

# Executores

## Objetivo

Foi desenvolvida uma forma de facilitar a execução de packages, queries e functions no banco de dados. Visando uma maior produtividade e exito na migração das rotinas ja existentes no STM VB. Os executores se encontram disponíveis na biblioteca Cebi.Stm.Executor que já está instalada nos módulos do stm web.

## Váriavel Global

Todos os executores, necessitam da carga de váriavel global, elas já são instanciadas em todas as nossas controllers

## Querys

O Executor de query é o mais simples, voce deverá montar a consulta sql em uma string, e passar essa consulta como parametro no construtor do executor:

```
var executor = new ExecutorQueryCommand("SELECT * FROM cargos_cebi WHERE CAR_SEQ = " + id);
```

Esse executor possui 2 métodos:

- `GetSingle<T>()`
- `GetList<T>()`

O primeiro, irá retornar apenas 1 linha do resultado da query, indicado para consultas por chaves primárias.

Já o segundo, retorna todas as linhas.

“ Em ambos os métodos, deverá ser informada a classe de retorno no parametro T da execução  
Os campos da classe devem ter o mesmo nome dos campos de retorno. Deverá também ser informado no método a carga de variável global.

## Packages

Para a execução de packages, no nosso construtor passamos 2 parametros:

- Nome da Pkg a ser executada

- Se o retorno é um record set

```
public class ExecutorPkgCommand : ExecutorCommand
{
    public string Nome { get; private set; }
    public bool RecordSet { get; private set; }

    public ExecutorPkgCommand(string nome, bool rs = false)
    {
        Nome = nome;
        RecordSet = rs;
    }
}
```

“ O uso ou não do record set será informado pela equipe, por ser um parametro opcional, se não for informado no construtor, seu valor sera "false"

o Executor, tem uma propriedade chamada "Parametros", de tipo "List< object >", que deve ser mapeada da seguinte maneira:

```
Parametros = new List<object>
{
    new ParametrosExecutor<int?>()
    {
        Nome = "CODIGO",
        Tipo = ParameterDirection.Input,
        TipoDado = OracleDbType.Int32,
        Valor = null
    },
    new ParametrosExecutor<int>()
    {
        Nome = "Atividade",
        Tipo = ParameterDirection.Input,
        TipoDado = OracleDbType.Int32,
        Valor = id
    }
}
```

“ Deverá ser informado na instancia da classe o tipo de dado do parametro. Os parametros também devem estar na mesma ordem da package no banco de dados

## Métodos de Execução

- ExecuteVoid Esse método não tem nenhum tipo de retorno, apenas executa a pkg.
- GetList<T> Esse método retorna uma lista do objeto do tipo informado no parametro T
- GetNextSeq Esse método, é utilizado para obter a proxima sequence do modulo desejado, retornando um valor do tipo "int"

“ Todos os métodos devem receber a variavel global como parametro

## Functions

As functions funcionam de forma parecida das packages, as diferenças são:

- Variável de Retorno:

Em sua instancia, deverá ser informada na variavel "Retorno" o tipo de dado oracle que a function retorna:

```
var executor = new ExecutorFncCommand("PKG_PARAMETROS_STM.Fnc_Parametro_Alfa")
{
    Retorno = OracleDbType.Varchar2
}
```

- Parametros Devem ser informados da mesma forma que nas packages.
- Retorno de Execução Deverá ser informado no parametro de tipo T qual o tipo do retorno da function, exemplo:

```
executor.Executar<string>
```

“ O método deve receber a variavel global como parametro

Revisão #6

Criado 3 abril 2025 14:14:56 por Gustavo Marques

Atualizado 4 abril 2025 12:37:17 por Gustavo Marques