CRUD com PDO MySQL

fatec – senai centro | Belo Horizonte

CRUD: PDO, Mysql e XAMPP

RUBEM CÂNDIDO

2021

Sumário

[Introdução 2](#_Toc62022377)

[Ferramentas necessárias 2](#_Toc62022378)

[XAMPP 2](#_Toc62022379)

[Git 2](#_Toc62022380)

[Conclusão 2](#_Toc62022381)

# Introdução

O propósito desse tutorial é criar um CRUD utilizando PDO e MySQL.

CRUD é um acrónimo do inglês Create, Read, Update e Delete. Representam as quatro operações básicas Inserir, Buscar, Atualizar e Deletar dados, utilizadas em banco de dados relacionais.

PDO acrónimo do inglês para *PHP Data Object* é uma extensão orientada a objetos da linguagem PHP para acesso a banco de dados.

XAMPP é um software que automatiza a instalação de programas necessários para gerar um ambiente de desenvolvimento.

MySQL ou MariaDB é um sistema de gerenciamento de banco de dados.

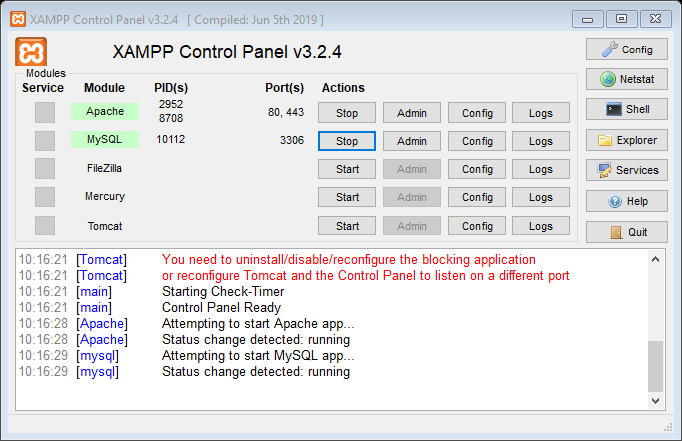
# Ferramentas necessárias

## XAMPP

## Editor de texto.

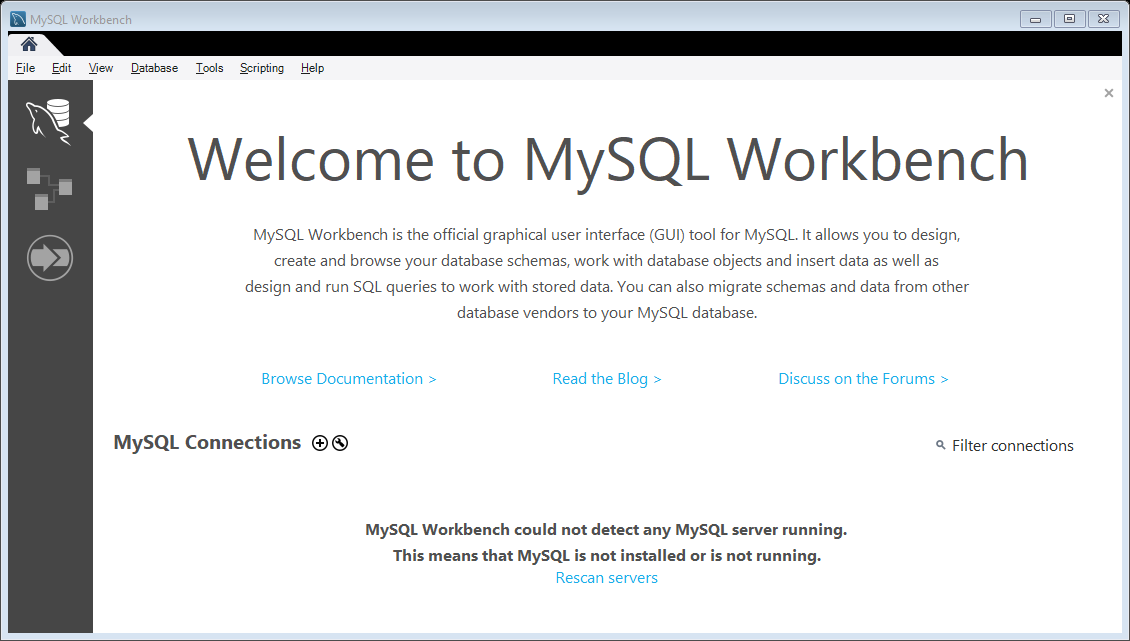
# Start o XAMPP

Para criar um servidor web, precisamos de iniciar o Apache. Para termos acesso à um SGBD, inicie o MySQL.

