

FACOM32603 – Desenvolvimento Web 1

Prof. Ian R. Cunha

Exercício de prática e fixação PHP

formulários – sessões – banco de dados

*Pode ser feito em duplas, e será avaliado com pontuação extra (a ser definida).

1: Construindo a Interface (Os Formulários HTML)

- Crie um arquivo **form_registro.php**. Ele deve conter um formulário HTML com os campos **nome**, **email**, **senha** e um botão de envio. O formulário deve usar `method="POST"` e apontar (action) para um futuro arquivo **registrar.php**.
- Crie um arquivo **form_login.php**. Ele deve conter um formulário com os campos **email**, **senha** e um botão de envio. Deve usar `method="POST"` e apontar (action) para **autenticar.php**.
- Crie um arquivo **painel.php**. Por enquanto, ele terá apenas um título com "Bem-vindo ao Paine!!" e um link para **logout.php**.
- Crie um arquivo **logout.php**. Por enquanto, ele pode ficar vazio.

2: Simulando o Login com Sessões

- Crie o arquivo **autenticar.php**.
 1. No topo do arquivo, inicie a sessão com `session_start();`.
 2. Receba os dados da requisição POST.
 3. Crie uma condição if que verifique se o e-mail é "teste@email.com" e a senha é "123456".
 4. Se a condição for **verdadeira**:
 - Armazene as informações na sessão: `$_SESSION['logado'] = true;` e `$_SESSION['nome_usuario'] = "Usuário Teste";`.
 - Redirecione o usuário para **painel.php** com: `header('Location: painel.php');`.
 5. Se a condição for **falsa**:
 - Redirecione de volta para **form_login.php**.
- Modifique o arquivo **painel.php**:
 1. Inicie(continue) a sessão;
 2. Adicione o "bloco de segurança" no topo do arquivo para verificar se o usuário está autenticado/logado.
 3. Altere o título para exibir o nome do usuário da sessão: "Bem-vindo ao Paine!, [NOME]".
- Implemente o arquivo **logout.php**:

1. Adicione o código completo de logout para limpar e destruir a sessão, e redirecionar para [form_login.php](#).

- **Teste se o login será realizado corretamente comas credenciais hardcoded.**

3: Preparando o Banco de Dados

- Abra o painel de administração do **phpMyAdmin**.
- Crie um novo banco de dados chamado `sistema_login`.
- Selecione o banco `sistema_login` e crie uma nova tabela `usuarios`.
- A tabela deve ter:

```
id INT AUTO_INCREMENT PRIMARY KEY,  
nome VARCHAR(100) NOT NULL,  
email VARCHAR(100) NOT NULL UNIQUE,  
senha_hash VARCHAR(255) NOT NULL
```

- Crie um novo arquivo chamado [conexao.php](#) e insira o código de conexão PDO, configurando-o para o banco `sistema_login`.

Etapa 4: Implementando o Registro no Banco de Dados

- Crie o arquivo [registrar.php](#) (o action do `form_registro.php`).
- Dentro dele, inclua o arquivo de conexão: `require_once 'conexao.php';`.
- Receba os dados de nome, email e senha via `$_POST`.
- **Segurança:** Use a função `password_hash()` para criar um hash seguro da senha recebida.
- **Segurança:** Crie uma instrução SQL INSERT utilizando **PDO Prepared Statements** para inserir os dados (nome, email, `senha_hash`) na tabela `usuarios`.
- Execute a instrução dentro de um bloco `try...catch` para tratar possíveis erros.
- Exiba uma mensagem de sucesso ao usuário e um link para a página de login, ou apenas redirecione para o login.
- **Teste se o cadastro funciona e as informações são armazenadas em seu banco, com a senha em formato hash.**

Etapa 5: Autenticação Real com o Banco de Dados

- Modifique o arquivo [autenticar.php](#).
- Inclua o arquivo de conexão: `require_once 'conexao.php';`.
- Remova a verificação `if` com os dados "hardcoded".
- Receba o email e a senha do formulário `$_POST`.
- Crie uma instrução SQL SELECT com **PDO Prepared Statements** para buscar um usuário na tabela `usuarios` onde o email corresponda ao e-mail fornecido.

- Use `fetch()` para obter os dados do usuário.
- Crie uma nova condição `if` que verifique duas coisas:
 1. Se um usuário foi encontrado (`$usuario` não é falso).
 2. E se a senha digitada confere com o hash do banco, usando `password_verify()`.
- Se a condição for verdadeira, a lógica de iniciar a sessão e redirecionar para o [painel.php](#) (criada na Etapa 2) continua a mesma, atualizando os valores armazenados no array superglobal `$_SESSION[]`.
- Se for falsa, crie uma chave na Sessão para erro `$_SESSION['erro_login']`, com uma mensagem de erro, redirecione o usuário para o login, e modifique o [form_login.php](#) para imprimir essa mensagem de erro na tela.
- **Teste Final:** Tente fazer todo o fluxo de cadastro – login – logout.