

A Classe SQLHelper

No SISMETRO Cloud, devido a sua estruturação desde o início, o mesmo foi desenvolvido com PHP puro e assim criado os SQLs sempre de forma manual.

Com isso, para facilitar este processo, principalmente para quando estamos efetuando a migração para o PDO, foi criada a classe SQLHelper que tem como objetivo de facilitar a criação de SQLs para serem usados nas consultas, inserções, atualizações e exclusões.

No momento a classe aceita consultas mais simples, mas a ideia é aos poucos serem implementados métodos para facilitar na geração de SQLs mais complexos.

Segue exemplo mínimo:

```
$sqlHelper = new SQLHelper;  
  
$sqlHelper->setMethod(SQLHelperMethod::SELECT);  
  
$sqlHelper->setTable("drawos");  
  
$sqlHelper->addWhereField("idos", ":idos");  
  
echo $sqlHelper->generateSQL();
```

No exemplo ele vai gerar um SQL igual a este:

```
SELECT * FROM drawos WHERE idos = :idos
```

Sendo este um pequeno exemplo dos usos possíveis da Classe.

ESTRUTURA BÁSICA

Toda vez que necessite criar um SQL é necessário fazer a instância do SQLHelper, para obter um objeto limpo para ser trabalhado.

Após a criação de um novo objeto é necessário indicar duas informações: O método e a tabela.

Para o método é usada a função setMethod() indicando pela enum SQLHelperMethod qual é o método a ser usado, sendo um dos possíveis:

- SQLHelperMethod::SELECT
- SQLHelperMethod::INSERT
- SQLHelperMethod::UPDATE

- SQLHelperMethod::DELETE

Já para definir qual é a tabela usa-se a função `setTable()` passando uma string com o nome da tabela. Se quiser colocar uma referência para uso na consulta também é possível, fazendo assim como no caso de uso de um SQL normal, por exemplo:

- `$sqlHelper->setTable("drawos os");`
- `$sqlHelper->setTable("drawequip equip");`
- `$sqlHelper->setTable("drawplanos plano");`

E após a parametrização de todos os parâmetros para a geração do SQL, é só chamar a função `generateSQL()` que é efetuada a criação do SQL e retornando um string com ele.

O WHERE

Além dos métodos já apresentados, existem alguns métodos focados no gerenciamento do WHERE, sendo os seguintes:

- `addWhereField($column, $field, $type)`
- `hasWhereField($field)`
- `getWhereFieldIndex($field)`
- `removeWhereField($i, $field)`

addWhereField(\$column, \$field, \$type)

`$column` = nome da coluna a ser usada no where, podendo ser usado a referência da tabela como usado em um SQL normal (obrigatório)

`$field` = nome do campo que será usado no `bindParam()` ou `bindValue()` para ser preenchido no SQL (obrigatório condicionalmente)

`$type` = tipo de comparação a ser usada, sendo o valor padrão `SQLHelperWhereType::EQUALS`

Para o campo `type` é possível o uso dos seguintes tipos:

- `SQLHelperWhereType::EQUALS` (igual/igualdade)
- `SQLHelperWhereType::IS_NULL` (nulo)
- `SQLHelperWhereType::IS_NOT_NULL` (não nulo)
- `SQLHelperWhereType::NOT_EQUALS` ou `SQLHelperWhereType::DIFFERENT` (mesmo comportamento - diferente do valor)
- `SQLHelperWhereType::BIGGER_THAN` (maior que)
- `SQLHelperWhereType::BIGGER_OR_EQUALS_THAN` (maior ou igual a)
- `SQLHelperWhereType::SMALLER_THAN` (menor que)
- `SQLHelperWhereType::SMALLER_OR_EQUALS_THAN` (menor ou igual a)

Se `SQLHelperWhereType::IS_NULL` ou `SQLHelperWhereType::IS_NOT_NULL` não é necessário indicar o campo `$field` pois ele já estará no SQL indicando que aquele campo IS NULL ou IS NOT NULL. Caso contrário, o campo `$field` é obrigatório.

Caso indique um `$field` que já existe, ele retornará um array com "status" = false, e em "reason" um valor da classe `SQLHelperError` indicando qual o erro.

hasWhereField(\$field)

Indica, retornando um boolean, que se aquele `$field` já existe nesta consulta. Importante dizer que ele valida apenas os valores de `$field` usados para o WHERE.

getWhereFieldIndex(\$field)

Caso o `$field` exista nos WHEREs, ele retorna o index desse `$field`. É usado geralmente para remover um `$field` do WHERE.

Caso não encontre, retornará um array com "status" = false e em "reason" um valor da classe `SQLHelperError` indicando qual o erro.

removeWhereField(\$i, \$field)

Serve para remover um WHERE, passando o `$i` (index do WHERE) ou o `$field`.

Revisão #2

Criado 29 August 2024 17:02:43 por João Polles

Atualizado 29 August 2024 17:35:29 por João Polles