

# Sistema de Vendas Online

Na 1º parte deste artigo vou abordar a construção de um carrinho para sistema de vendas online, iremos usar 2 tabelas MySQL. Uma chamada 'carrinho' onde ficaram os itens adicionados e a tabela 'compras' onde ficaram os registros das compras efetuadas no site, salientando que um registro da tabela compras poderá estar ligado a vários registros na tabela carrinho através de uma chave estrangeira. Usaremos o PHP para a lógica do sistema e um pouco de JavaScript para efetuarmos adição de itens, remoção, e recalcule de valores. Este sistema utilizará ainda tabela dos produtos e a de clientes que ficam a critério do usuário sua implementação, neste exemplo estou usando o PHP SESSION para gravar os dados do cliente.

Bom, vamos a prática. Começaremos criando as tabelas no BD:  
Tabela 'carrinho':

```
1 CREATE TABLE IF NOT EXISTS `carrinho` (  
2   `cod_compra` int(8) NOT NULL,  
3   `cod_produto` int(8) NOT NULL,  
4   `unidades_vendidas` int(8) NOT NULL,  
5   `data_insercao` date NOT NULL,  
6   `hora_insercao` time NOT NULL  
7 );
```

Tabela 'compras':

```
1  
2 CREATE TABLE IF NOT EXISTS `compras` (  
3   `cod_compra` int(8) PRIMARY KEY AUTO_INCREMENT NOT NULL,  
4   `cod_cliente` int(8) NOT NULL,  
5   `forma_pgto` varchar(15) NOT NULL,  
6   `parcelamento` int(2) NOT NULL,  
7   `valor_liquido` varchar(10) NOT NULL,  
8   `valor_final` varchar(10) NOT NULL,  
9   `status` varchar(10) NOT NULL  
10 );
```

Adição de itens no carrinho:

Para adicionarmos no carrinho um item usaremos o código do produto e a quantidade itens.

```
1 <input type='hidden' name='cod_produto' id='cod_produto' value='001'>  
2 <label>Quantidade de produtos:</label>  
3 <input type='number' name='quantidade' id='quantidade'>  
4 <input type='button' id='adicionar' value='Adicionar ao Carrinho'>  
5 <input type='button' id='excluir' value='Excluir do Carrinho'>  
6 <input type='button' id='visualizar' value='Visualizar Carrinho'>
```

Código simples com o código do produto oculto, campo para digitação da quantidade de itens e botão que acionará nosso Javascript, temos também o botão de excluir e visualizar que esconderemos via Javascript.

Javascript:

```
1 $(document).ready(function() {  
2   $("#excluir").hide('fast');
```

```

2      $("#visualizar").hide('fast');
3      $("#adicionar").click(function(){
4          var cod_produto = $("#cod_produto").val();
5          var quantidade = $("#quantidade").val();
6          $.getJSON('adicionar_carrinho.php?cod_produto_quantidade='+cod_produto+'
7              if(retorno[0].resposta == 'inseriu'){
8                  alert('O produto foi inserido em seu carrinho.');
```

```

8                  $("#quantidade").hide('fast');
9                  $("#adicionar").hide('fast');
10                 $("#excluir").show('fast');
11                 $("#visualizar").show('fast');
12             }else{
13                 if(retorno[0].resposta == 'já inserido'){
14                     alert('Produto já inserido no carrinho!');
```

```

15                     $("#quantidade").hide('fast');
16                     $("#adicionar").hide('fast');
17                     $("#excluir").show('fast');
18                     $("#visualizar").show('fast');
19                 }else{
20                     alert('O produto não foi inserido!');
```

```

21                 }
22             }
23         });
24     $("#excluir").click(function(){
25         var cod_produto = $("#cod_produto").val();
26         $.getJSON('exclui_carrinho.php?cod_produto='+cod_produto, function(retor
27             if(retorno[0].resposta == 'excluiu'){
28                 alert('O produto foi excluído de seu carrinho.');
```

```

29                 $("#quantidade").show('fast');
30                 $("#adicionar").show('fast');
31                 $("#excluir").hide('fast');
32                 $("#visualizar").hide('fast');
33             }
34         });
35     });
36     $("#visualizar").click(function(){
37         window.open('carrinho.php', '_blank');
```

```

38     });
39 });
40
41
42

