

Desenvolvimento de Sistemas para Web

Prof. Me. Anderson Burnes

PDO - PHP Data Objects

- Driver de conexão com banco de dados
- PHP > 5.1
- Pode se conectar a diversos bancos diferentes, além do MySQL e do MariaDB
 - SQL Server
 - PostgreSQL
 - Oracle
 - SQLite
 - Firebird

PDO - PHP Data Objects

- Exemplo de conexão:

```
1 <?php
2 $pdo = new PDO("mysql:host=servidor;dbname=banco_de_dados", "usuario", "senha");
3
```

- Parâmetros necessários:
 - Host
 - Usuário
 - Senha
 - Banco de Dados
 - Driver

PDO - PHP Data Objects

- Parâmetro opcional: charset

```
1 <?php
2 $pdo = new PDO("mysql:host=servidor;dbname=banco_de_dados;charset=utf8", "usuario", "senha");
3
```

PDO - PHP Data Objects

- Para utilizar o PDO com outro banco é necessário habilitar a extensão do banco no php.ini

```
1 ;extension=pdo_firebird
2 ;extension=pdo_mysql
3 ;extension=pdo_oci
4 ;extension=pdo_odbc
5 ;extension=pdo_pgsql
6 ;extension=pdo_sqlite
7
```

PDO – PHP Data Objects

- Vantagens:
 - Feito especificamente para trabalhar com procedimentos relacionados a banco de dados
 - Fazer a aplicação sem se preocupar com qual banco será utilizado
 - Compatibilidade com diferentes bancos
 - Segurança
 - Utilizado na maioria dos Frameworks atuais

PDO – PHP Data Objects

- Funcionalidades
 - Tratamento de Transações (beginTransaction, commit, rollback)
 - Queries parametrizadas (bindParam, bindValue)

PDO - PHP Data Objects

- Exemplo (named placeholders):

```
1 <?php
2
3 $id = 1;
4
5 $sql = "select id, titulo from noticia where texto where id = ".(int)$id;
6 $consulta = mysqli_query($con, $sql);
7 $dados = mysqli_fetch_array($consulta);
8
9 $sql = "select id, titulo from noticia where id = :id";
10 $consulta = $con->prepare($sql);
11 $consulta->bindParam(":id", $id);
12 $consulta->execute();
13 $dados = $consulta->fetch(PDO::FETCH_OBJ);
14
```


PDO – PHP Data Objects

- **`$consulta->prepare()`**
 - Prepara um sql para execução (prepared statement)
- **`$consulta->execute()`**
 - Executa um prepared statement
- **`$consulta->bindParam`**
 - Fornece um parâmetro para a consulta
- **`$consulta->fetch`**
 - Retorna os resultados (objs ou arrays)

PDO - PHP Data Objects

- Exemplo (data_type):

```
1 <?php
2
3 $id = 1;
4 $titulo = "Olá Mundo";
5
6 $sql = "select id, titulo from noticia where id = :id and titulo = :titulo";
7 $consulta = $con->prepare($sql);
8 $consulta->bindParam(":id", $id, PDO::PARAM_INT);
9 $consulta->bindParam(":titulo", $titulo, PDO::PARAM_STR, 30);
10 $consulta->execute();
11 $dados = $consulta->fetch(PDO::FETCH_OBJ);
12
```

PDO – PHP Data Objects

- Data types:
 - PARAM_BOOL
 - PARAM_NULL
 - PARAM_INT
 - PARAM_STR

PDO - PHP Data Objects

- Exemplo (question marker):

```
1  <?php
2
3  $id = 1;
4  $titulo = "Olá Mundo";
5
6  $sql = "select id, titulo from noticia where id = ? and titulo = ?";
7  $consulta = $con->prepare($sql);
8  $consulta->bindParam(1, $id);
9  $consulta->bindParam(2, $titulo);
10 $consulta->execute();
11 $dados = $consulta->fetch(PDO::FETCH_OBJ);
12
```

PDO - PHP Data Objects

- Exemplo (named placeholders):

```
3 $id = 1;
4 $titulo = "Olá Mundo";
5
6 $consulta = $con->prepare("select id, titulo from noticia where id = ? and titulo = ?");
7 $consulta->bindParam(1, $id);
8 $consulta->bindParam(2, $titulo);
9 $consulta->execute();
10 $dados = $consulta->fetch(PDO::FETCH_OBJ);
```

