Minhas colas de PHP:

**$x = 0;**

**While( $x =< 10){**

**x++;**

**echo “O numero X vale “.$x.”<br/>”**

**}**

Esse código mostra a função do While num código PHP, nesse caso ter um contador de 1 até 10, agora usando For:

For($x = 0; $x=10; $x++){

Echo “o numero X vale “.$x.”<br/>”

}

Basicamente a mesma função do While, porém a gente passa as variáveis e os parâmetros dentro do próprio parênteses.

**Foreach**

**$nomes = array(“Alexandre”,”Leo”,”Mikaelle”,”Eloá”,”Arthur”);**

**foreach($nomes as $aluno){**

**echo “Aluno: “.$aluno.”<br/>”**

**}**

Quase a mesma função do For, porém ele transforma um Array numa apresentação listagem apresentada no as, podendo transformar o mesmo Array em vários objetos.

De array 1 para 2, 3, 4 ou até 5, outro exemplo:

**$nomes = array(**

**array(“nome”=>”Alexandre”,”idade”=>18),**

**array(“nome”=>”Leo”,”idade”=>30),**

**array(“nome”=>”Mikaelle”,”idade”=>26),**

**array(“nome”=>”Eloá”,”idade”=>3),**

**array(“nome”=>”Arthur”,”idade”=>14),**

**);**

**foreach($nomes as $aluno){**

**echo “Aluno: “.$aluno.[“nome”].” - ” Idade: “.$aluno.[“idade”].”<br/>”**

**}**

*Um outro exemplo para mostrar também o nome dos valores, pois senão imprime apenas os valores na tela:*

*$aluno = array(*

*“nome”=>”Alexandre”,*

*“idade”=>10,*

*“estado”=>”RJ”,*

*“país”=>”Brasil”);*

*Foreach($aluno as $chave => $dado){*

*Echo $chave.” = “$.dado.”<br/>”;*

*}*

**FUNÇÕES**

**Function soma (a, b){**

**Resultado = a + b;**

**}**

**Resultado da soma = fazer soma(10, 20);**

*Um outro exemplo para mostrar também o nome dos valores, pois senão imprime apenas os*

**MATEMÁTICA**

**Floor:** Arredonda o valor de modo decrescente

**Round:** Arredonda o valor baseado nele mesmo

**Abs:** Valor absoluto

**Ceil:** Arredonda o valor de modo crescente

**Rand:** Número aleatorio, exemplo rand(0, 10); vai puxar um numero aleatorio entre 0 e 10.

$lista = array(“Alexandre”, “Leo”, “Eloa”, “Arthur”);

x = round(0, 4);

echo “E o sorteado é: “.$lista[$x];

**TEXTOS**

**Explode:** Separa as arrays, código exemplo:

$nome = “Alexandre Santos”;

$x = explode(“PARTE QUE QUER SEPARAR, PODE SER UM NOME OU O ESPAÇO”, “$nome”);

Print\_r($x);

**Implode:** Junta as arrays, faz o contrário do apresentado acima para arrays separadas.

Como nome e sobrenome num input diferente.

**Number\_format:** Formata números longos, exemplo:

$x = number\_format(8.2631221, quantidade de numeros depois da virgula, se for 2);

echo $x; vai aparecer 8.26.

**str\_replace:** Código exemplo:

$texto = “O rato roeu a roupa!”;

$string = str\_replace(“palavra que quero trocar”, “por qual trocar”, $texto);

**Srttolower e srttoupper:** Após passar os parametros, cada um desses comandos ou deixa tudo minusculo ou tudo em caps.

**substr:** Para mostrar parte das strings, ex:

$texto = “Alexandre”;

$x = substr($texto, 0(posicaoinicial), 3(numerodepalavrasamostrar));

Echo $x;

**ucfirst:** Converte para maiúsculo o primeiro nome de uma String

**ucwords:**Converte para maiúsculo cada palavra da sua string.

**CRIPTOGRAFIA**

**Md5:** Para criptografar strings, senhas e afins ex:

$senha = alexandre90

$senhacripto = md5($senha);

Senha = hdfijhgdif6sdfds87fjsdf

**COOKIE**

**Setcookie(“parametrodocookie”,”parametrodocookie”, time()+3600);**

Para o cookie durar uma hora.

