**Efetuando o login e cookies**

Legal, temos um formulário de login onde digitamos o usuário e senha e ao enviarmos ele mostra uma mensagem de sucesso ou fracasso.

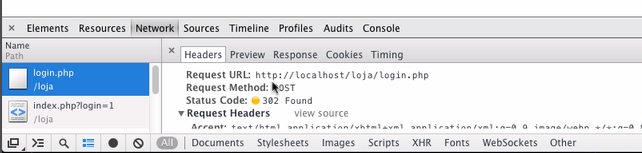
Agora eu gostaria de mostrar uma mensagem de logado e sumir com o formulário enquanto você está logado. Isto é, gostaria de saber se você é o rapaz que se logou ou não. Em uma segunda requisição, eu tenho que saber que você é aquela pessoa que havia se logado em outra requisição anterior.

Para fazer isso tenho que conseguir marcar você, cliente, com alguma informação. Tenho que deixar minha marca em você, algo como "usuario\_logado = guilherme.silveira@alura.com.br" para que você em todas as requisições subsequentes envie essa marca de volta, me dizendo que você é o usuario logado **guilherme.silveira@alura.com.br**. A sacada aqui é conseguir entregar algo para você que deixe essa marca, e você envie de volta essa marca toda vez que fizer uma requisição. Quero deixar um cookie com você, e você enviar de volta esse cookie.

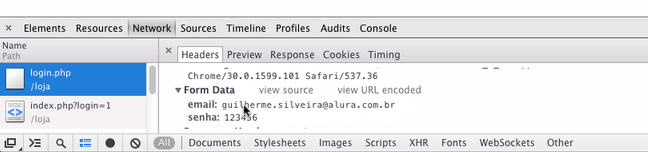
Vamos então ao **login.php** e fazemos no caso de login com sucesso:

setcookie("usuario\_logado", $usuario["email"]);

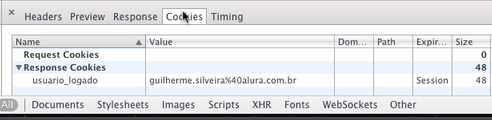
Abrimos o navegador e a aba de desenvolvimento (View, Developer, Developer Tools, Network no caso do Google Chrome). Fazemos o login e ele mostra a mensagem de logado com sucesso. Se olharmos o resultado, temos o resultado de nosso POST como um 302, redirect:



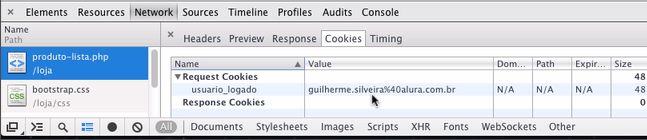
Descendo, encontramos as informações do formulário enviado:



Mas e se escolhermos o item **Cookies**?



Podemos ver nosso cookie com a informação **usuario\_logado = guilherme.silveira@alura.com.br**. Se clicarmos em uma requisição podemos ver que sempre, como cliente, passamos a enviar o cookie:



Se eu estou no servidor, consigo então verificar se o cliente enviou esse cookie. Vou então na minha página**index.php** e verifico se a variável está setada:

if(isset($\_COOKIE["usuario\_logado"])) {

}

No caso de sucesso quero mostrar o email do usuário logado usando **$\_COOKIE["usuario\_logado"]** logo após a mensagem de bem vindo:

<?php

if(isset($\_COOKIE["usuario\_logado"])) {

?>

<p class="text-success">Você está logado como <?= $\_COOKIE["usuario\_logado"] ?></p>

<?php

}

?>

Fazemos o teste com o cookie no nosso navegador e vemos:



Agora vamos remover o formulário caso você esteja logado. Isto é, só mostramos o formulário caso ele não esteja logado:

<?php

if(isset($\_COOKIE["usuario\_logado"])) {

?>

<p class="text-success">Você está logado como <?= $\_COOKIE["usuario\_logado"] ?></p>

<?php

} else {

?>

CODIGO DO FORMULARIO AQUI

<?php

}

?>

Pronto, testamos novamente e conseguimos mostrar a mensagem sem o formulário:



Mas e para testar o formulário deslogado? Temos que esperar o cookie expirar, ele deixar de ser válido. O padrão de um novo cookie é até o navegador ser fechado, enquanto isso não acontecer o mesmo cookie será enviado para o servidor. Fecho o navegador e acesso novamente, pronto, estou deslogado e vejo o formulário de login.

Mas e se eu quiser um tempo de expiração diferente deste, por exemplo 1 minuto. Quando criamos nosso cookie com o setcookie, dizemos o tempo:

setcookie("usuario\_logado", $usuario["email"], time() + 60);

Testamos o login e podemos no network verificar que o cookie expirará em 1 minuto. Esperamos e testamos novamente, estamos deslogados.

Vimos como marcar um usuário com uma determinada informação, que fica do lado do cliente. Toda nova requisição ele envia todos os dados novamente para nós. Quanto maior essa informação, mais dados enviados de lá para cá (cliente para o servidor) e vice versa. Além disso, tudo que salvamos é uma string, não é número, não é objeto, não é data. Os cookies são strings e, portanto, chatos de manipular. Além disso tudo temos probleminhas de segurança quando abordamos assim com cookies que guardam o email. Veremos como melhorar tudo isso nos próximos capítulos, mas primeiro vamos proteger nossa página, no próximo capítulo veremos como impedir que o usuário adicione um produto caso ele não esteja logado.