PDO - PHP Data Objects

- bindParam e bindValue
 - Parecidos... mas diferentes.

```
19 $cidade = "Alegre";
20
21 $sql = "select * from cidade where cidade = :cidade ";
22 $consulta = $pdo->prepare($sql);
23 $consulta->bindParam(':cidade', $cidade);
24 $cidade = "Anchieta";
25 $consulta->execute();
26 $dados = $consulta->fetch(PDO::FETCH_OBJ);
27
28 echo "<p>Cidade: {$dados->cidade}</p>";
29
30 $cidade = "Alegre";|
31
32 $sql = "select * from cidade where cidade = :cidade ";
33 $consulta = $pdo->prepare($sql);
34 $consulta->bindValue(':cidade', $cidade);
35 $cidade = "Anchieta";
36 $consulta->execute();
37 $dados = $consulta->fetch(PDO::FETCH_OBJ);
38
39 echo "<p>Cidade: {$dados->cidade}</p>";
```

PDO - PHP Data Objects

- bindParam e bindValue
 - Parecidos... mas diferentes.

```
19 $cidade = "Alegre";
20
21 $sql = "select * from cidade where cidade = :cidade ";
22 $consulta = $pdo->prepare($sql);
23 $consulta->bindParam(':cidade', $cidade);
24 $cidade = "Anchieta";
25 $consulta->execute();
26 $dados = $consulta->fetch(PDO::FETCH_OBJ);
27
28 echo "<p>Cidade: {$dados->cidade}</p>";
29
30 $cidade = "Alegre";|
31
32 $sql = "select * from cidade where cidade = :cidade ";
33 $consulta = $pdo->prepare($sql);
34 $consulta->bindValue(':cidade', $cidade);
35 $cidade = "Anchieta";
36 $consulta->execute();
37 $dados = $consulta->fetch(PDO::FETCH_OBJ);
38
39 echo "<p>Cidade: {$dados->cidade}</p>";
```

Cidade: Anchieta

Cidade: Alegre

PDO - PHP Data Objects

- Mostrando os dados retornados:

```
1 <?php
2
3 $id = 1;
4 $titulo = "Olá Mundo";
5
6 $consulta = $con->prepare("select id, titulo from noticia where id = ? and titulo = ?");
7 $consulta->bindParam(1, $id);
8 $consulta->bindParam(2, $titulo);
9 $consulta->execute();
10 $dados = $consulta->fetch(PDO::FETCH_OBJ);
11
12 ?>
13
14 <h1><?=$dados->titulo;?></h1>
15
```


PDO - PHP Data Objects

- Resultados
 - fetchAll
 - `$consulta->fetchAll()`
 - FETCH_OBJ (objetos)
 - `$consulta->fetch(PDO::FETCH_OBJ);`
 - FETCH_ASSOC (array)
 - `$consulta->fetch(PDO::FETCH_ASSOC);`

PDO - PHP Data Objects

- Outras funções importantes:
 - Inicia uma transação:
 - `$pdo->beginTransaction()`
 - Confirma transação
 - `$pdo->commit()`
 - Volta transação
 - `$pdo->rollBack()`

PDO - PHP Data Objects

- Outras funções importantes:
 - Retorna último ID inserido:
 - `$pdo->lastInsertId()`
 - Número de linhas de resultados
 - `$consulta->rowCount()`

OO e o PHP

```
1 <?php
2     class Exemplo {
3
4         public $valor;
5
6         public function getValor($valor) {
7             $this->valor = $valor;
8         }
9
10        public function retornaValor() {
11            return "<p>Valor retornado: {$this->valor}</p>";
12        }
13
14    }
15
16    $exemplo = new Exemplo();
17    $exemplo->getValor(100);
18    echo $exemplo->retornaValor();
19
```

OO e o PHP

```
1  <?php
2      class Exemplo {
3
4          public static function retornaValor($valor) {
5              return "<p>Valor retornado: {$valor}</p>";
6          }
7
8      }
9
10     echo Exemplo::retornaValor(100);
11
```

PHP 8

– Union Types

```
1 <?php
2
3     function recebeValor( int|float $valor, int|float $valor2 ):float {
4         return $valor * $valor2;
5     }
6
7     echo recebeValor(1.2,2);
8
```

PHP 8

- Throw Exception

```
1 <?php
2
3     $id = $_GET['id'] ?? throw new Exception("Erro ao recuperar ID");
4
```