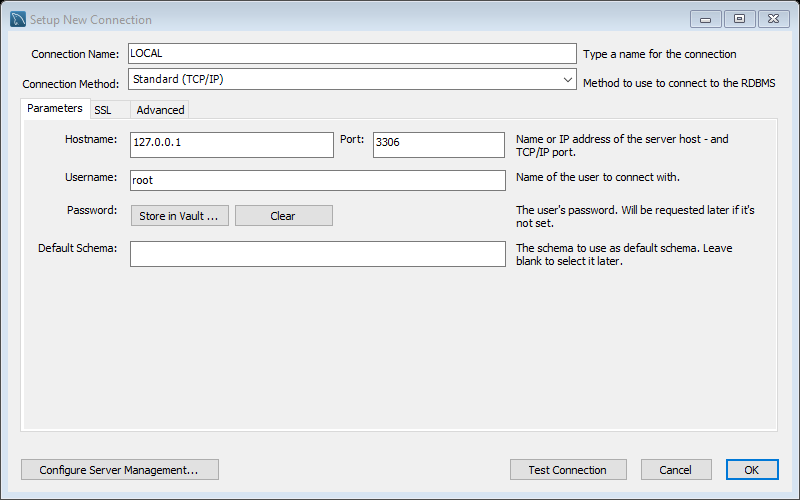


# Criando uma instancia no banco de dados

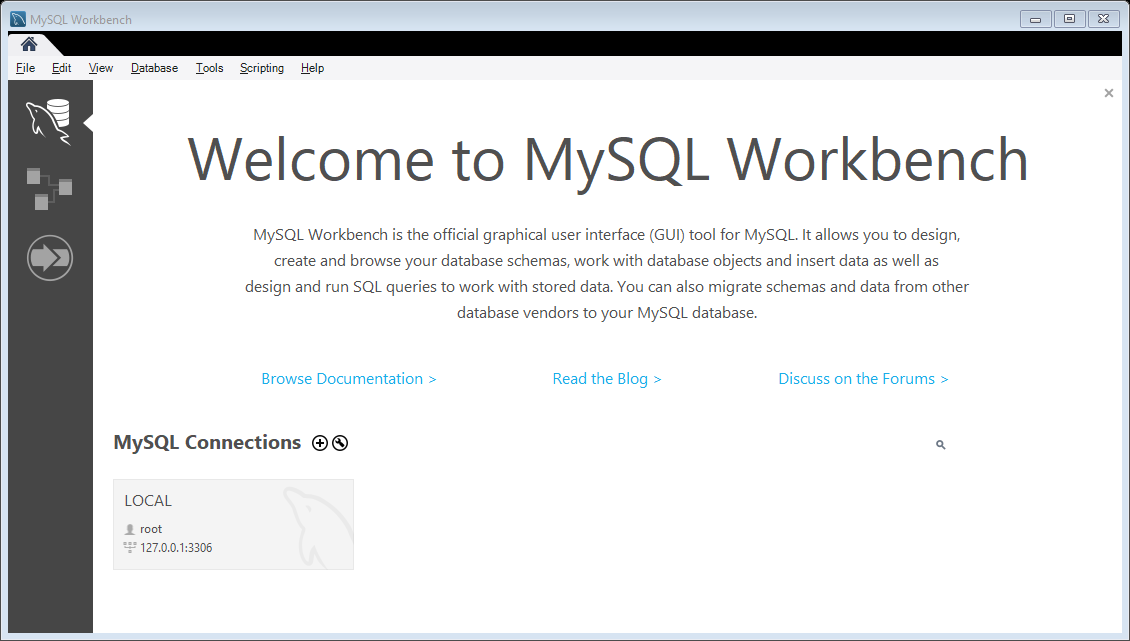
Abra o Workbench e crie uma conexão clicando em MySQL Connection +



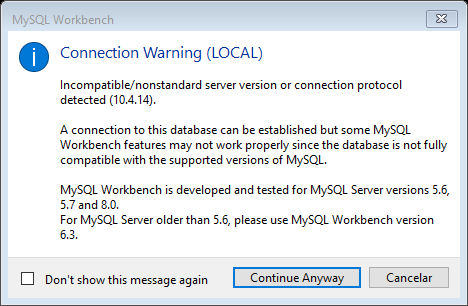
Para uma conexão local e padrão, basta nome.



