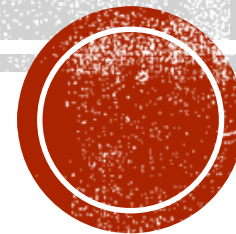


# REVISÃO E AULA CONEXÃO BANCO

Prof: Valdeir



# PHP E MYSQLI

## Classe MySQLi

- Classe que representa uma conexão entre o PHP e um banco de dados MySQL.
  - Atributos
    - `affected_rows` — Número de linhas afetadas pela operação MySQL anterior
    - `error` — String que descreve o ultimo erro
    - `server_info` — Versão do servidor MySQL
    - `server_version` — Versão do servidor MySQL como um integer



# PHP E MYSQLI

## Classe MySQLi

- Métodos

`__construct()`

- Abre uma conexão com servidor MySQL

`close()`

- Fecha uma conexão aberta anteriormente

`select_db()`

- Seleciona uma base de dados(database)

`query()`

- Realiza uma consulta na base de dados

...



# PHP E MYSQLI

## Classe MySQLi\_Result

- Representa o conjunto de resultados da consulta feita à base de dados(database)
  - Atributos
    - num\_rows — Número de linhas retornado da consulta
    - field\_count — Número de colunas retornada da consulta
    - lengths — Tamanho das colunas da atual linha retornado da consulta
    - current\_field — Linha atual retornado do método fetch\_field()





# PHP E MYSQLI

## Classe MySQLi\_Result

- Métodos

`mysqli_fetch_assoc()`

- Obtem uma linha do conjunto de resultados como uma matriz associativa
- Percorre resultados de uma pesquisa

`mysqli_fetch_row()`

- Obtém uma linha do resultado como uma matriz numerada

`fetch_object()`

- Retorna a linha atual do conjunto de resultados como um objeto

`free()`

- Libera memória do conjunto de resultados chamado



# PHP E MYSQLI

## Classe MySQLi

- Conexão com o banco de dado

```
__construct(string $host,  
            string $username, string $passwd,  
            string $dbname, string $port,  
            string $socket);
```

– Exemplo

```
$db = new mysqli($host,$username,$passwd);
```

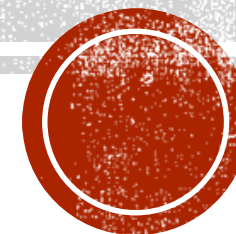
# PHP E MYSQLI

## Conexão com o banco de dado

```
<?php
    $host = "localhost";
    $username = "root";    $passwd = "";

    $db = new mysqli($host,$username,$passwd);

    if ($db->connect_error){// Verifica a conexão
        die('Erro('.$db->connect_errno.')'
            . $db->connect_error);
    }else{
        echo "Conexão aberta";
    }
?>
```



Prof: Valdeir

# PHP E MYSQLI

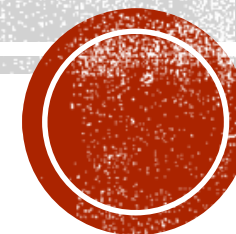
## Classe MySQLi

- Fechar uma conexão aberta anteriormente com o banco de dados
  - Método

```
bool close ( void );
```

- Exemplo

```
$db->close();
```





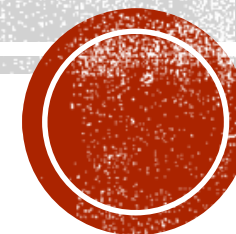
# PHP E MYSQLI

## Conexão com o banco de dado

```
<?php
$host = "localhost";
$username = "root";    $passwd = "";

$db = new mysqli($host,$username,$passwd);

if ($db->connect_error){// Verifica a conexão
    die('Erro('.$db->connect_errno.')'.$db->connect_error);
}else{
    echo "Conexão aberta";
    $db->close();
}
?>
```



