



Escolas e Faculdades



Criando Uma Função para Criptografar URL e Código Fonte + Continuação de Inserir Dados e Listar Dados Banco de Dados -

**Andreza Barcaro e Brian
Neukamp
2020**

PHP OO + Composer + PHP Storm

Antes de Criar a Função Vamos Melhorar a listagem de Dados

```
<table class="table">
  <thead>
    <tr>
      <th scope="col">ID</th>
      <th scope="col">Nome</th>
      <th scope="col">Estado</th>
      <th scope="col">Mensagem</th>
    </tr>
  </thead>
  <tbody>

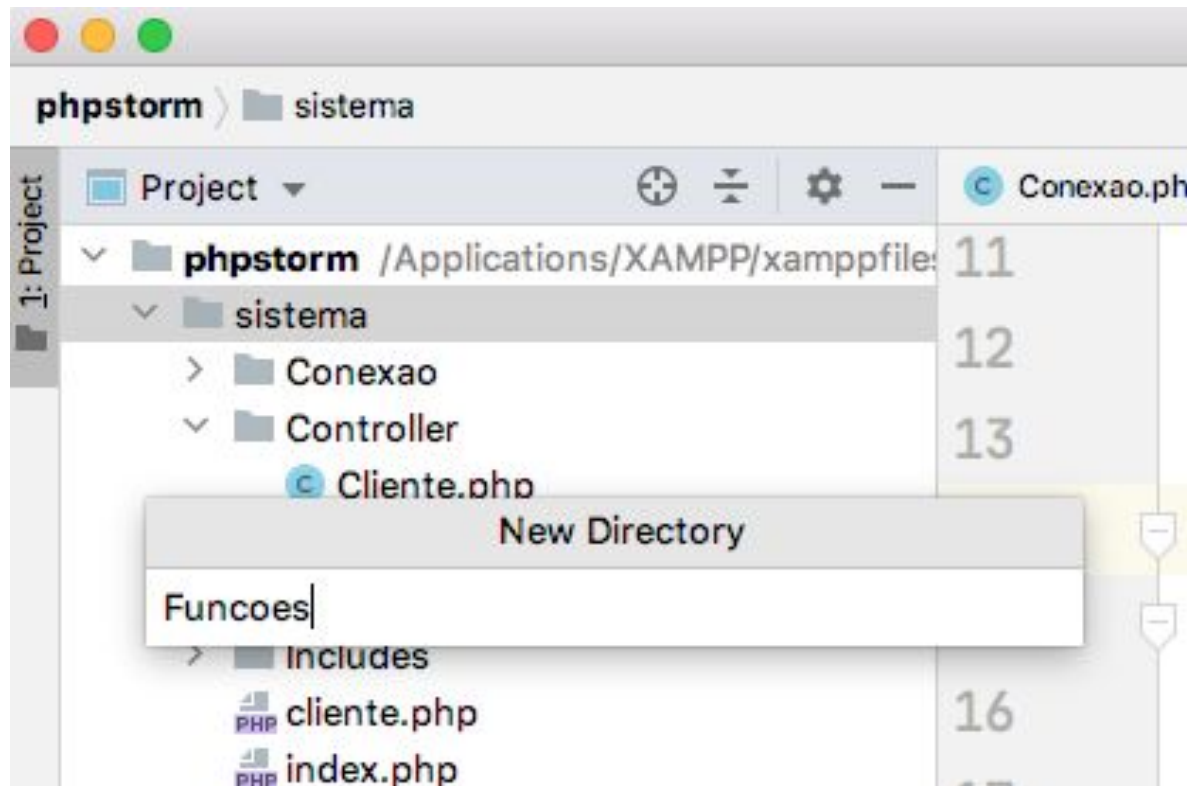
  <?php
    $contagem = 1;
    foreach ($a->querySeleciona() as $rst)
    {
      ?>
      <tr>
        <th scope="row"><?php echo $contagem++; ?></th>
        <td><?php echo $rst["nome"]; ?></td>
        <td><?php echo $rst["estado"]; ?></td>
        <td><?php echo $rst["mensagem"]; ?></td>
      </tr>
    <?php }?>
  </tbody>
</table>
```