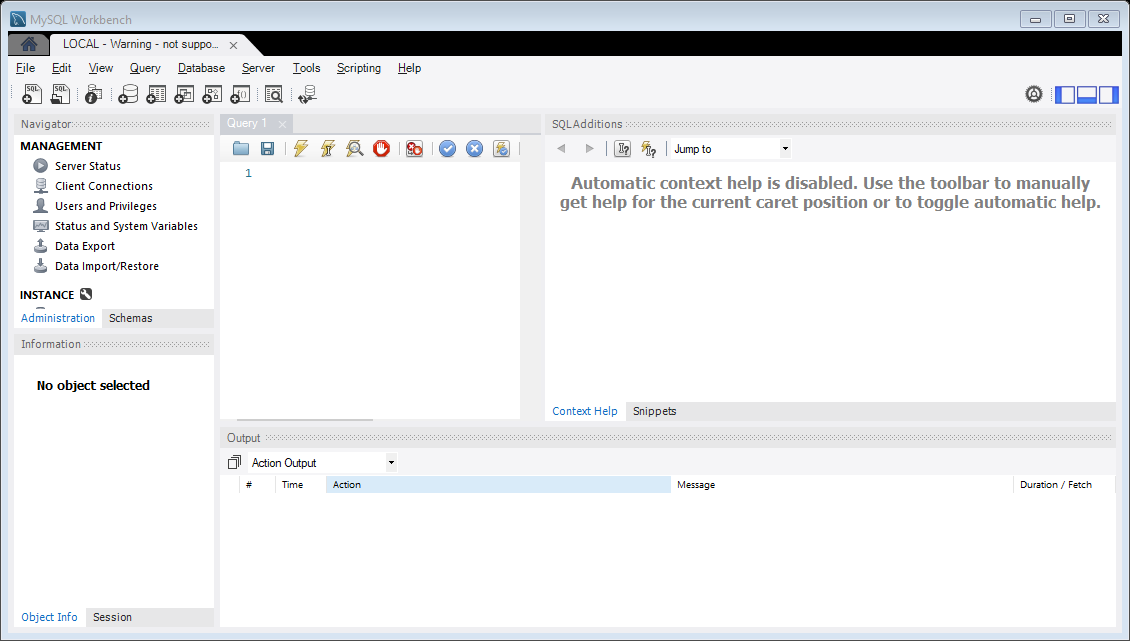
Ao clicar em OK. Duplo clique na conexão



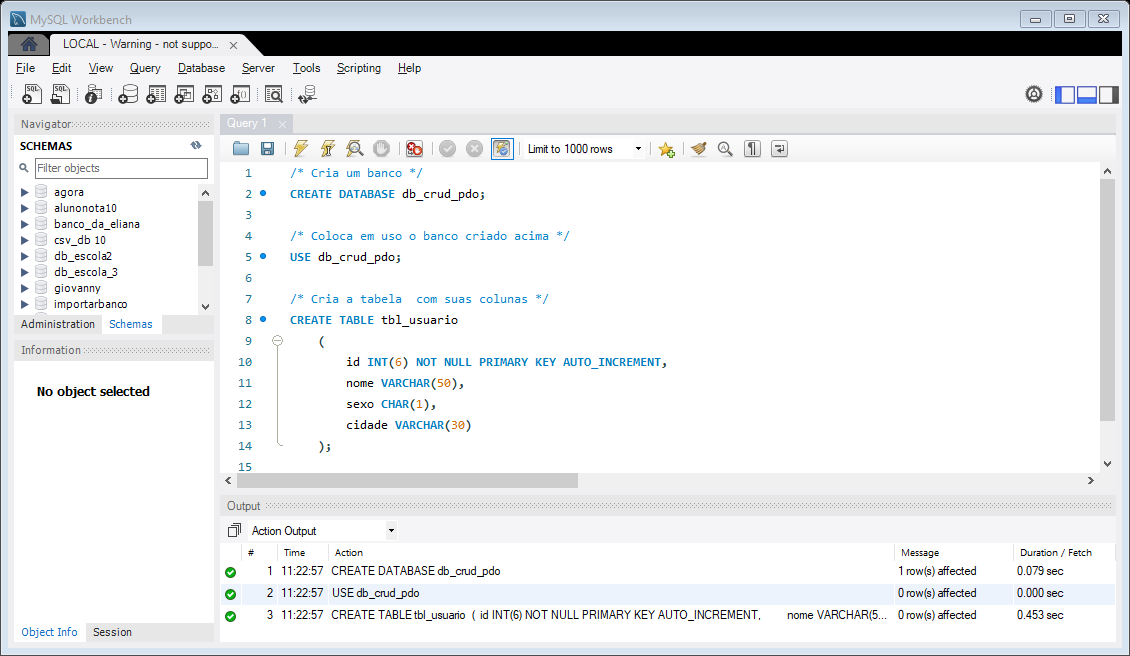
A primeira vez que o workbench se conecta no MySQL. Marque Continue Anyway.