- Fatal error: Uncaught Exception: **Erro ao recuperar ID** in /Applications/MAMP/htdocs/2021/3 si/php8/ex1.php:3 Stack trace: #0 {main} thrown in /Applications/MAMP/htdocs/2021/3 si/php8/ex1.php on line 3

PHP 8

– Mais rápido

- JIT (Just in Time) compilador que passa instruções direto da CPU (opcode – código existente em uma CPU para que ela realize uma tarefa)
- Só enviará para o MV da Zend códigos necessários
- Ganho de performance (4x superior ao PHP7)
- https://www.youtube.com/watch?v=dWH65pmnsrl&feature=emb_title&ab_channel=ZeevSuraski

PHP 8

– Engine Warnings

```
1  <?php
2
3  echo $numero;
```

- Warning: Undefined variable \$numero in /Applications/MAMP/htdocs/2021/3 si/php8/ex1.php on line 3
- https://wiki.php.net/rfc/engine_warnings

PHP 8

– Engine Warnings

```
1  <?php
2
3  echo $numero;
```

- Warning: Undefined variable \$numero in /Applications/MAMP/htdocs/2021/3 si/php8/ex1.php on line 3

PHP 8

– str_contains

```
1  <?php
2
3      $texto = "O melhor curso de Sistemas para Internet";
4
5      if ( str_contains ( $texto, "Sistemas" ) ) {
6          echo "Sistemas encontrado";
7      } else {
8          echo "Não encontrado";
9      }
10
```

PHP 8

– str_starts_with e ends_with

```
1  <?php
2
3      $texto = "O melhor curso de Sistemas para Internet";
4
5      if ( str_starts_with ( $texto, "O" ) ) {
6          echo "Verdadeiro";
7      } else {
8          echo "Falso";
9      }
10
11     echo "<br>";
12
13     if ( str_ends_with ( $texto, "Internet" ) ) {
14         echo "Verdadeiro";
15     } else {
16         echo "Falso";
17     }
18
```

PHP > 5

– str_replace

```
<?php  
  
$valor = "1.000,00";  
  
$valor = str_replace(".", ",", $valor); // 1000,00
```

PHP > 5

– addslashes

```
<?php
$texto = "Dois colehos com uma caixa d'agua";
echo addslashes($texto); //Dois colehos com uma caixa d\'agua
```

```
<?php
$texto = 'E ele disse "Oi Meninas" ';
echo addslashes($texto); //E ele disse \"Oi Meninas\"
```

PHP > 5

– htmlspecialchars

```
<?php
    $texto = '<p>Olá Mundo!!!!<br>Quebra de Linha</p>';
    echo htmlspecialchars($texto); |
```

```
<p>Olá Mundo!!!!<br>Quebra de Linha</p>
```

PHP > 5

– explode

```
<?php  
  
$data = "02/10/2021";  
  
$array = explode("/", $data);  
  
print_r ( $array );|
```

```
Array ( [0] => 02 [1] => 10 [2] => 2021 )
```


PHP > 5

– explode

```
<?php
```

```
$data = "02/10/2021";
```

```
$array = explode("/", $data);
```

```
echo $array[2] . "-" . $array[1] . "-" . $array[0]; //2021-1-02
```

PHP > 5

– implode

```
<?php  
  
$array = array("laranja","limão","jaca","banana");  
  
echo implode(", ", $array); //laranja, limão, jaca, banana
```

PHP > 5

– array_search

```
<?php
$array = array("laranja","limão","jaca","banana");
echo array_search("jaca", $array); //2
```

– asort

```
<?php
$array = array("laranja","limão","jaca","banana");
asort($array);
print_r($array); //Array ( [3] => banana [2] => jaca [0] => laranja [1] => limão )
```

PHP > 5

– arsort

```
<?php
$array = array("laranja","limão","jaca","banana");
arsort($array);
print_r($array); // Array ( [1] => limão [0] => laranja [2] => jaca [3] => banana )
```

– ksort

```
<?php
$array = array("laranja","limão","jaca","banana");
ksort($array);
print_r($array); // Array ( [0] => laranja [1] => limão [2] => jaca [3] => banana )
```

PHP > 5

– array_key_exists

```
<?php  
  
$array = array("nome"=>"Jorge da Silva","idade"=>16,"cidade"=>"Xambrê");  
  
if (array_key_exists('nome', $array)) {  
    echo "Encontrado";  
}
```