Escolas e Faculdades



PHP OO + Composer + PHP Storm

Crie um novo Diretório Chamado Funcoes

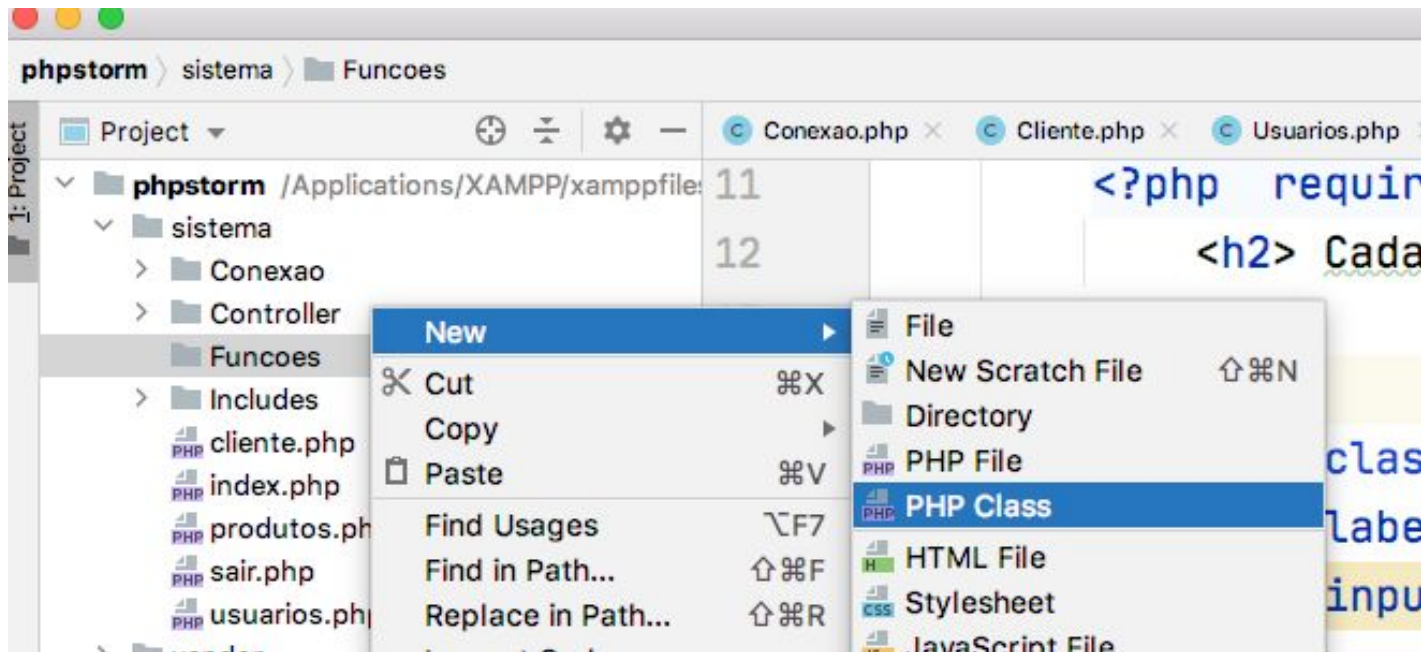


Escolas e Faculdades



PHP OO + Composer + PHP Storm

Crie uma nova Classe Chamada Funcoes isso servirá para editar e deletar

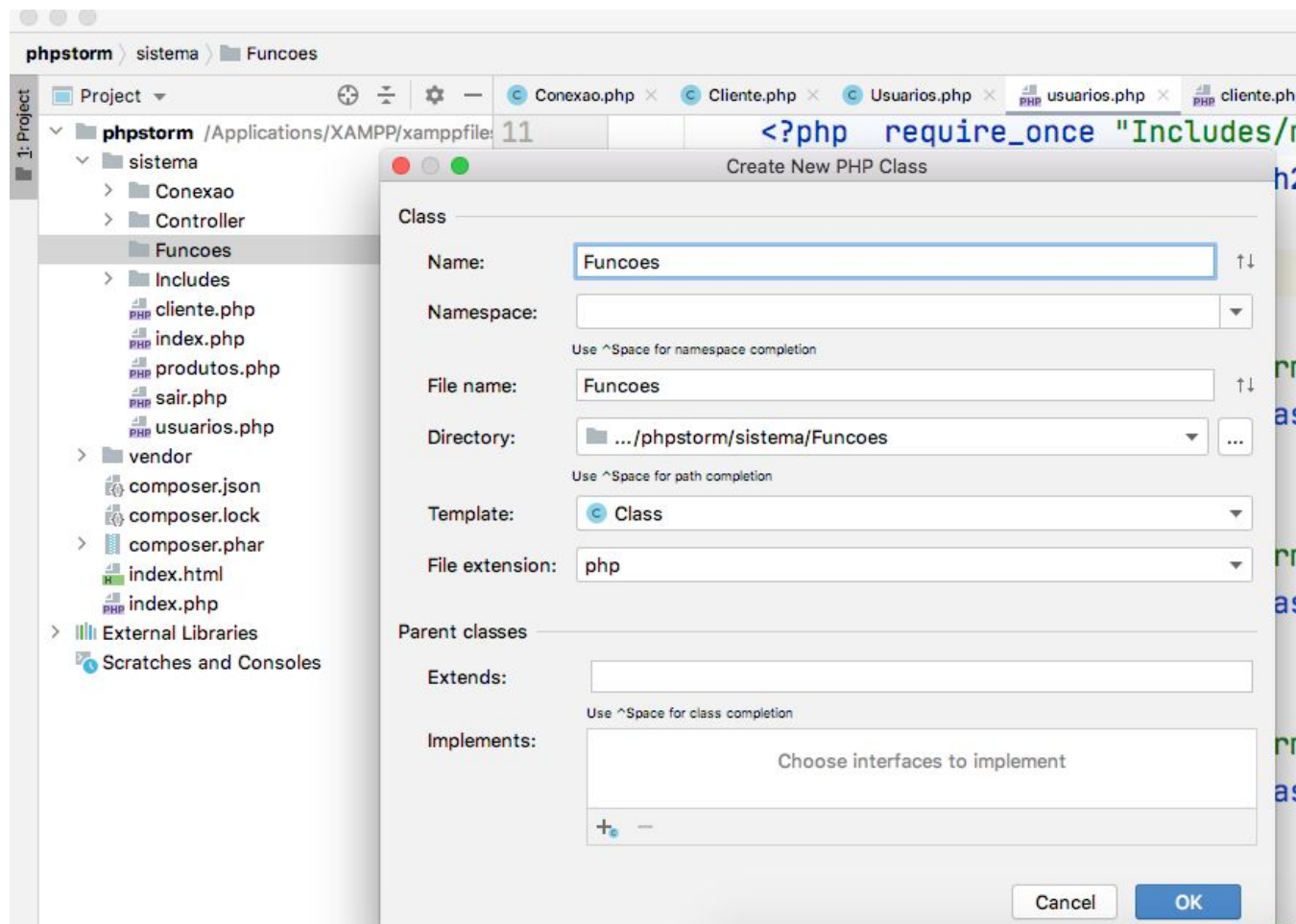


Escolas e Faculdades



PHP OO + Composer + PHP Storm

Crie uma nova Classe Chamada Funcoes isso servirá para editar e deletar



Escolas e Faculdades



PHP OO + Composer + PHP Storm

Crie uma nova Classe Chamada Funcoes isso servirá para editar e deletar



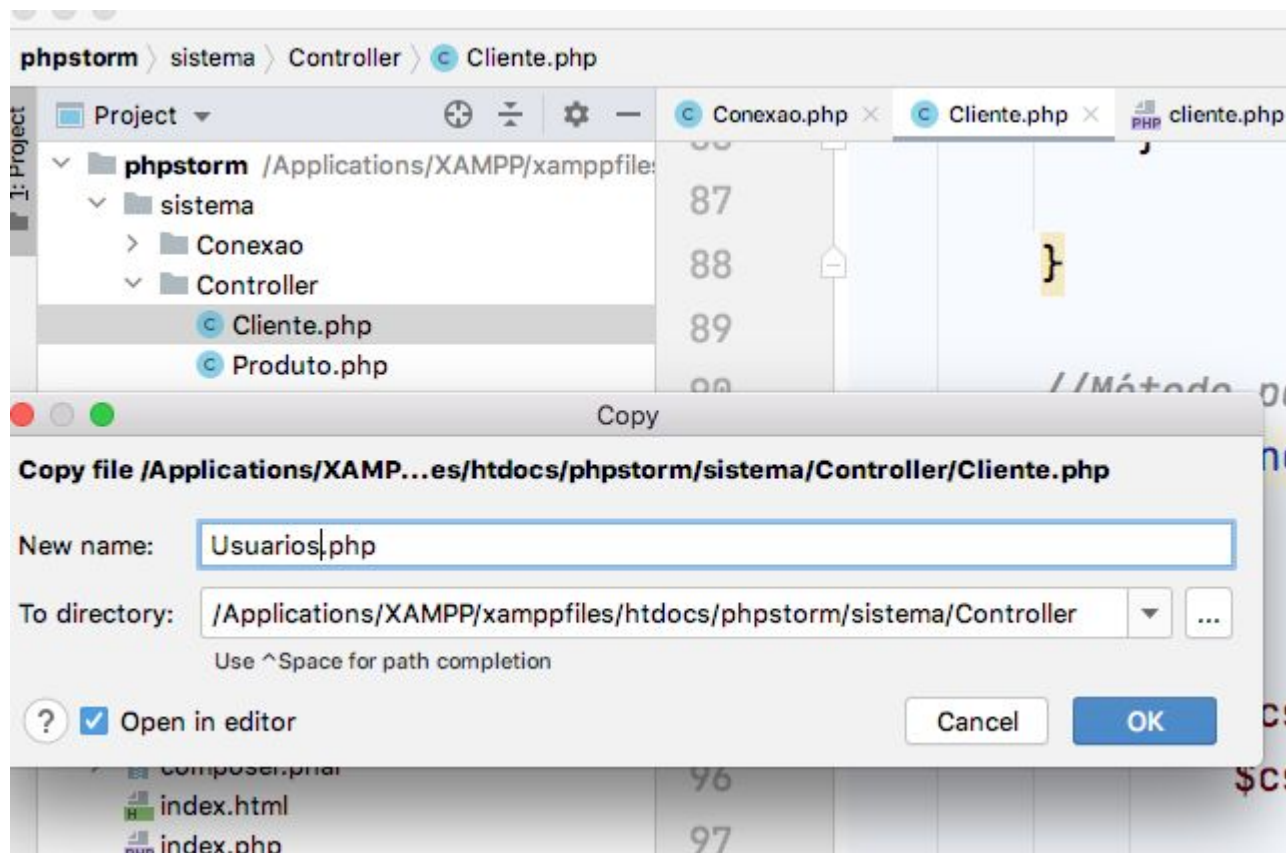
```
1 <?php
2
3 class Funcoes
4 {
5     public function base64 ($vlr, $tipo)
6     {
7         switch ($tipo)
8         {
9             case 1 : $rst = base64_encode($vlr); break;
10            case 2 : $rst = base64_decode($vlr); break;
11        }
12        return $rst;
13    }
14 }
```

Escolas e Faculdades



PHP OO + Composer + PHP Storm

Inserindo e Listando Dados Usuários Crie um nova Classe Chamada Usuarios pode duplicar de Cliente