Interface do Workbench

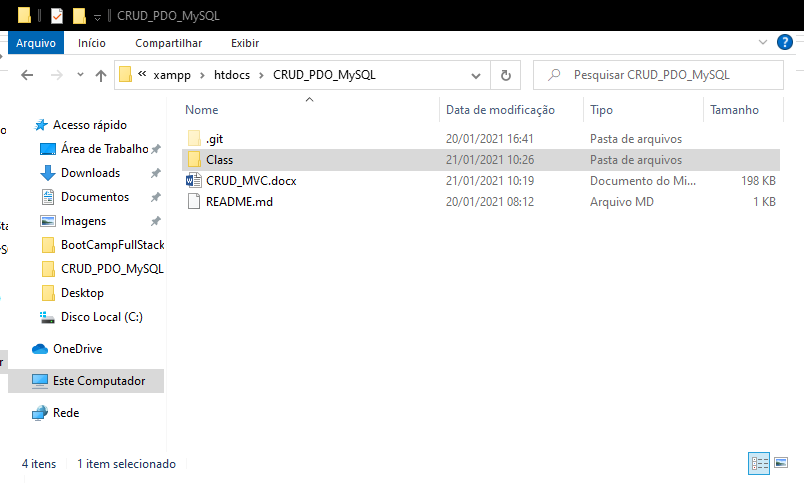


Criando o banco e uma tabela com suas colunas

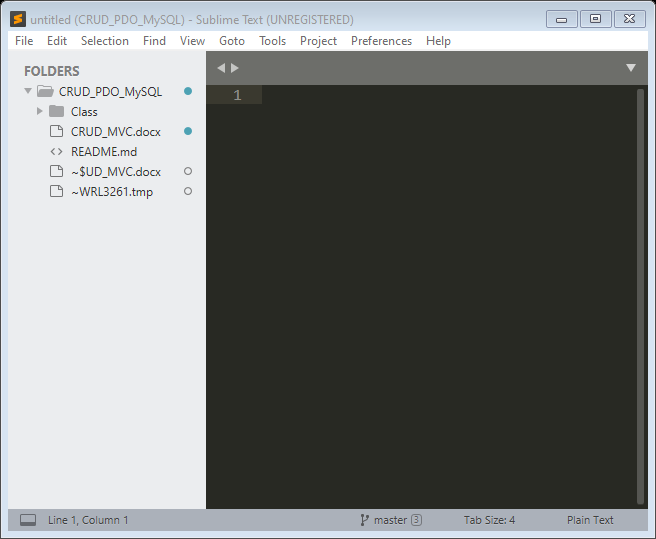


# Criando a classe

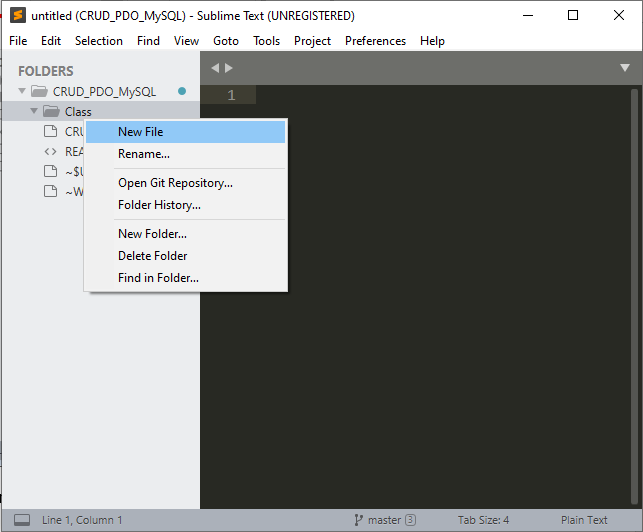
Crie um diretório chamado Class na raiz do seu projeto.



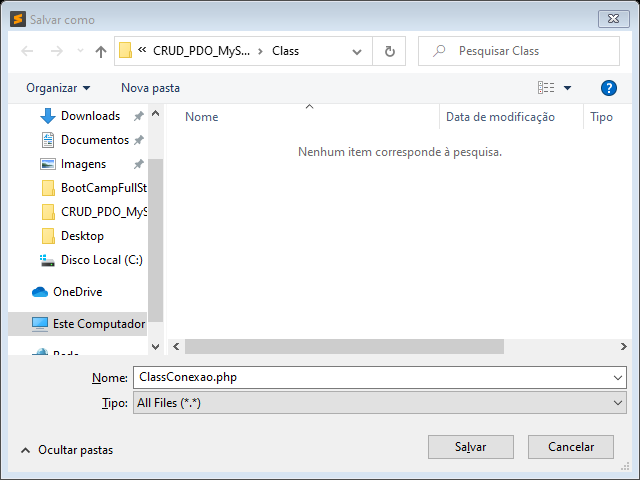
Abra o folder com seu editor de texto.