# PHP E MYSQLI

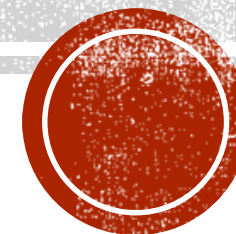
## Classe MySQLi

- Selecionar uma base de dados

```
bool select_db(string $dbname);
```

- Exemplo

```
$db->select_db( "books" );
```



# PHP E MYSQLI

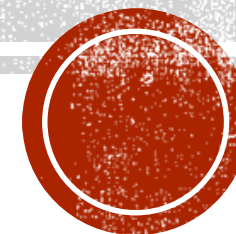
## Conexão com o banco de dado

```
<?php
$host = "localhost";
$username = "root";    $passwd = "";

$db = new mysqli($host,$username,$passwd);

if ($db->connect_error){// Verifica a conexão
    die('Erro('.$db->connect_errno.')'.$db->connect_error);
}else{
    $db->select_db("books");

    $db->close();
}
?>
```



# PHP E MYSQLI

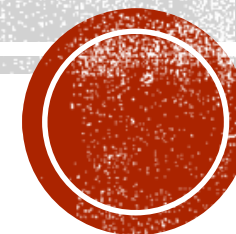
## Classe MySQLi

- Realizar uma consulta na base de dados

```
mixed query(string $query)
```

– Exemplo

```
$db->query( "SELECT author FROM books" );
```





# PHP E MYSQLI

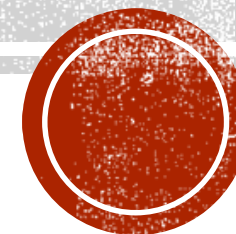
## Criação da base de dados

- SQL

```
CREATE DATABASE dbname;
```

- PHP

```
$sql = "CREATE DATABASE ".$dbname;  
$db->query($sql);
```



# PHP E MYSQLI

## Criação da base de dados

```
<?php
$db = new mysqli($host,$username,$passwd);

if ($db->connect_error){// Verifica a conexão
    die('Erro('.$db->connect_errno.')'.$db->connect_error);
}

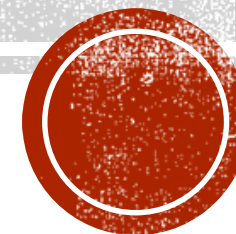
$dbname = "books";

$sql = "CREATE DATABASE ".$dbname;

if($db->query($sql)){
    echo "Base de dados criada com sucesso";
}

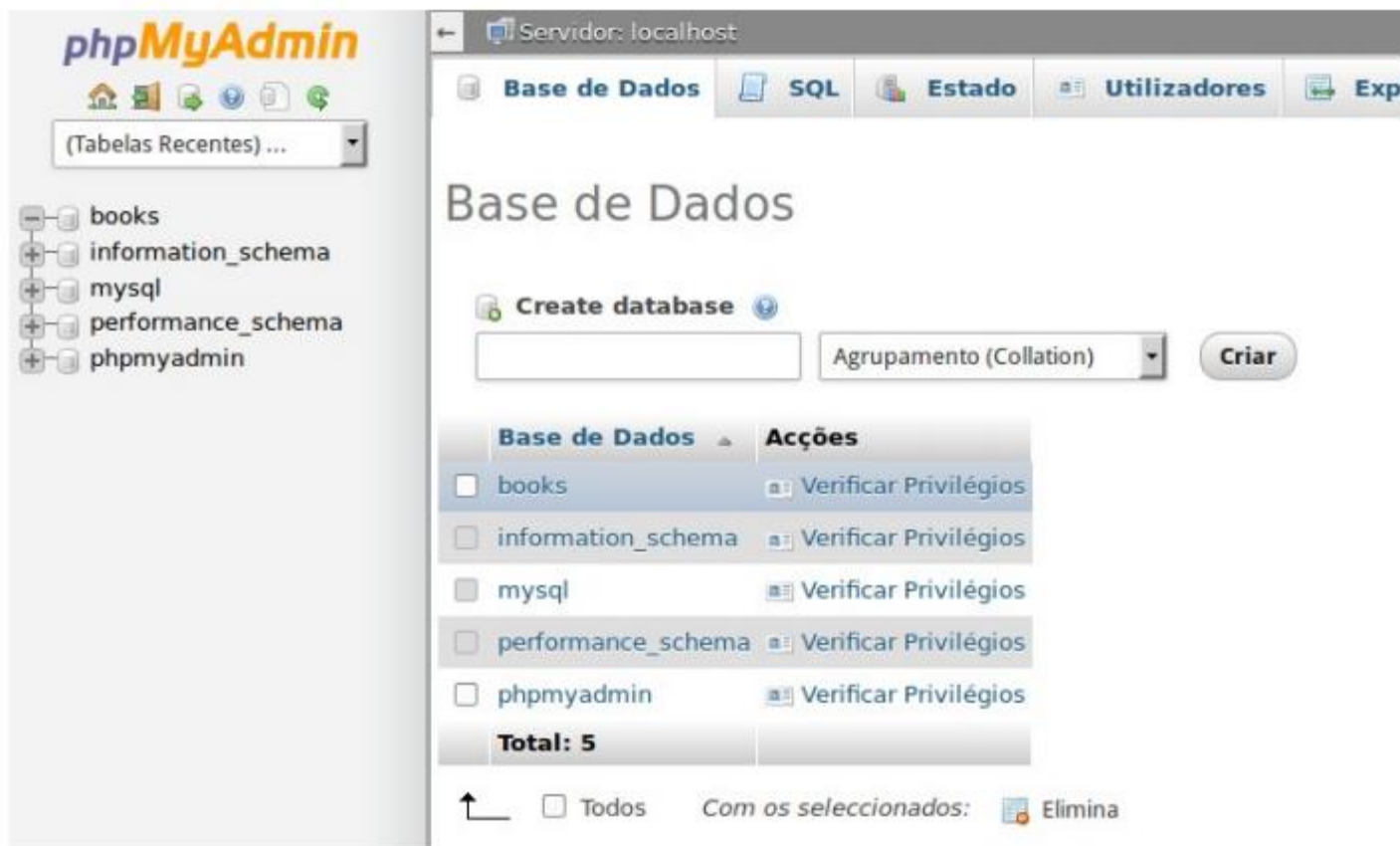
$db->close();

?>
```