Escolas e Faculdades



PHP OO + Composer + PHP Storm

Volte a classe Cliente e no construtor do da Classe adicione

```
//Instanciar a Conexao  
function __construct()  
{  
    $this->con = new Conexao();  
    $this->objfcn = new Funcoes();  
}
```


PHP OO + Composer + PHP Storm

Para Treinar voltamos ao inserir e selecionar - Import o autoload e Declare os Atributos de acordo com arquivo usuarios.php o name que vamos adicionar ao input do html

```
<?php

__DIR__ . "../vendor/autoload.php";

class Usuarios
{
    private $nome;
    private $email;
    private $senha;
    private $nivel;
    private $mensagem;
```

Após Gere os Métodos get e set

Escolas e Faculdades



PHP OO + Composer + PHP Storm

Inclua a Conexao e a Funcao

```
//Instanciar a Conexao  
function __construct()  
{  
    $this->con = new Conexao();  
    $this->objfcn = new Funcoes();  
}
```

Escolas e Faculdades



PHP OO + Composer + PHP Storm + Banco

Agora Vamos aos métodos de inserção e Listagem

```
//Método Inserir
public function queryInsert($dados)
{
    try
    {
        $this->nome = $dados["nome"];
        $this->email = $dados["email"];
        $this->senha = sha1($dados["senha"]);
        $this->nivel = $dados["nivel"];
        $this->mensagem = $dados["mensagem"];
        $cst = $this->con->conectar()->prepare( statement: "INSERT INTO usuarios (nome,email,senha,nivel,mensagem) VALUES(:nome,:email,:senha,:nivel,mensagem)" );
        $cst->bindParam( parameter: ":nome", &variable: $this->nome, data_type: PDO::PARAM_STR);
        $cst->bindParam( parameter: ":email", &variable: $this->email, data_type: PDO::PARAM_STR);
        $cst->bindParam( parameter: ":senha", &variable: $this->senha, data_type: PDO::PARAM_STR);
        $cst->bindParam( parameter: ":nivel", &variable: $this->nivel, data_type: PDO::PARAM_INT);
        $cst->bindParam( parameter: ":mensagem", &variable: $this->mensagem, data_type: PDO::PARAM_STR);

        if($cst->execute())
        {
            return "ok";
        }
        else
        {
            return 'erro';
        }
    }
    catch (PDOException $exc)
```

