**Pacote "Lou Paginar Mysql Class"**

Este pacote PHP, denominado " Lou Paginar Mysql", fornece uma forma reutilizável e organizada de implementar a paginação para dados recuperados de uma base de dados MySQL. É constituído por dois componentes principais:

As principais vantagens são:

1. Encapsulamento:

A ligação à base de dados e a lógica de paginação são encapsuladas dentro das classes, tornando o código mais organizado e reutilizável.

1. Lógica centralizada:

Todo o código relacionado com a paginação está num só local, facilitando a manutenção e a modificação.

1. Utilização simplificada:

O DatabasePaginationWrapper simplifica o processo de implementação de paginação, exigindo código mínimo no ficheiro principal da aplicação.

1. Código Limpo:

O código está bem estruturado e comentado, facilitando a compreensão.

1. Reutilizável:

As classes podem ser facilmente reutilizadas noutros projetos.

1. Separação clara de preocupações:

A ligação à base de dados e a lógica de paginação estão separadas em classes diferentes.

Na sua essência, este pacote é uma solução bem concebida e orientada a objetos para adicionar paginação a aplicações PHP que interagem com bases de dados MySQL.

Classe DatabasePaginationWrapper:

Objetivo:

Esta classe atua como um wrapper, encapsulando a ligação à base de dados e a lógica de paginação. Simplifica o processo de ligação à base de dados, recuperando resultados paginados e exibindo links de paginação.

Características:

1. Ligação à base de dados: trata da ligação à base de dados MySQL usando mysqli.
2. Tratamento de erros: verifica os erros de ligação e termina a execução se ocorrer um erro.
3. Integração de Paginação: Cria uma instância da classe Paginação para gerir a lógica de paginação.

Métodos:

* ExibirResultados():
  + Recebe uma consulta SQL como entrada.
  + Recupera os resultados paginados da base de dados utilizando a classe Paginacao.
  + Apresenta os links de paginação (Primeiro, Anterior, números de página, Seguinte, Último).
  + Retorna o conjunto de resultados da página atual.
* FecharConexao(): Fecha a ligação à base de dados.
* Construtor:
  + Toma como argumentos os parâmetros de ligação à base de dados (host, utilizador, palavra-passe, nome da base de dados) e o número de resultados por página.
  + Estabelece a ligação com a base de dados.
  + Inicializa o objeto Paginação.

Class de paginação:

Objetivo:

Esta classe lida com a lógica de paginação principal. Calcula o número total de páginas, determina a página atual e gera os links de paginação.

Métodos

* obterResultados():
  + Recebe uma consulta SQL como entrada.
  + Conta o número total de resultados que correspondem à consulta.
  + Calcula o número total de páginas.
  + Determina o OFFSET da página atual.
  + Acrescenta as cláusulas ORDER BY id DESC e LIMIT à consulta para recuperar apenas os resultados da página atual.
  + Executa a consulta modificada e devolve o conjunto de resultados.
* getLinks():
  + Gera os links HTML para paginação (Primeiro, Anterior, números de página, Seguinte, Último).
  + Constrói os URLs para cada link, incluindo o parâmetro página na string de consulta (por exemplo, index.php?pagina=2).
  + Retorna os links como uma string, separados por " | ".
* Construtor:
  + Toma como argumentos o objeto de ligação à base de dados e o número de resultados por página.
  + Define o número de resultados por página.
  + Determina a página actual a partir do parâmetro $\_GET['pagina'] (o predefinido é 1 se não estiver definido).

Funcionamento (em index.php)

* Inclui o ficheiro paginate.php, disponibilizando as classes.
* São definidos os detalhes da ligação à base de dados (host, utilizador, palavra-passe, nome da base de dados) e o número de resultados por página.
* É criado um objeto DatabasePaginationWrapper, passando os detalhes da configuração para o seu construtor.
* É definida uma consulta SQL (por exemplo, SELECT \* FROM noticias).
* O método exibirResultados() do wrapper é chamado, passando a consulta.
* Recupera os resultados paginados.
* Apresenta os links de paginação.
* Retorna o conjunto de resultados da página atual.
* Itera pelo conjunto de resultados e apresenta os dados de cada linha.
* Fechar ligação