# PHP E MYSQLI

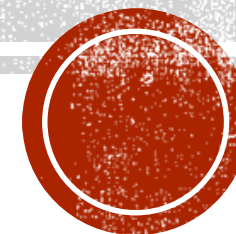
## Criação da base de dados



The screenshot shows the phpMyAdmin interface for a local server. The left sidebar lists databases: books, information\_schema, mysql, performance\_schema, and phpmyadmin. The main panel is titled 'Base de Dados' and includes tabs for 'Base de Dados', 'SQL', 'Estado', 'Utilizadores', and 'Exp'. Under 'Base de Dados', there is a 'Create database' section with a text input field, a 'Agrupamento (Collation)' dropdown, and a 'Criar' button. Below this is a table listing the existing databases and their actions.

Base de Dados	Acções
<input type="checkbox"/> books	<a href="#">Verificar Privilégios</a>
<input type="checkbox"/> information_schema	<a href="#">Verificar Privilégios</a>
<input type="checkbox"/> mysql	<a href="#">Verificar Privilégios</a>
<input type="checkbox"/> performance_schema	<a href="#">Verificar Privilégios</a>
<input type="checkbox"/> phpmyadmin	<a href="#">Verificar Privilégios</a>
<b>Total: 5</b>	

At the bottom, there is a navigation bar with an upward arrow, a 'Todos' checkbox, the text 'Com os seleccionados:', and an 'Elimina' button with a trash icon.



# PHP E MYSQLI

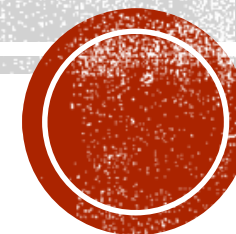
## Deletar uma base de dados

- SQL

```
DROP DATABASE dbname;
```

- PHP

```
$sql = "DROP DATABASE ".$dbname;  
$db->query($sql);
```





# PHP E MYSQLI

## Deletar Base de Dados

```
$dbname = "books";

$sql = "DROP DATABASE ".$dbname;

if($db->query($sql)){
    echo "Base de dados deletada com sucesso";
}
```

