

Cookies e sessões

- HTTP é um protocolo que não mantém estado.
- Cada pedido que um browser faz ao servidor web é independente do pedido anterior.
- Muitas aplicações necessitam de manter o estado ao saltar de página em página.
- Exemplo: carrinho de compras num site de comércio electrónico.

Aplicações que mantêm estado

- Pode-se guardar informação de estado no browser do cliente (utilizando cookies).
- Pode-se guardar informação de estado no servidor web (utilizando sessões).

Cookies

- São uma espécie de variáveis que ficam guardadas no browser.
- Cookies são enviadas pelo servidor web através do protocolo HTTP.
- Cookies são enviados em futuros pedidos do browser para esse servidor web.

Cookies (cont.)

- Cookies podem ter parâmetros que:
 - façam com que expire a uma determinada data/hora.
 - restringe o seu envio a um determinado domínio.
 - não permita o seu uso no acesso a determinados directórios dentro do site.

Cookies com PHP

- PHP tem a função `setcookie` que gera o cabeçalho HTTP.
- as cookies ficam disponíveis através do array associativo `$_COOKIE`

Limitações das cookies

- Browser só pode guardar até 20 cookies enviadas de um determinado domínio.
- Cookies não podem ter mais do que 4KB.
- Utilizadores podem desactivar cookies no browser.

Sessões

- Permite guardar informação de estado (variáveis) no servidor.
- Em vez do browser guardar informação sobre várias variáveis, guarda apenas um “session id” .
- Esse “session id” permite aceder a variáveis associadas à sessão.

Sessões (cont.)

- Variáveis de sessão têm de ser guardadas no servidor web (por exemplo, em ficheiros de texto ou numa base de dados).
- Sessões necessitam de ter um “timeout”. Caso contrário, o servidor web poderá não saber se uma sessão terminou ou não.

Sessões com PHP

- função `session_start()` cria uma nova sessão ou identifica uma sessão já estabelecida.
- Quando um script PHP chama `session_start()` pela primeira vez, é gerado um “session id”.
- Por defeito, esse “session id” é enviado como cookie para o browser com o nome de `PHPSESSID`
- o “session id” é uma string aleatória de 32 dígitos hexadecimais, tal como `fcc17f071bca934ae2f24f290343c3b7`

Sessões com PHP (cont.)

- Depois de se fazer `session_start()` podemos guardar ou aceder a variáveis de sessão.

```
$_SESSION["equipa"] = "Benfica";
```

- Uma vez registadas, as variáveis ficam disponíveis para todos os scripts que chamem `session_start()`
- No exemplo, `equipa` é uma variável de sessão que fica guardada no servidor web com o valor ‘Benfica’.

Sessões com PHP (cont.)

- Variáveis podem ser removidas da sessão atribuindo-lhes o valor `FALSE`

```
$_SESSION["equipa"] = FALSE;
```

- `session_id()` devolve o “session id” da sessão.
- `session_destroy()` elimina todas as variáveis associadas à sessão.

Sessões com PHP (cont.)

- Se o browser desactivar as cookies, tem de se passar PHPSESSID na URL.
- Para ver se a cookie está activa, pode-se verificar o valor de `$_COOKIE["PHPSESSID"]`
- Se não estiver activa, temos de de passar o 'session id'. Exemplo:

```
$url = "page.php?PHPSESSID=".session_id();
```

- Em alternativa, pode-se usar a constante SID que corresponde a `"PHPSESSID=".session_id()`

Sessões com PHP (cont.)

- Uma outra alternativa é configurar o motor de PHP para incluir "PHPSESSID=". `session_id()` automaticamente em todos os links.
- Chama-se a isto *URL rewriting*
- Tem de se editar o ficheiro `php.ini` e colocar `session.use_trans_sid = 1`
- Obviamente que só poderão fazer isto se forem administradores do servidor web.