

MÓDULO

JAVASCRIPT/AJAX

UNIDADE

INTERAÇÃO DE AJAX/JAVASCRIPT COM
OUTRAS LINGUAGENS DE PROGRAMAÇÃO
(ASP, PHP, JSP)

ÍNDICE

OBJETIVOS.....	3
INTRODUÇÃO.....	4
1. CHAVES E VANTAGENS	5
CONCLUSÃO	13
AUTOAVALIAÇÃO	15
SOLUÇÕES.....	19
PROPOSTAS DE DESENVOLVIMENTO DO ESTUDO.....	20
BIBLIOGRAFIA	21

OBJETIVOS

Com esta unidade didática, pretende-se que desenvolva os seguintes objetivos de aprendizagem:

- Aprender como interagir com o código de uma linguagem de programação web (ASP e PHP) com o JavaScript.
- Conhecer as vantagens de usar duas linguagens.

INTRODUÇÃO

O JavaScript é uma linguagem de programação web que funciona no lado do cliente, ou seja, o servidor serve os dados que são interpretados pelo navegador e é este que executa os códigos do script.

Além de HTML e JavaScript, as páginas web profissionais utilizam linguagens de programação de servidor como PHP, ASP, JSP, etc., que permitem que o código seja executado no servidor e exiba o resultado no cliente. A vantagem destas linguagens é que são mais potentes do que o JavaScript e com mais opções, como a leitura de banco de dados, leitura de ficheiros no servidor, entre outros.

Nesta unidade, verá como estas linguagens e o JavaScript podem interagir.

1. CHAVES E VANTAGENS

As linguagens de programação web para os servidores (ASP, PHP) executam o código no servidor e devolvem o código HTML gerado ao navegador do cliente, ou seja, depende do JavaScript o poder de interagir com o utilizador no website.

O primeiro dado que é necessário saber é que estas são linguagens completamente diferentes, e variáveis, funções, constantes e outros componentes não podem ser partilhados diretamente. As linguagens de servidor, quando compiladas no próprio servidor, não podem aceder aos dados JavaScript, ou seja, em ASP, por exemplo, não se pode referir aos dados JavaScript. Porém, numa página web pode referir-se a uma variável ASP dentro do código JavaScript.

A explicação é simples: como o ASP é executado no servidor, quando o servidor atinge a linha JavaScript que contém a variável ASP, interpreta o valor dessa variável e atribui esse valor no código. Segue-se um exemplo para clarificar.

Uma função JavaScript que mostra um alert com a soma de duas variáveis ASP ou PHP, seria possível com o seguinte código:

Em ASP:

```
<%  
  
var1 =4  
  
var2 =6
```

```
%>

function soma(){
    alert('Valor da soma <%= (var1 + var2)%>');
}
```

Em PHP:

```
<?PHP

$var1 =6;
$var2 =4;

?>

function soma(){
    alert('<?echo "Valor da soma ".(var1 + var2)?>');
}
```

Como pode observar nos exemplos, é possível inserir valores em JavaScript, mas também em qualquer outro tipo de código.

O exemplo a seguir mostra um alert em JavaScript, mas o conteúdo depende de uma variável de idioma do servidor:

Em ASP:

```
function mostrar(){
    alert('<%if v = "s" thenresponse.write("ok") elseresponse.write("
ko")%>');
}
```


Em PHP:

```
function mostrar(){  
    alert('<? if (v = "s") { echo "ok" }else{ echo "ko" } ?>');  
}
```

A única opção para interagir de forma oposta, ou seja, usar ASP ou PHP para controlar o JavaScript, é através das condições das linguagens do servidor. Não é possível utilizar uma variável JavaScript no ASP, mas pode utilizar-se uma instrução if do ASP para controlar se é ou não executado um script JavaScript. Segue-se um exemplo que mostra um alert se a variável de idioma tiver o valor "sim":

Em ASP:

```
<%  
variable = "sim"  
  
if variable = "sim"  
thenResponse.write("<script>alert('Teste')</script>") endif  
%>
```

Em PHP:

```
<?  
$variable = "sim";  
  
if($variable = "sim"){  
    echo ("<script>alert('Teste')</script>");  
}  
?>
```

A principal desvantagem das linguagens de servidor como ASP e PHP é que, se pretender executar código para devolver alguns registos ou fazer qualquer tipo de processo, é necessário fazer uma chamada para o servidor e atualizar a página para que devolva os novos dados.

No entanto, graças ao AJAX, isso já não é uma desvantagem. A principal vantagem de integrar linguagens de servidor com JavaScript é o uso do AJAX, que torna possível executar o código ASP ou PHP que pretende numa página web no evento HTML que desejar sem que precise de atualizar a página.

Esta é a técnica mais utilizada pelos web developers, pois permite paginar resultados sem ter de atualizar a página web ou criar campos de pesquisa capazes de prever sem ter de atualizar.

Para isso, na chamada do objeto AJAX **XMLHttpRequest**, a URL será a página da web em PHP que pretende carregar. Este objeto irá devolver um resultado que pode carregar em qualquer elemento da página web.