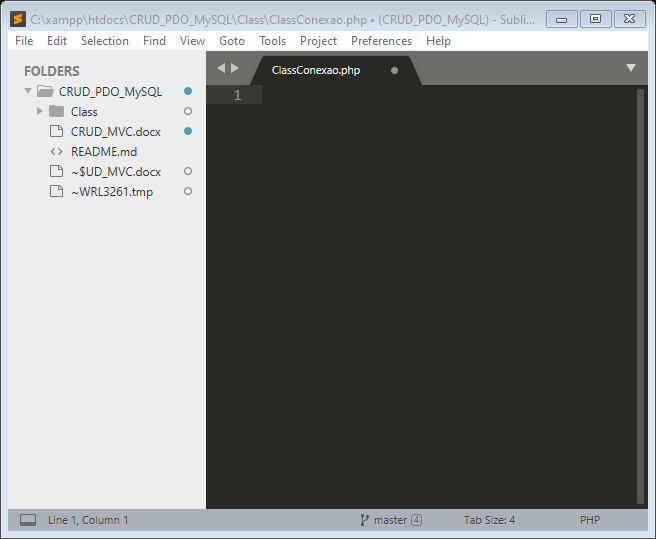
Crie um arquivo chamado ClassConexao.php

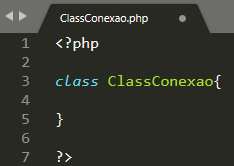


Salvando com o atalho Ctrl + S

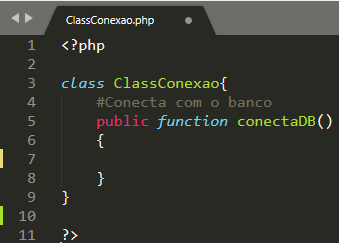


Arquivo criado





# Criando uma função para conectar com o banco



## try catch

Vamos tratar possíveis erros com um try catch.