Escolas e Faculdades



PHP OO + Composer + PHP Storm + Banco

Agora Vamos aos métodos de inserção e Listagem

```
//Método Selecionar
public function selecionarDados()
{
    try
    {
        $cst = $this->con->conectar()->prepare( statement: "SELECT * FROM usuarios");
        $cst->execute();

        return $cst->fetchAll();
    }

    catch (PDOException $exc)
    {
        echo $exc->getTraceAsString();
    }
}
```

les



PHP OO + Composer + PHP Storm + Banco

No arquivo usuarios.php

```
<?php
    require_once "../vendor/autoload.php";

    $a = new Usuarios();

    if(isset($_POST["btCadastrar"]))
    {
        if($a->queryInsert($_POST) == "ok")
        {
            echo "Cadastro com Sucesso";
            header( string: "Location:http://localhost/phpstorm/sistema/usuarios.php");
        }
        else
        {
            echo "Não foi possível Realizar o cadastro";
        }
    }

?>
```

; e Faculdades



PHP OO + Composer + PHP Storm + Banco

No arquivo usuarios.php

```
<table class="table">
  <thead>
    <tr>
      <th scope="col">ID</th>
      <th scope="col">Nome</th>
      <th scope="col">Email</th>
      <th scope="col">Nível</th>
      <th scope="col">Mensagem</th>
    </tr>
  </thead>
  <tbody>


  <?php
  $contagem = 1;
  foreach ($a->querySeleciona() as $rst)
  {
    ?>
    <tr>
      <th scope="row"><?php echo $contagem++; ?></th>
      <td><?php echo $rst["nome"]; ?></td>
      <td><?php echo $rst["email"]; ?></td>
      <td><?php echo $rst["nivel"]; ?></td>
      <td><?php echo $rst["mensagem"]; ?></td>
    </tr>
  <?php }?>
  </tbody>
</table>
```

Escolas e Faculdades



PHP OO + Composer + PHP Storm + Banco

Criar a Tabela Usuários

 Vista de impressão  Data dictionary



Criar tabela

Nome:

Número de colunas: 

Escolas e Faculdades



PHP OO + Composer + PHP Storm + Banco

Criar a Tabela Usuários

	#	Nome	Tipo	Agrupamento (Collation)	Atributos	Nulo	Predefinido	Extra	Acções
<input type="checkbox"/>	1	id 	int(11)			Não	None	AUTO_INCREMENT	 Muda  Elimina  Primári
<input type="checkbox"/>	2	nome	varchar(256)	latin1_swedish_ci		Não	None		 Muda  Elimina  Primári
<input type="checkbox"/>	3	email	varchar(256)	latin1_swedish_ci		Não	None		 Muda  Elimina  Primári
<input type="checkbox"/>	4	senha	varchar(256)	latin1_swedish_ci		Não	None		 Muda  Elimina  Primári
<input type="checkbox"/>	5	nivel	int(11)			Não	None		 Muda  Elimina  Primári
<input type="checkbox"/>	6	mensagem	varchar(256)	latin1_swedish_ci		Não	None		 Muda  Elimina  Primári

Escolas e Faculdades



PHP OO + Composer + PHP Storm + Banco

Criar a Tabela Usuários

[Home](#) [Cliente](#) [Produtos](#) [Usuários](#) [Sair](#)

Cadastro Usuários

Nome do Usuário:

Email do Usuário:

Senha do Usuário:

Nível do Usuário:

Mensagem:

Cadastrar Usuário

ID	Nome	Email	Nível	Mensagem
1	nome	email	1	asdas

Dúvidas



Escolas e Faculdades



Referência Bibliográfica

- **Desenvolvimento Web com HTML, CSS e JavaScript.** Caelum, Disponível via web em <www.caelum.com.br>. 2015.
- GLAZAR, Jean Eduardo. **Programação para Web.** IFES, 2011.
- **Aprender PHP.** <http://aprenderphp.com.br>