**Melhorando a listagem de produtos**

Agora que já conseguimos executar a SQL no banco de dados, vamos melhorar nosso código. Vamos guardar a inteligência de trazer a lista de produtos em uma função específica, que nos devolverá a lista de produtos. Para isso, usaremos arrays. Arrays são um tipo especial no PHP, que nos possibilita guardar "muitos elementos".

Veja o código abaixo. Repare que precisamos ainda receber a conexão como parâmetro da função, já que a variável foi declarada do lado de fora:

<?php

function listaProdutos($conexao) {

$produtos = array();

$resultado = mysqli\_query($conexao, "select \* from produtos");

while($produto = mysqli\_fetch\_assoc($resultado)) {

array\_push($produtos, $produto);

}

return $produtos;

}

?>

Veja que então criamos um array, e a cada iteração do loop, guardamos o novo produto nesse array. No fim, devolvemos todo o array. Isolar em funções é sempre uma boa ideia, já que aumenta o reuso e facilita a manutenção do código.

Mais abaixo, vamos agora usar essa função que acabamos de criar:

$produtos = listaProdutos($conexao);

Com esse array em mãos, podemos navegar na lista de todos os produtos, usando o foreach. Repare na sintaxe, passamos a lista e a variável que conterá cada um dos produtos:

$produtos = listaProdutos($conexao);

foreach($produtos as $produto) {

echo $produto['nome'] . "<br/>";

}

Tudo continua funcionando.

Nosso código já está bem melhor, mas o problema que temos agora é que as consultas SQL estão separadas em diversos arquivos (temos no arquivo de listagem, e no arquivo de adição). Vamos centralizar todas elas em um único arquivo. Isso facilita a manutenção, pois o desenvolvedor saberá que todas as queries de produto estarão no mesmo arquivo. Vamos criar o banco-produto.php:

<?php

function listaProdutos($conexao) {

$produtos = array();

$resultado = mysqli\_query($conexao, "select \* from produtos");

while($produto = mysqli\_fetch\_assoc($resultado)) {

array\_push($produtos, $produto);

}

return $produtos;

}

function insereProduto($conexao, $nome, $preco) {

$query = "insert into produtos (nome, preco) values ('{$nome}', ${preco})";

$resultadoDaInsercao = mysqli\_query($conexao, $query);

return $resultadoDaInsercao;

}

Vamos agora incluir esse arquivo nas outras páginas, e reutilizar essas funções. Toda nova consulta de produtos deve ser criada nesse arquivo. É assim que fazemos em qualquer linguagem de programação, seja PHP, Java ou C#. É sempre importante separar as responsabilidades em diferentes arquivos.

Pronto! Com o código PHP bonito, vamos trabalhar no nosso código HTML, para que a listagem de produtos fique bonita. Vamos usar uma tabela:

<?php

$produtos = listaProdutos($conexao);

?>

<table class="table table-striped table-bordered">

<?php

foreach($produtos as $produto) {

?>

<tr>

<td><?= $produto['nome'] ?></td>

<td><?= $produto['preco'] ?></td>

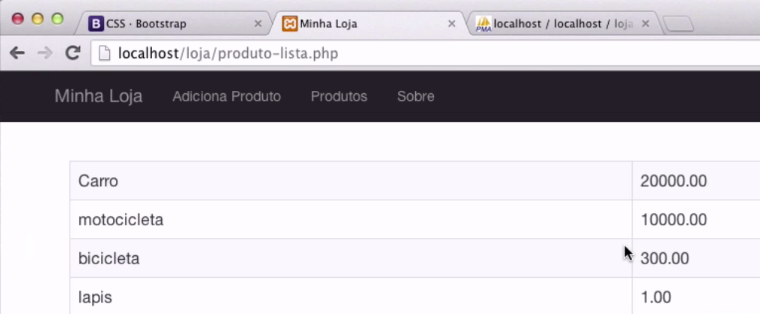
</tr>

<?php

}

?>

</table>



Observe o código direito. Veja que abrimos e fechamos PHP na hora de fazer uso de código HTML.

Alguns programadores não gostam desse abre e fecha chaves, que acontece em código PHP. Existe uma outra maneira de fazer a mesma coisa. Veja o código abaixo com o endforeach. Você pode aplicar a mesma ideia para qualquer outra tag, como if, e etc.

<?php

$produtos = listaProdutos($conexao);

?>

<table class="table table-striped table-bordered">

<?php

foreach($produtos as $produto) :

?>

<tr>

<td><?= $produto['nome'] ?></td>

<td><?= $produto['preco'] ?></td>

</tr>

<?php

endforeach

?>

</table>