```

Explicando o código acima: Começamos escondendo os elementos com id 'excluir' ou 'visualizar' através do método hide(). Depois na linha 4 criamos uma função que será acionada quando o botão adicionar for clicado. Nas linha 5 e 6 gravamos nas variáveis 'cod\_produto' e 'quantidade' a valor dos campos 'cod\_produto' e 'quantidade'. Enviamos via GET o cod\_produto e quantidade separados por um '@' para o arquivo 'adicionar\_carrinho.php' através do \$.getJSON. Criamos a função retorno e verificamos na linha 8 se o JSON retornou dentro da variável resposta o valor 'inseriu' se sim escondemos os elementos com ids 'adicionar' e 'quantidade' e mostramos os elementos com ids 'excluir' e 'visualizar', também exibimos para o usuário uma mensagem de sucesso através do alert(). Se não, verificamos se a resposta não foi 'já inserido', se for exibimos a

mensagem dizendo que o produto já está inserido, esconde os elementos com ids 'quantidade' e 'adicionar' e exibe os elementos com ids 'excluir' e 'visualizar'. Caso o retorno seja diferente de 'inseriu' ou 'jainserido' exibimos a mensagem avisando que não foi possível inserir o produto.

Na linha 27 criamos uma função que será acionada quando clicarmos no elemento com id 'excluir'. Na linha 28 fazemos o mesmo procedimento que na função anterior mas agora so gravamos na variável o código do produto, pois para excluir não precisaremos da quantidade. Enviamos novamente via GET através do \$.getJSON para a página 'exclui\_carrinho.php' a variável cod\_produto. Na linha 30 verificamos se o retorno foi 'excluiu', se sim, mostramos os elementos com ids 'adicionar' e 'quantidade', escondemos os elementos com ids 'excluir' e 'visualizar' e exibimos a mensagem de sucesso. Na linha 39 criamos uma função que será acionada quando o elemento 'visualizar' for clicado. Essa função irá abrir a página do carrinho em uma nova janela através do método window.open().

Agora vamos criar as páginas em php:

adicionar\_carrinho.php

```
1 <?php
2     include('conf.php');
3     session_start();
4
5     $parametro_recebido = $_GET['cod_produto_quantidade'];
6     $parametro_dividido = explode('@', $parametro_recebido);
7     $cod_produto = $parametro_dividido[0];
8     $quantidade = $parametro_dividido[1];
9     $data = date('Y/m/d');
10    $hora = date('H:i:s');
11
12    $consulta_compra_aberta = mysql_query("SELECT cod_compra FROM compras WHERE cod_
13    $resultado_compra_aberta = mysql_num_rows($consulta_compra_aberta);
14    $resposta = "";
15    if($resultado_compra_aberta == 0){
16        $insere_compra = mysql_query("INSERT INTO compras(cod_cliente, status) VALUE
17        $consulta_cod_compra = mysql_query("select cod_compra from compras where sta
18        $row = mysql_fetch_array($consulta_cod_compra);
19        $cod_compra = $row['cod_compra'];
20        $insere_produto = mysql_query("INSERT INTO carrinho(cod_compra, cod_produto,
21        $quantidade', '$data', '$hora')");
22        if($insere_compra && $insere_produto){
23            $resposta .= "inseriu";
24        }
25    }else{
26        $row = mysql_fetch_array($consulta_compra_aberta);
27        $cod_compra = $row['cod_compra'];
28        $consulta_produto_inserido = mysql_query("select cod_produto from carrinho v
29        $produto_inserido = mysql_num_rows($consulta_produto_inserido);
30        if($produto_inserido == 0){
31            $insere_produto = mysql_query("INSERT INTO carrinho(cod_compra, cod_pro
32            $quantidade', '$data', '$hora')");
33            if($insere_produto){
34                $resposta .= "inseriu";
35            }
36        }else{
37            $resposta .= "jainserido";
38        }
39    }
40 }
```

```

35     $respostaJSON = Array(
36         array(
37             "resposta"=>"". $resposta."
38         ));
39     echo json_encode($respostaJSON);
40?>
41
42
43
44
45

```

Na linha 2 incluímos o arquivo 'conf.php' que faz a conexão com o banco de dados. Na linha 5 atribuímos a variável '\$parâmetro\_recebido' o valor recebido via GET. Depois na linha 6 separamos o que está antes e depois do @ gravando o resultado de antes do @ em '\$parametro\_dividido[0]' e o que vier depois em '\$parametro\_dividido[1]'. Atribuímos os resultados as variáveis '\$cod\_produto' e '\$quantidade'. Nas linhas 9 e 10 usamos o método date() para pegar a data e a hora. Na linha 12 efetuamos uma consulta para verificar se há algum registro na tabela compras com o código do cliente e com status igual a 'aberta'. Usamos o método mysql\_num\_rows() para contar o número de registros retornados.

Usamos um if para verificar se o resultado é 0, caso seja, inserimos na tabela compra um registro com o código do cliente e status igual a 'aberta'. Na linha 19 atribuímos a variável '\$cod\_compra' o valor consultado na linha 17, que no nosso caso é o cod\_compra. Depois inserimos na tabela 'carrinho' o(s) item(s) usando cod\_compra que fará a relação com a tabela compra, o código do produto, a quantidade, a data e a hora. Na linha 21 verificamos se as inserções ocorram com sucesso, se sim, atribuímos a variável resposta o valor 'inseriu'.