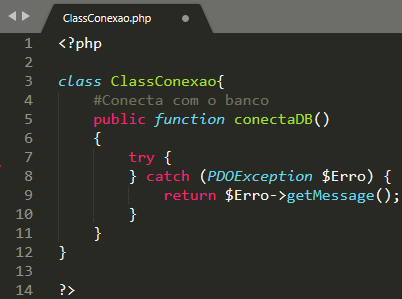
Atalho: Digite try e pressione a tecla tab para auto completar



Vamos nomear de acordo com nossa aplicação

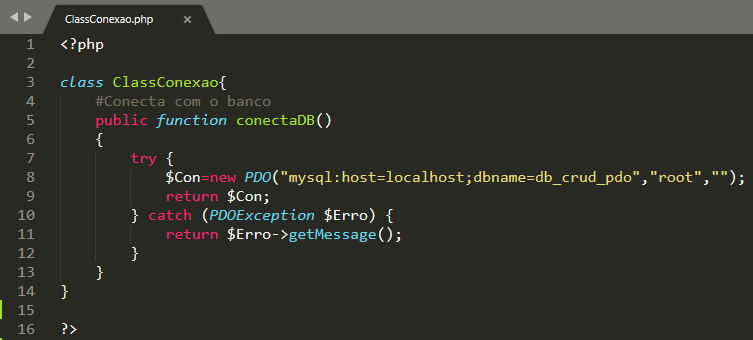


O catch irá retornar uma mensagem de erro caso a conexão falhe.



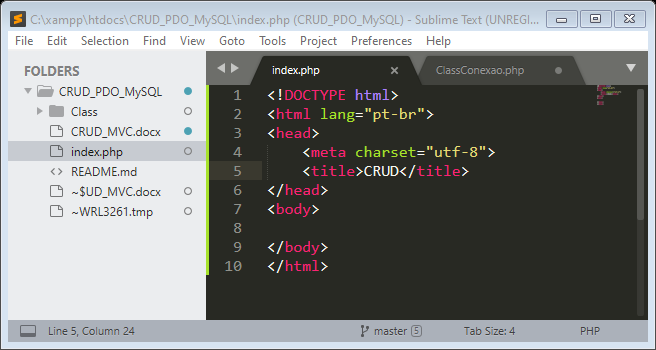
## PDO

Criando a variável de conexão com PDO



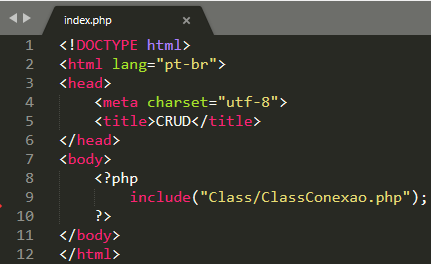
# Index.php

Na raiz do projeto, crie um arquivo chamado index.php com o esqueleto padrão do HTML5

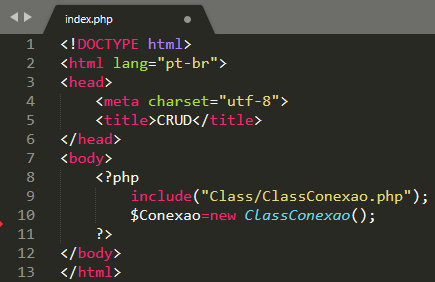


# include()

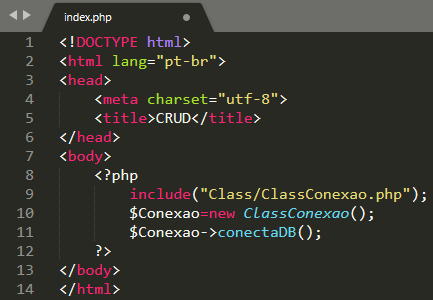
Com a função include, vamos referenciar a classe para dentro do index.php



Instanciar a classe ClassConexao()

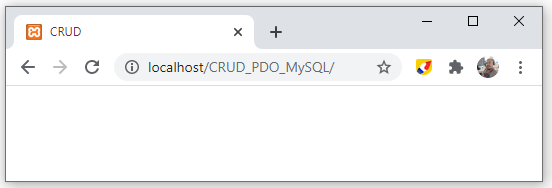


Vamos chamar a função



# Teste Unitário - Conexão

Testando a conexão, vamos abrir o navegador e digitar o caminho do nosso projeto: http://localhost/CRUD\_PDO\_MySQL/

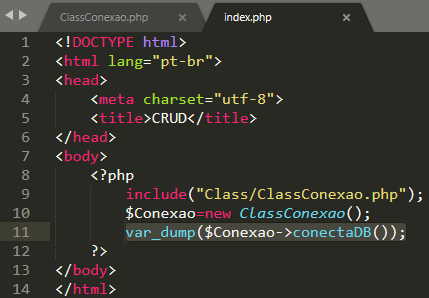


Tudo certo.

