### **Cookies e Sessions em PHP**

**Cookies:**

* **O que são?** Cookies são pequenos pedaços de dados que um site armazena no computador de quem está visitando. Esses dados são enviados pelo servidor e guardados pelo navegador do usuário.
* **Para que servem?** Eles são usados para lembrar informações sobre o usuário entre uma visita e outra. Por exemplo, quando você volta a um site e ele ainda lembra seu nome, é provável que ele esteja usando um cookie.
* **Limitações dos Cookies:** Cookies não são seguros para guardar informações importantes, como senhas ou dados confidenciais, porque qualquer pessoa pode ver ou modificar o cookie no computador onde ele está guardado.

**Exemplo de Cookie:**

Vamos imaginar um site de loja online. Quando você adiciona produtos ao carrinho, essa informação pode ser guardada em um cookie. Da próxima vez que você visitar a loja, os produtos ainda estarão no seu carrinho, porque o site lembrou de você usando o cookie.

**Como definir um Cookie em PHP:**

**setcookie("nomeDoCookie", "valor", time() + (86400 \* 30)); // 86400 = 1 dia**

Nesse exemplo, o cookie chamado nomeDoCookie vai guardar o valor "valor" por 30 dias.

**Sessões:**

* **O que são?** Sessões são uma maneira de armazenar informações que precisam ser usadas durante a navegação do usuário, mas não ficam salvas no computador dele como os cookies. Em vez disso, as sessões guardam os dados no servidor.
* **Como funcionam?** Quando uma sessão começa, o PHP cria um identificador único, que é como um nome que identifica a sessão de cada usuário. Esse identificador pode ser armazenado em um cookie ou ser passado na URL.
* **Vantagens das Sessões:** Diferente dos cookies, as sessões são mais seguras, pois os dados são armazenados no servidor e não podem ser acessados diretamente pelo usuário.

**Exemplo de Sessão:** Imagine um site onde você faz login. Quando você entra, o site cria uma sessão para você. Enquanto estiver navegando no site, essa sessão guarda as informações que dizem que você está logado. Se fechar o navegador e voltar depois, a sessão vai terminar, e você terá que fazer login novamente.

**Como iniciar uma Sessão em PHP:**

**session\_start();**

**$\_SESSION["usuario"] = "orlando";**

Nesse exemplo, a sessão armazena o valor "orlando" em uma variável chamada usuario.

### Arquivo login informações **. Arquivo de Login (index.php)**

* **Parte PHP:**
  + **Verifica o método de envio do formulário:** Se for POST, inicia uma sessão.
  + **Checa as credenciais:** Compara o username e password com valores fixos (orlando e 123mudar).
  + **Inicia uma sessão:** Se as credenciais estiverem corretas, define uma sessão como loggedin = TRUE e redireciona para a página welcome.php.
  + **Caso contrário:** Define loggedin = FALSE e não faz nada além disso.
* **Parte HTML:**
  + **Formulário de Login:** Exibe um formulário simples onde o usuário insere o nome de usuário e a senha.

### **2. Arquivo de Boas-vindas (welcome.php)**

* **Parte PHP:**
  + **Verifica se o usuário está logado:** Se a sessão loggedin não estiver definida ou for FALSE, redireciona para a página de login.
* **Parte HTML:**
  + **Mensagem de boas-vindas:** Exibe uma mensagem de boas-vindas ao usuário logado, mostrando o nome armazenado na sessão ($\_SESSION["username"]).
  + **Links para outras páginas:** Inclui links para a página de cadastro, visualização de pessoas cadastradas, e para fazer logout.

### **3. Arquivo de Cadastro (cadastro.php)**

* **Parte HTML:**
  + **Formulário de Cadastro:** Exibe um formulário onde o usuário pode inserir seu nome e CPF. Esse formulário é enviado para a página cadastro\_nome\_cpf.php.

### **4. Arquivo para Salvar Cadastro (cadastro\_nome\_cpf.php)**

* **Parte PHP:**
  + **Verifica se os dados foram enviados:** Se os campos nome e cpf estiverem preenchidos, continua a execução.
  + **Abre ou cria um arquivo para salvar os dados:** Verifica se o arquivo cadastros.txt existe. Se não existir, cria um novo. Caso exista, abre-o em modo de adicionar (append).
  + **Salva os dados:** Escreve o nome e CPF no arquivo cadastros.txt, cada cadastro separado por uma nova linha.
  + **Exibe confirmação e conteúdo:** Exibe uma mensagem de sucesso e mostra todo o conteúdo do arquivo cadastros.txt.

### **5. Arquivo para Mostrar Pessoas Cadastradas (visualizacao.php)**

* **Parte PHP:**
  + **Abre o arquivo e exibe os dados:** Verifica se o arquivo cadastros.txt existe. Se existir, abre-o e exibe seu conteúdo linha por linha. Se não existir, informa que o arquivo não foi encontrado.

### **6. Arquivo de Logout (logout.php)**

* **Parte PHP:**
  + **Similar ao visualizacao.php:** Este código parece um erro, pois deveria destruir a sessão para fazer logout, mas ele está tentando exibir os dados do arquivo cadastros.txt. Se fosse para fazer logout, o código deveria destruir a sessão e redirecionar para a página de login. Provavelmente, este código foi copiado incorretamente do visualizacao.php.