectedData = db.Query(selectCommand, termoPesquisa);
    var grid = new WebGrid(source: selectedData, defaultSort: "Género", rowsPerPage:3);
}
```

- a. Sempre que a página é executada, abrimos a base de dados e selecionamos todos os filmes.
- b. A variável termoPesquisa é inicializada.
- c. A linha `if(!Request.QueryString["searchGenre"].IsEmpty() )`, verifica se a caixa de texto de pesquisa está ou não vazia, funcionando da seguinte forma:
  - i. Obtém o valor de `Request.QueryString["searchGenre"]`, ou seja o valor que foi digitado pelo utilizador no elemento do form searchGenre;
  - ii. Verifica se está vazio usando o método `IsEmpty`, que verifica se, por exemplo, um elemento do form está vazio;
  - iii. Como queremos executar código quando não está vazio, acrescentamos o operador ! (NOT); traduzindo: se a caixa de texto searchGenre não estiver vazia então...
- d. Se o utilizador especificou um valor para a seleção por Género, a query a executar será uma que inclui uma cláusula WHERE.
- e. Finalmente é inicializado o controlo webgrid usando o resultado da query.

14. Podemos também acrescentar a funcionalidade de pesquisar todos os filmes que tenham uma determinada palavra no título, executando a query apresentada:

`SELECT * FROM Filmes WHERE Título LIKE '%Wind%'`, em que estamos a selecionar todos os filmes que tenham a palavra Wind no Título.

15. Acrescentar o seguinte código a seguir ao </div> da pesquisa por Género:

```
<div>
  <label for="SearchTitle">O título do filme contém:</label>
  <input type="text" name="searchTitle" value="@Request.QueryString["searchTitle"]" />
  <input type="Submit" value="Pesquisar título" /><br/>
</div>
```

16. O código para esta pesquisa é semelhante ao código da pesquisa por Género, exceto na cláusula LIKE.

Acrescentar o seguinte código, no topo da página, logo a seguir ao bloco if da pesquisa por Género:

```
if(!Request.QueryString["searchTitle"].IsEmpty() ) {
  selectCommand = "SELECT * FROM Filmes WHERE Título LIKE @0";
  searchTerm = "%" + Request.QueryString["searchTitle"] + "%";
}
```