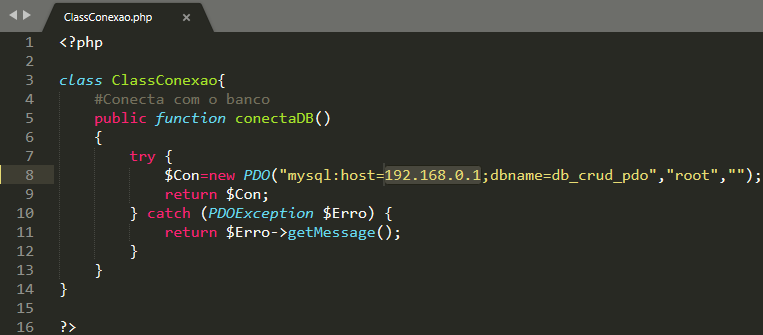
Vamos simular alguns erros:

## var\_dump()

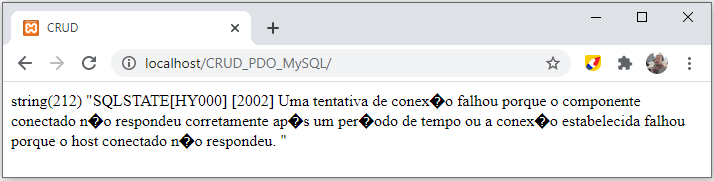
Acrescente a função var\_dump() para debugar a conexão.



Altere o host

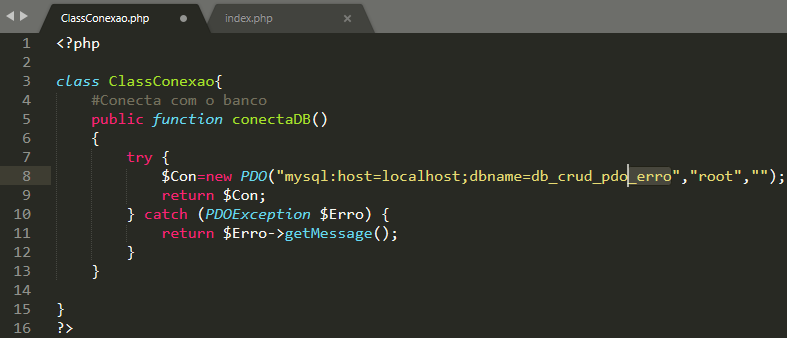


Atualize a página:

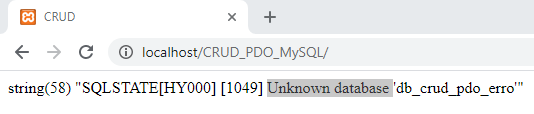


Volte o host para localhost.

Altere o nome do banco para forçar um erro

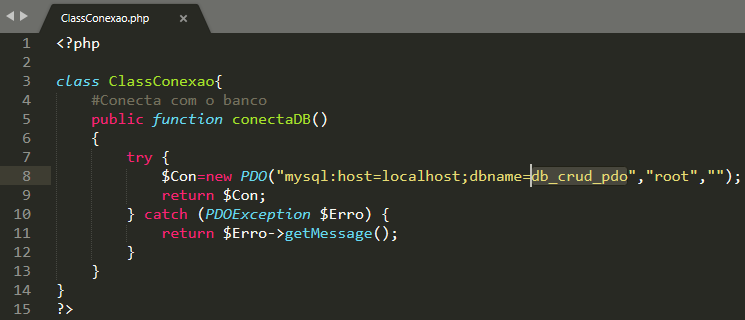


Atualize a página

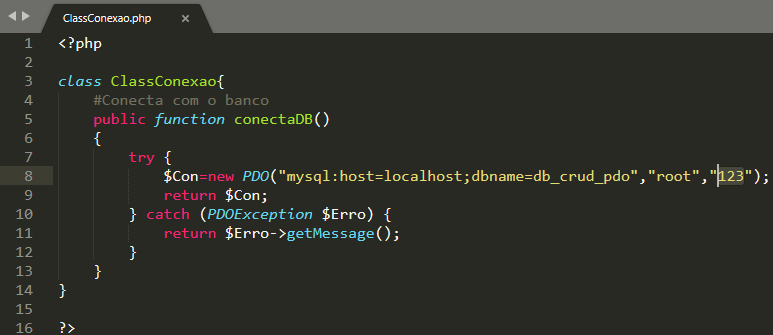


Banco de dados não encontrado.