Na linha 24 temos o else que executará caso o valor retornado da primeira consulta seja diferente de 0. Usamos o mysql\_fetch\_array para pegar os resultados da consulta. Depois na linha 26 gravamos na variável '\$cod\_compra' o valor retornado para o campo 'cod\_compra'. Na linha 27 fizemos uma consulta para verificar se este produto já foi inserido nesta compra, se não, efetuamos a inserção no carrinho da mesma forma como fizemos dentro do if. Verificamos se a inserção funcionou, se sim, gravamos na variável 'resposta' o valor 'inseriu'. Se sim gravamos na variável 'resposta' o valor 'jainseriu'. Na linha 39 criamos a variável '\$respostaJSON' a qual usaremos para enviar a resposta no formato de JSON. Na linha 44 usamos o método json\_encode() para retornar a resposta.

exclui\_carrinho.php

```

1     <?php
2         include('conf.php');
3         session_start();
4
5         $cod_produto = $_GET['cod_produto'];
6         $consulta_cod_compra = mysql_query("SELECT cod_compra FROM compras WHERE cod_
7         $row = mysql_fetch_array($consulta_cod_compra);
8         $cod_compra = $row['cod_compra'];
9         $deleta_registro = mysql_query("DELETE FROM carrinho WHERE cod_compra=".$cod_
10
11         $resposta = '';
12         if($deleta_registro){
13             $resposta .= 'excluiu';
14
15         }
16
17         $respostaJSON = Array(

```

```

15         array(
16             "resposta"=>"". $resposta."
17         ));
18     echo json_encode($respostaJSON);
19     ?>
20
21
22

```

Na linha 5 gravamos na variável '\$cod\_produto' o valor recebido via GET. Na linha 6 efetuamos uma consulta para pegar o cod\_compra. Gravamos o resultado na variável '\$cod\_compra'. Na linha 9 deletamos o registro da tabela carrinho que possuir o código da venda igual ao consultado e código do produto igual ao recebido via GET. Na linha 12 verificamos se deletou, se sim, atribuímos a variável \$resposta o valor 'excluiu'. Geramos a resposta no formato de JSON e retornamos a resposta através do método json\_encode.

carrinho.php

```

1  <table>
2      <tr>
3          <td>Produto</td>
4          <td>Preço Un.</td>
5          <td>Quantidade</td>
6          <td>Preço Total</td>
7          <td>Excluir</td>
8          <td>Atualizar</td>
9      </tr>
10     <?php
11         session_start();
12         $consulta_compra = mysql_query("SELECT cod_compra FROM compras WHERE cod_cl
13         $resultatdo_consulta_compra = mysql_num_rows($consulta_compra);
14         if($resultatdo_consulta_compra == 0){
15             echo "<tr><td colspan='4'>O carrinho encontra-se vazio! </td></tr>";
16         }else{
17             $row = mysql_fetch_array($consulta_compra);
18             $cod_compra = $row['cod_compra'];
19             $preenche_carrinho = mysql_query("SELECT carrinho.cod_produto, carrinho
20             carrinho.cod_compra = ".$cod_compra." AND carrinho.cod_produto = produtos.id");
21             $resultado_consulta_carrinho = mysql_num_rows($preenche_carrinho);
22             if($resultado_consulta_carrinho == 0){
23                 echo "<tr><td colspan='4'>O carrinho encontra-se vazio! </td></tr>";
24             }else{
25                 $total = 0;
26                 $i = 1;
27                 while($dados = mysql_fetch_array($preenche_carrinho)){
28                     $preco_total = $dados['preco']*$dados['unidades_vendidas'];
29                     echo "<tr><td colspan='2'>". $dados['nome']. "</td><td>". $dados['preco
30                     value='". $dados['cod_produto']. "'><input type='number' id='quantidade".$i.'" name='q
31                     value='". $dados['unidades_vendidas']. "'></td><td>". $preco_total. "</td><td><input ty
32                     onclick='atualizar(\"linha".$i.\"\", \"quantidade".$i.\"\")' value='Atualizar'></td>
33                     $total += $preco_total;
34                     $i++;
35                 }
36                 echo "<tr><td colspan='4'>Total: ".$total." </td></tr>";
37             }
38         }
39     ?>
40 </table>
41