Exemplo 01

Este exemplo mostrará diferentes opções de filme de acordo com o ano que se selecionar a partir de uma lista suspensa, utilizando uma função AJAX no evento **onchange** da lista suspensa para ler os dados de um ficheiro PHP chamado "dados.php" (este pode chamar um banco de dados para obter os dados; neste caso, consiste num seletor que devolve um ou outro dado). Este exemplo necessita de um servidor web para funcionar corretamente. Os ficheiros HTML e PHP deverão ser gravados na pasta htdocs do XAMPP ou similar.

dados.php:

```
<?php

$ano = $_GET['ano'];

if ($ano == "2015") {

    echo "Sniper Americano<br>Mad Max<br>Creed: O Legado de Rocky";

} elseif ($ano == "2016") {

    echo "Dheepan<br>Bons Rapazes<br>O Ornitólogo";
```

```
} elseif ($ano == "2017") {  
    echo "Paterson<br>O Quadrado<br>Coco";  
}  
} elseif ($ano == "2018") {  
    echo "Guerra Fria<br>O Lamento<br>Linha Fantasma";  
}  
} elseif ($ano == "2019") {  
    echo "Todos Sabem<br>A Favorita<br>Correio de Droga";  
}  
} else {  
    echo "Retrato da Rapariga em Chamas<br>Os  
Miseráveis<br>Mulherzinhas";  
}  
?>
```

unidade26.html:

```
<!DOCTYPE html>  
  
<html lang="pt">  
  
<head>  
    <meta charset="utf-8">  
  
<head>  
    <title>Unidade 26 - Exemplo 01</title>  
  
    <style>  
        #caixa {  
            width: 400px;  
            height: 400px;
```

```
        border: 1px solid #FC0;

        position: relative;

    }
</style>

<script type="text/JavaScript">

function carregar(ano) {
    var objHttp = null;

    if(window.XMLHttpRequest) {

        objHttp = new XMLHttpRequest();

    } else if(window.ActiveXObject) {

        objHttp = new ActiveXObject("Microsoft.XMLHTTP");

    }

    objHttp.open("GET", "dados.php?ano=" + ano , true);
    objHttp.onreadystatechange = function() {

        if (objHttp.readyState == 4) {

            document.getElementById('caixa').innerHTML =
objHttp.responseText;

        }

    }

    objHttp.send(null);

}

</script>

</head>

<body>
```

```
Filmes de

<select onchange="carregar(this.value)" name="ano">

  <option value="">Selecione um ano</option>

  <option value="2015">Ano 2015</option>

  <option value="2016">Ano 2016</option>

  <option value="2017">Ano 2017</option>

  <option value="2018">Ano 2018</option>

  <option value="2019">Ano 2019</option>

  <option value="2020">Ano 2020</option>

</select>

<div id="caixa">

  A lista de filmes do ano selecionado irá aparecer
  nesta caixa.

  </div>

</div>

</body>

</html>
```

A visualização inicial da página, quando carregada será:

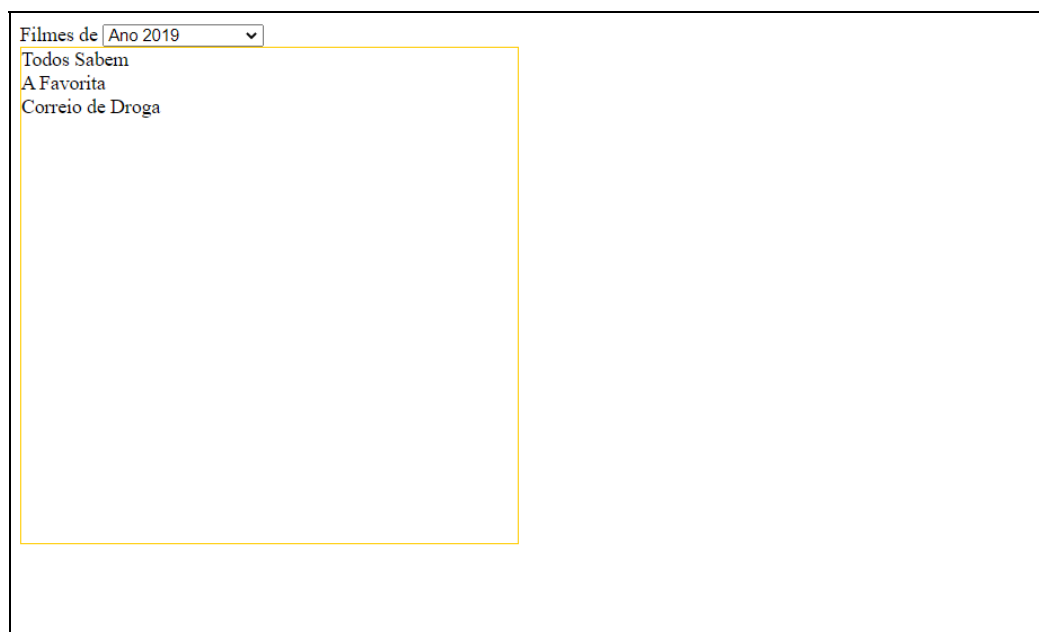


Filmes de Seleção um ano ▼

A lista de filmes do ano selecionado irá aparecer nesta caixa.

