

Desenvolvedor

Avaliação

Nome: Kaio Masaru Nakazono

Você deve utilizar o seguinte stack:

- PHP
- HTML, CSS (Bootstrap, Material Design), JS (Opções: JQuery, VueJS, Nativo)
- MySQL
- Composer (se necessário)

Entrega

Você deve criar um fork do repositório no seu GitHub, assim que finalizar o projeto, nos envie o link por email

1. É possível utilizar `mysql_query` na versão 7.2 do PHP? Se sim, escreva um exemplo de realizar uma consulta no banco de dados MySQL, se não, informe como utilizar e faça um exemplo.

Resp: A função `mysql_query` foi removida da versão do PHP 7. Podemos usar o `Mysqli` (Mysql Improvement) ou o `PDO` (PHP Data Objects) para trabalhar com o banco de dados. O `PDO` tem a vantagem de trabalhar com outros banco de dados, como o `Postresql`; enquanto que o `Mysqli` só funciona com o banco `Mysql`.

```
<?php
//variaveis para conexao com o BD
$servername = "localhost";
$usernameDB = "user";
$passwordDB = "suaSenha";
$database = "seuDatabase";

try{
    //cria uma conexao
    $conn = new PDO("mysql:host=$servername; dbname=$database", $usernameDB, $passwordDB);

    //cria uma consulta e a executa
    $stmt = $conn->prepare('select * from clientes');
    $stmt->execute();

    //retorno do resultado, neste caso virá no formato associativo key->value
    $result = $stmt->fetchAll(PDO::FETCH_ASSOC);
}
catch(PDOException $e){
    echo "Falha: " . $e->getMessage();
}
?>
```

2. Descreva os serviços necessários para uma arquitetura web que dê suporte a um sistema desenvolvido em PHP:

Resp:

- Navegador para o cliente acessar o site.
- Página HTML onde o cliente possa interagir com a aplicação.
- Requisição HTTP, que é um protocolo de troca de informações entre o cliente e o servidor.
- Servidor com Apache instalado. O Apache irá receber a requisição vinda do cliente e irá executar o arquivo .php solicitado, podendo redirecionar ou não o tráfego para o banco de dados. Irá retornar uma página HTML.
- Banco de dados relacional como o Mysql ou Postresql.

3. Criar uma classe que possa ler o conteúdo de um arquivo texto. Essa classe deverá conter um atributo privado que será usado para guardar o conteúdo do arquivo lido, um método para carregar o arquivo e outro para visualizar o atributo.

Resp: arquivo no Github

4. Utilizando a classe criada acima, criar um form que solicite ao usuário um arquivo e que ao submeter seja visualizado o conteúdo.

Resp: arquivo no Github

5. Construa um exemplo de sessão logada
Você deve utilizar: PHP, Javascript e HTML.

Resp: arquivo no Github

6. Comente a função abaixo informando o que faz e retorna

```
function NO_Test($class_name) {  
    try{  
        if(class_exists($class_name)){  
            return;  
        }  
    }  
}
```

```
        $path = explode(PATH_SEPARATOR, get_include_path());
        foreach($path as $dir){
            if (is_file($dir . DIRECTORY_SEPARATOR .
$class_name . '.class.php' )){
                require_once $dir . DIRECTORY_SEPARATOR .
$class_name . '.class.php';
                return;
            }else if (is_file($dir . DIRECTORY_SEPARATOR .
strtolower($class_name) . '.cls.php' )){
                require_once $dir . DIRECTORY_SEPARATOR .
strtolower($class_name) . '.cls.php';
                return;
            }
        }
    }catch(Exception $e){
        trigger_error("ERRO: \n\nDetalhes:" .
$e->getMessage(), E_USER_ERROR);
    }
}
```

Resp:

A função verifica se determinada classe existe. Caso exista, retorna. Caso contrário guarda na variável '\$path' do tipo array uma lista de diretórios onde as funções require, include, fopen(), file(), readfile() e file_get_contents() irão procurar por arquivos.

O loop foreach irá iteragir para cada diretório verificando se o nome do arquivo especificado é um arquivo regular. Se sim, irá copiar esses arquivos que existem nesse diretório para dentro do arquivo atual. Caso não entre em nenhuma condição, não irá retornar nada.

7. Atividade. Construa um simples CRUD para cadastro de clientes com a seguinte estrutura:

- ID
- Nome
- CPF ou CNPJ (utilizar máscara)
- E-mail
- Latitude
- Longitude

- Datas para cadastro, modificação e última atividade

Requisitos

- Utilizar Bootstrap ou Material Design
- As interações entre o Front e o Back-end devem ser feitas via AJAX utilizando JSON
- Classes públicas
- Qualidade de código
- Limpeza
- Criatividade

Você deve enviar os arquivos (html,php,css,js,img ou GIT) e os scripts .sql (CREATE TABLE)

8. Escreva uma Query para MySQL para a seguinte situação: Selecionar usuários que tem o tempo de 'ultima_atividade' menor que 5 minutos. (você pode simular o conteúdo de uma tabela de usuários comum, ex.: nome, usuário, senha, dt_cad, ...)

Resp:

```
select usuario, DATE_FORMAT(ultima_atividade, '%Y-%m-%d
%H:%i') from table_users where ultima_atividade >=
CURRENT_TIMESTAMP - INTERVAL 5 MINUTE;
```