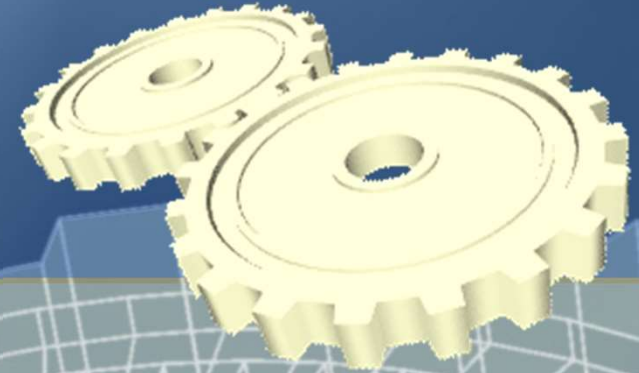


# PHP

## Funções particulares e Segurança

### Redes de Comunicação – Módulo V



# função Header()

Existem algumas **funções especiais** que iremos utilizar com frequência

A função **header()** é uma dessas funções

Utilização:

```
header("Location: página_de_destino.php");
```

Exemplo:

Num processo de **Login** tradicional se o utilizador é do **tipo administrador** será encaminhado para a página **admin.php**, caso seja um **utilizador normal** será enviado para a página **main.php**

```
(...)  
if ($tipo=="admin")  
    header("Location: admin.php");  
elseif ($tipo=="regular")  
    header("Location: main.php");  
(...)
```



# Include

A linguagem PHP permite **inserir código** contido em **outros ficheiros**

Através da função **include()** é possível **omitir código redundante**

```
include("nome_do_fx_com_código.php");
```

Num **Web site** é recorrente as páginas que o constituem **conterem** elementos **repetitivos** como **cabeçalho**, **rodapé**, **menu**, **logótipo**, etc

Com a ajuda do **include()** é possível ao programador poupar tempo e trabalho em **futuras alterações** bem como apresentar um **código** mais **esclarecedor**

## **Exemplo:**

```
<?php  
    include("logotipo.php");// página onde se encontra o código do logotipo.  
    include("menu.php");// página onde se encontra o código do menu.  
    echo "Aqui ficará a parte do conteúdo do site. Apenas esta parte será diferente.";  
    (...)  
?>
```

Existe uma função em PHP que **testa** se uma determinada **variável** está ou não **definida**

Essa função chama-se **isset()** e retorna **verdadeiro** ou **falso** conforme a variável se encontra ou não definida

```
isset($nome_da_variável_a_testar)
```

**Exemplo:**

```
<?php  
if(isset($utilizador))  
    header("Location: utilizador_registado.php");  
else  
    header("Location: visitante.php");  
?>
```



# Cookies e Session

Os sites desenvolvidos em PHP **acedem** quase sempre a **BD**

Certas páginas do Web Site contêm **informação confidencial**, pelo que é necessário **proteger** o seu **conteúdo**

Através de **cookies** e **Sessions** é possível **restringir** o **acesso** a páginas **privadas** do nosso web **site**

Contudo, existem algumas diferenças entre **cookies** e **Sessions**:

## **Cookie:**

- armazenadas **localmente**
- tempo limite de vida (**expiram**)
- **baixa** segurança

## **Session:**

- armazenadas no **servidor**
- **não expiram** a não ser que o programador assim o determine
- maior **segurança**

# Cookies

As **cookies** têm **obrigatoriamente** de ser **declaradas antes** da tag **<HTML>**

```
setcookie(name, value, expire, path, domain, secure)
```

**Name:** nome da Cookie

**Value:** valor a guardar temporariamente

**Expire:** tempo em segundos de duração da cookie

**Path:** automático

**Domain:** automático

**Secure:** 1 para ligação segura(HTTPS) ou 0 para não segura

Exemplo (**criar cookie**):

```
<?php
    setcookie("uti", "$nome", time()+60);//expira após 1 minuto
?>
```

Exemplo (**eliminar cookie**):

```
<?php
    setcookie("uti", "$nome", time()-60);
?>
```



# Acesso a Cookies

Após a **criação** das **cookies** é possível **aceder** a esses **dados** através de um **array associativo** específico

No caso das cookies:

```
$_COOKIE['nome_da_cookie']
```

Exemplo:

```
<?php
    (...)
    $nome = $_POST['utilizador'];
    setcookie("uti", "$nome", time()+60); //expira após 1 minuto
    echo $_COOKIE['uti'];
    (...)
?>
```

A utilização de **sessões** permite **maior controlo** e **segurança** face à utilização de cookies

Da mesma forma que as Cookies têm de ser **declaradas antes** da tag **<HTML>**

**session\_start();** // inicia uma sessão

**unset(\$variável\_de\_sessão\_a\_destruir);** //destrói uma variável de sessão específica

**session\_destroy();** // destrói completamente todas as sessões existentes

Exemplo (**iniciar Sessão**):

```
<?php
    session_start();
?>
<html>
    <body>
    </body>
</html>
```

Exemplo (**destruir Sessão**):

```
<?php
    session_start();
    session_destroy();
?>
```



# Acesso a Cookies e Sessions

O **array associativo** onde são **guardadas** as **variáveis** de **sessão** é `$_SESSION[]`

**Utilização:**

```
$_SESSION['nome_da_sessão']
```

**Exemplo:**

```
<?php
```

```
    session_start();
```

```
    $_SESSION['user']=$_POST['user']; //criação de uma variável de sessão
```

```
?>
```

# Acesso a Cookies e Sessions

A grande vantagem destas **variáveis** (Cookies ou Sessions) é serem **universais**

As **variáveis locais** e **globais** apenas **existem** numa **página** específica do nosso Web Site, isto é na página **onde são declaradas**

Por outro lado, as **variáveis** do tipo **cookies** e **sessions existem** em **qualquer página do nosso Web site**, daí designarem-se por variáveis universais

Podem ser importantes, por exemplo, para **impedir** que **utilizadores não registados** acessem a **páginas reservadas** do nosso Web Site

Para o **utilizador** que tenha **efetuado** o **login** será **criada** uma **variável de sessão** ou **cookie**

**Caso contrário não há variável de sessão** ou **cookie** criada **impedindo** o **utilizador** de **aceder** a uma página **privada**





**Final da Apresentação**