

## Aula 14: PHP com MySQL

Professor: Jorge H. B. Casagrande

[casagrande@ifsc.edu.br](mailto:casagrande@ifsc.edu.br)

Notas de aula adaptada da original do prof. Emerson R. de Mello

## 1 Introdução

O PHP apresenta um conjunto de funções para trabalhar com diversos bancos de dados, por exemplo, MySQL, PostgreSQL, Firebird, entre outros. Os bancos de dados consistem em uma forma organizada para o armazenamento de dados.

Em um banco de dados as informações são organizadas em tabelas, sendo que cada linha na tabela representa um **registro** e as colunas da tabela representam os campos presentes em cada registro. Por exemplo, se desejarmos criar uma tabela para armazenar os nomes de usuários e as respectivas senhas, então as colunas desta tabela seriam: **usuário** e **senha**. Cada usuário do nosso sistema será representado por uma linha na tabela. Veja o exemplo através da tabela 1.

usuario	senha
joão	123456
pedro	654321
mariana	abc123

Tabela 1: Representação da tabela de usuários

## 2 MySQL

O MySQL é um sistema de gerenciamento de banco de dados e também por objetivo permite gerenciar o acesso, manipulação e organização de dados. Com o MySQL é possível criar diversas bases de dados, chamadas de esquemas, e indicar os direitos de acesso de cada usuário sobre cada base de dados. Segue abaixo os passos para a instalação e criação de uma base de dados e a associação de um usuário e senha responsável por administrar tal base.

1. Para a instalação do MySQL execute o comando:

```
1 sudo apt-get install mysql-server mysql-client
```

2. Durante o processo da instalação será solicitada uma senha do administrador (root) do MySQL, use por conveniência “aluno”;
3. Para associar este banco de dados ao PHP já instalado em aula anterior, execute o comando:

```
1 sudo apt-get install php5-mysql
```

- Por padrão o MySQL no Ubuntu só aceitará conexões oriundas da própria máquina;
- Se desejar conectar remotamente ao MySQL deve-se colocar o valor 0.0.0.0 para o campo `bind-address` no arquivo `/etc/mysql/my.cnf` e reiniciar o serviço com:

```
1 cd /etc/init.d e em seguida ./mysql restart ou ./mysqld restart
```

4. Entre no terminal interativo do MySQL:

```
1 mysql -u root -p
```

5. Crie a base de dados aula:

```
1 create database aula;
```

6. Garanta todos os direitos de acesso ao usuário 'aluno' com senha '123':

```
1 GRANT ALL PRIVILEGES on aula.* to 'aluno'@'%' identified by '123';
```

7. Saia do modo interativo e entre novamente agora com o usuário **aluno** e selecionando a base de dados **aula**:

```
1 quit
2 mysql aula -u aluno -p
```

### 3 Linguagem para interação com o banco de dados

A interação com o banco de dados se dá através de uma linguagem específica, chamada *Structured Query Language* (SQL). O SQL é formado por um conjunto padronizado de instruções, das quais podemos destacar seis:

#### Atenção:

- i) Para aplicar qualquer comando em uma base de dados é necessário que ela esteja em edição (use: “use [nomedabasededados]” dentro do ambiente MySQL;
- ii) dentro do ambiente, para visualizar detalhes da base de dados, voce também pode usar a qualquer momento a instrução “show [opção]” onde opção= “databases”, “tables”, entre outros (use “?”) para descobrir mais instruções.

- CREATE – Pode ser usado para criar tabelas onde ficam armazenados os dados em si;

```
1 CREATE TABLE contatos (codigo INT, nome VARCHAR(40), email VARCHAR(50));
```

ou:

Criando uma tabela de forma que o campo código seja chave primária e tenha seus valores incrementados automaticamente a cada nova inclusão de registro (atenção: se voce for aplicar o comando abaixo, é necessário apagar (ver comando DROP abaixo) a tabela contatos criada acima):

```
1 CREATE TABLE contatos (codigo INT NOT NULL AUTO_INCREMENT, nome VARCHAR(40), email
  VARCHAR(50), PRIMARY KEY(codigo));
```

- DROP – Pode ser usado para apagar tabelas no banco;

```
1 DROP TABLE contatos;
```

- INSERT – Para inserir uma nova linha em uma tabela;

```
1 INSERT INTO contatos (codigo,nome,email) VALUES (10,'Joao','j@e.com');
```

- DELETE – Pode ser usado para excluir linhas de uma tabela;

```
1 DELETE FROM contatos WHERE codigo=10;
```

- SELECT – Para obter os dados armazenados em uma tabela;

```
1 SELECT * FROM contatos;  
2 SELECT nome,email FROM contatos WHERE codigo=10;  
3 SELECT * FROM contatos WHERE codigo = 10 OR codigo = 5;  
4 SELECT * FROM contatos WHERE email LIKE '%@ifsc.edu.br';
```

- UPDATE – Para atualizar os registros em um tabela;

```
1 UPDATE contatos SET email='joao@email.com' WHERE codigo=10;
```

## 4 Executando instruções SQL em PHP

```
1 <?
2 //conectando no servidor
3 $conexao = mysql_connect("localhost","usuario","senha");
4
5 //selecao da base de dados
6 $base_ok = mysql_select_db("basedoaluno",$conexao);
7
8 //inserindo um registro
9 $inserir = "INSERT INTO contatos (nome,email) VALUES ('emerson','mello@sj')";
10 mysql_query($inserir,$conexao);
11
12 //fechando a conexao
13 mysql_close($conexao);
14
15 ?>
```

Figura 1: Inserindo dados em um banco

```
1 <html>
2 <head>
3   <title>Listando os contatos</title>
4 </head>
5 <body>
6 <center><h1>Contatos</h1></center><p>
7 <?
8   //conectando no servidor
9   $conexao = mysql_connect("localhost","usuario","senha");
10
11 //selecao da base de dados
12 $base_ok = mysql_select_db("basedoaluno",$conexao);
13
14 //obtendo os registro da tabela contatos
15 $consulta = "SELECT * from contatos";
16 $resultado = mysql_query($consulta,$conexao);
17
18 //percorrendo a linhas obtidas pela consulta
19 while($linha = mysql_fetch_array($resultado)){
20   echo "codigo: " . $linha['codigo'];
21   echo "nome: " . $linha['nome'];
22   echo "email: " . $linha['email'];
23   echo "<BR><HR><BR>";
24 }
25
26 //liberando a consulta
27 mysql_free_result($resultado);
28
29 //fechando a conexao
30 mysql_close($conexao);
31
32 ?>
33 </body>
34 </html>
```

Figura 2: Obtendo dados de um banco