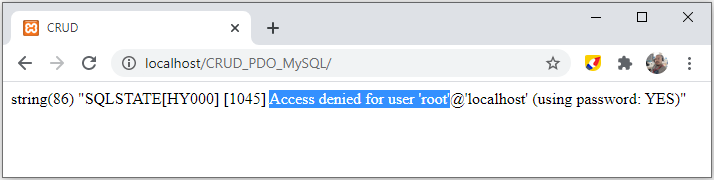
Corrija o nome do banco



Agora insira uma senha incorreta

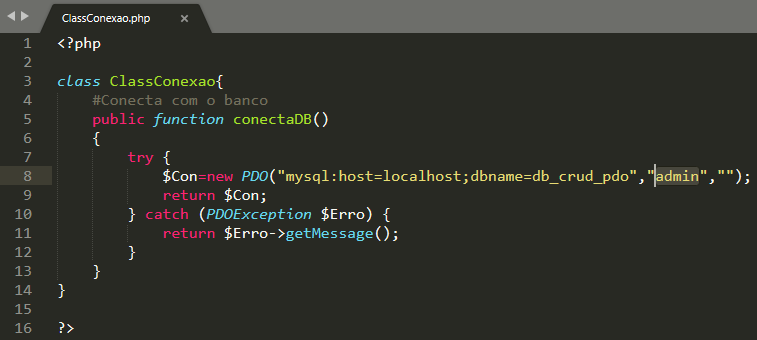


Atualize a página:

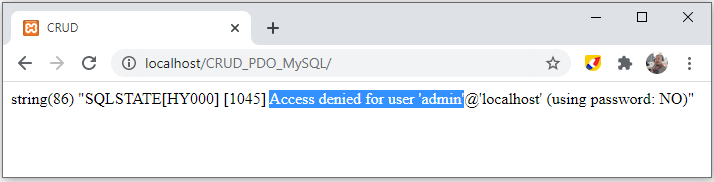


Acesso negado para o usuário root

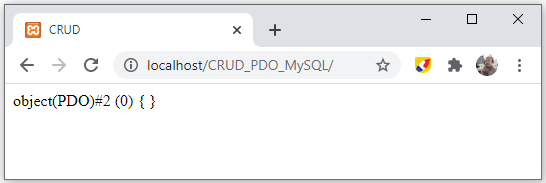
Corrija a senha e altere o usuário:



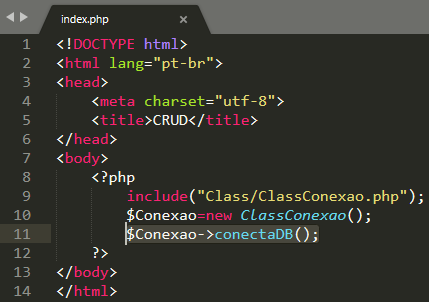
Atualize a página:



Mensagem esperada



Retire o var\_dump



# Git

Endereço do repositório no git:

https://github.com/endroni/CRUD\_PDO\_MySQL.git

# Conclusão

Uma estruturação em quatro camadas pode ser um tanto complexo no início, porém possibilita um grande reaproveitamento em trabalhos que possuem funções semelhantes.

Espero que tenha valia para seus estudos.

**BIBLIOGRAFIA BÁSICA:**

PRESSMAN, Roger S. **Engenharia de Software: uma abordagem profissional**. 7. ed. São Paulo: Pearson Makron Books, 2011.

SOMMERVILLE, Ian. **Engenharia de Software**. 8. ed. São Paulo: Pearson Addison Wesley, 2007.

TONSIG, Sergio Luiz. **Engenharia de Software: análise e projeto de sistemas**. 2. ed. Rio de Janeiro: Ciência Moderna, 2008.

**BIBLIOGRAFIA COMPLEMENTAR:**

BEZERRA, Eduardo. **Princípios de Análise e Projeto de Sistemas com UML**. Rio de Janeiro: Elsevier, 2007.

UDEMY. **Programando em MVC com C#.** Hugo Vasconcelos