Visualização da página quando carregada.

Ao escolher um ano, surgirá na caixa a lista de filmes referentes a esse ano, tal como se exemplifica na figura abaixo.



Filmes de Ano 2019 ▼

Todos Sabem
A Favorita
Correio de Droga

Visualização da página quando é selecionado o ano 2019.

CONCLUSÃO

A integração de JavaScript com as linguagens de servidor permite dar poder de carregamento às páginas web (já que não é necessário atualizar) e facilita a interação com os utilizadores.

AUTOAVALIAÇÃO

- 1. Qual é a diferença entre as linguagens de servidor e o JavaScript?**
 - a) O JavaScript é executado no navegador e as outras linguagens no servidor.
 - b) As linguagens do servidor consomem mais recursos.
 - c) O JavaScript é mais poderoso do que as linguagens de servidor.
 - d) As linguagens de servidor são mais potentes.

- 2. Qual das seguintes afirmações está incorreta?**
 - a) Os componentes do JavaScript e linguagens de servidor não podem ser partilhados diretamente.
 - b) JavaScript é executado no servidor e fornece os dados ao navegador.
 - c) As linguagens de servidor não podem exibir dados sem solicitá-los ao servidor.
 - d) Numa página web, é possível referir-se a uma variável ASP em JavaScript.

- 3. O que é que pode ser inserido no código JavaScript?**
 - a) Referências à variável das linguagens de servidor.
 - b) Funções da linguagem de servidor.
 - c) Referências às variáveis e funções da linguagem de servidor.
 - d) É possível inserir tudo o que se pretender.

4. Qual é a principal vantagem de integrar AJAX com linguagens de servidor?
- a) A capacidade de criar efeitos visuais.
 - b) Obter uma velocidade de carregamento mais rápida de uma página web.
 - c) A capacidade de trazer os dados do servidor com uma página PHP ou ASP sem a necessidade de atualizar a página web.
 - d) A capacidade de ler XML.
5. Qual é o objeto AJAX utilizado para fazer chamadas a páginas da linguagem de servidor?
- a) ObjRequest.
 - b) httpRequest.
 - c) XMLHttpRequest.
 - d) Não é possível fazer chamadas a páginas da linguagem de servidor.
6. Qual é a única opção para incorporar código JavaScript em ASP ou PHP?
- a) Fazer referência às variáveis.
 - b) Criar variáveis em ficheiros externos para ler os mesmos.
 - c) Chamar as variáveis JavaScript da mesma forma que as variáveis do ASP ou PHP.
 - d) Usar uma condição ASP ou PHP para controlar a execução do código JavaScript.
7. Qual das seguintes instruções está correta?
- a) alert('<%= (3+4) %>').
 - b) alert('<%= (3+4) %>').
 - c) alert('<= (3+4) >').
 - d) alert('<Php= (3+4) %>').

- 8. Qual das seguintes instruções está correta?**
- a) `alert('<phpif (v = "s") { echo "ok" }else{ echo "ko" } >')`.
 - b) `alert('<% if (v = "s") { echo "ok" }else{ echo "ko" } ?>')`.
 - c) `alert('<if (v = "s") { echo "ok" }else{ echo "ko" } >')`.
 - d) `alert('<? if (v = "s") { echo "ok" }else{ echo "ko" } ?>')`.
- 9. Qual é a principal desvantagem das linguagens de servidor como ASP e PHP?**
- a) ASP e PHP não devolvem dados sem atualizar a página web.
 - b) ASP e PHP não têm acesso a dados em formato XML ou JSON.
 - c) ASP e PHP não funcionam em todos os navegadores.
 - d) ASP e PHP não têm nenhuma desvantagem.
- 10. Qual é o objeto JavaScript utilizado para integrar AJAX com as linguagens de servidor?**
- a) XMLHttpRequest.
 - b) XMLHttpRequest.
 - c) RequestData.
 - d) XMLHttpRequest.

SOLUÇÕES

1.	a	2.	b	3.	c	4.	c	5.	c
6.	d	7.	a	8.	d	9.	a	10.	b

PROPOSTAS DE DESENVOLVIMENTO DO ESTUDO

O poder do que aprendeu nesta unidade depende do domínio das linguagens de servidor como ASP ou PHP. Sugerimos os seguintes recursos, para aprender mais sobre estas linguagens:

- Para PHP (além do módulo seguinte desta formação: “Programação PHP”):
 - Gutiérrez, J. D. (2010). *Desarrollo Web con PHP 6 y MySQL 5.1*. Madrid: AnayaMultimedia.
- Para ASP:
 - Shepherd, G. (2010). *ASP.net 4.0*. Madrid: AnayaMultimedia

BIBLIOGRAFIA

- Babin, L. (2010). *Introducción a Ajax con PHP*. Madrid: AnayaMultimedia.