```

35  
36

Na nossa página de visualização do carrinho usaremos uma tabela, então criamos o seu cabeçalho em HTML. Na linha 12 efetuamos uma consulta para verificar se algum registro com o status 'aberta' para este cliente. Na linha 13 usamos o método `mysql_num_rows()` para contar o número de registros retornados. Verificamos se é igual a 0, se sim, exibimos 'O seu carrinho encontra-se vazio'. Caso contrario na linha 17 usamos o método `mysql_fetch_array()` para pegar o resultado. Na linha 18 gravamos na variável '\$cod\_compra' o valor retorno do campo `cod_compra`. Efetuamos uma consulta em duas tabelas na linha 19, onde selecionamos na tabela 'carrinho' o `cod_produto` e `unidades_vendidas` e na tabela 'produtos' o nome e o preço, quando o campo `cod_compra` for igual ao valor armazenado na variável '\$cod\_compra' e quando o `cod_produto` da tabela carrinho for igual ao id da tabela produtos. Contamos os resultados desta consulta e verificamos se é igual a 0, caso seja, exibimos 'O seu carrinho encontra-se vazio'.

Se não, criamos as variáveis 'total' que armazenará o valor total de todos os itens e a variável 'i' que contara as linhas para que depois possamos atualizar o carrinho. Na linha 26 criamos um while que executara enquanto tivermos dados vindos da nossa consulta ao carrinho. Na linha 27 calculamos o preço total de cada item, sendo o preço da unidades \* quantidade comprada. Na linha 28 exibimos os resultados no formato de linha dividido pelas colunas, tag span com o `id='linha'.$i.'` é importante, pois através dele que localizaremos o `cod_produto` para que possamos alterar a quantidade ou excluir o produto. Usamos o evento `onclick='atualizar(\"linha\".$i.\"')'` ou `onclick='excluir(\"linha\".$i.\"')'` para enviar a função javascript o id, que será o mesmo que o gerado no id do span. E usamos `$dados['nome_do_campo']` para pegar os dados resultados na consulta SQL. Na linha 28 usamos o '+' para somar na variável '\$total' o preço total de um produto. Na linha 32 exibimos o valor total de todos os itens do carrinho.

Função Javascript da página 'carrinho.php'.

```
1
2
3     function atualizar(linha, quantidade){
4         var codigo = document.getElementById(linha).value;
5         var quantidade = document.getElementById(linha).value;
6         $.getJSON('atualizar_carrinho.php?cod_produto_quantidade='+cod_produto+'@'+quant
7             if(retorno[0].resposta == 'atualizou'){
8                 alert('Carrinho atualizado com sucesso!');
9                 document.location.reload(true);
10            }else{
11                alert('Não foi possível atualizar o carrinho!');
12            }
13    });
14    }
15    function excluir(linha){
16        var codigo = document.getElementById(linha).cod_produto.value;
17        $.getJSON('exclui_carrinho.php?cod_produto='+cod_produto, function(retorno){
18            if(retorno[0].resposta == 'excluiu'){
19                document.location.reload(true);
20            }
21        });
22    }
23    }
24    }
25    }
26    }
27    }
28    }
29    }
30    }
31    }
32    }
33    }
34    }
35    }
36    }
```

Neste código temos as funções 'excluir' e 'atualizar' que serão acionadas pelo método `onclick()`. Na linha 2 e 3 gravamos respectivamente dentro das variáveis código e quantidade o valor que estiver dentro dos elementos com id's `cod_produto` e `quantidade` que por sua vez estiverem dentro do elemento com id recebido do método `onclick`.

Usamos o mesmo procedimento das outras funções para enviar os dados. Na linha 7 usamos o método `document.location.reload()` para recarregar a página. A função `excluir` segue a mesma lógica da função `atualizar`.

`atualizar_carrinho.php`

```
1
2
3  <?php
4      include('conf.php');
5      session_start();
6
7      $parametro_recebido = $_GET['cod_produto_quantidade'];
8      $parametro_dividido = explode('@', $parametro_recebido);
9      $cod_produto = $parametro_dividido[0];
10     $quantidade = $parametro_dividido[1];
11
12     $consulta_cod_compra = mysql_query("SELECT cod_compra FROM compras WHERE cod_
13     $row = mysql_fetch_array($consulta_cod_compra);
14     $cod_compra = $row['cod_compra'];
15     $atualiza_registro = mysql_query("UPDATE carrinho SET unidades_vendas=". $qua
16
17     $resposta = '';
18     if($atualiza_registro){
19         $resposta .= 'atualizou';
20     }
21
22     $respostaJSON = Array(
23         array(
24             "resposta"=>"".$resposta.""
25         ));
26
27     echo json_encode($respostaJSON);
28
29 ?>
```

Esse código funciona como o de excluir itens, porém atualiza a quantidade de produtos através do comando `UPDATE` na linha 13, se atualizar, devolve a resposta `atualizou`.

.