**PDO**

**Código para criar uma conexão ao Banco de Dados usando PDO.**

$dsn = “mysql:dbname=nomedodb;host=localhost”:

$dbuser = “root”;

$dbpass = “root”;

Try {

$pdo = new PDO($dsn, $dbuser, $dbpass);

Echo “Conexão estabelecida com sucesso!”;

} catch(PDOException $e) {

Echo “Falhou: “.$e->getMessage();

}

**Código apresentar os Nomes presentes no BD e na tabela Usuários, podendo fazer isso com emails, fotos, o que for.**

$dsn = “mysql:dbname=nomedodb;host=localhost”:

$dbuser = “root”;

$dbpass = “root”;

Try {

$pdo = new PDO($dsn, $dbuser, $dbpass);

$sql = “SELECT \* FROM usuarios”;

$sql = pdo->query($sql);

If($sql->rowCount() > 0){

Foreach($sql->fetchAll() as $usuario){

Echo “Nome: “.$usuario[“nome”].”<br>”;

}

} else{

Echo “Não há usuários cadastrados!”;

}

} catch(PDOException $e) {

Echo “Falhou: “.$e->getMessage();

}

**Código para inserir dados na tabela.**

Try {

$pdo = new PDO($dsn, $dbuser, $dbpass);

$nome = “Teste”;

$email = “[teste@gmail.com](mailto:teste@gmail.com)”;

$senha = md5(“teste”);

$sql = “INSERT INTO usuarios SET nome = ‘$nome’, email = ‘$email’, senha = ‘$senha’”;

$sql = $pdo->query($sql);

Echo “Usuário inserido com sucesso: ”.$pdo->lastInsertId();

} catch(PDOException $e) {

Echo “Falhou: “.$e->getMessage();

}

**Código para atualizar dados na tabela.**

Try {

$pdo = new PDO($dsn, $dbuser, $dbpass);

$sql = “UPDATE usuarios SET email = ‘novoemail@gmail.com’ WHERE email = ‘emailantigo@gmail.com’”;

$pdo->query($sql); //Update não retorna nada, então não precisar criar uma $sql antes do $pdo

Echo = “Usuário alterado com sucesso!”;

$sql = $pdo->query($sql);

Echo “Usuário inserido com sucesso: ”.$pdo->lastInsertId();

} catch(PDOException $e) {

Echo “Falhou: “.$e->getMessage();

}

**Código para deletar dados na tabela.**

Try {

$pdo = new PDO($dsn, $dbuser, $dbpass);

$sql = “DELETE FROM usuarios WHERE id = 12”;

$pdo ->query($sql);

Echo = “Usuário deletado com sucesso!”;

} catch(PDOException $e) {

Echo “Falhou: “.$e->getMessage();

}

**Código para fazer um delete dinamico (usuario se deletar), serve para INSERT, Update e afins na tabela.**

Try{

$pdo = new PDO($dsn, $dbuser, $dbpass);

$id = $\_GET[‘id’];

$nome = $\_GET[‘nome’];

$sql = “UPDATE usuarios SET nome = :novonome WHERE id = :id”;

$sql = $pdo->prepare($sql);

$sql->bindValue(‘:novonome’, $nome);

$sql->bindValue(‘:id’, $id);

$sql->execute();

} catch(PDOException $e) {

Echo “Falhou: “.$e->getMessage();

}

**Para enviar arquivos.**

<form *method*="POST" *enctype*="multipart/form-data">

    <input *type*="text" *name*="nome"><br><br>

    <input *type*="submit" *value*="Enviar">

</form>