**GYMSoftware**

**Padrão para Nomenclatura de Banco de Dados**

Versão <1.0>

**Histórico**

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versão | Descrição | Autor |
| 08/07/2013 | 1.0 | Versão inicial. | Lúcio |
|  |  |  |  |
|  |  |  |  |

[1 Introdução 4](#_Toc203191522)

[2 BANCO DE DADOS 4](#_Toc203191523)

[2.1 Regra Geral 4](#_Toc203191524)

[2.2 Sintaxe 4](#_Toc203191525)

[2.3 Exemplos 4](#_Toc203191526)

[3 TABELAS 5](#_Toc203191527)

[3.1 Regra Geral 5](#_Toc203191528)

[3.2 Sintaxe 5](#_Toc203191529)

[3.3 Exemplos 5](#_Toc203191530)

[4 ATRIBUTOS 5](#_Toc203191531)

[4.1 Regra Geral 5](#_Toc203191532)

[4.2 Sintaxe 5](#_Toc203191533)

[4.3 Exemplos 6](#_Toc203191534)

[5 VIEWS 6](#_Toc203191535)

[5.1 Regra Geral 6](#_Toc203191536)

[5.2 Sintaxe 6](#_Toc203191537)

[5.3 Exemplos 6](#_Toc203191538)

[6 ÍNDICES 7](#_Toc203191539)

[6.1 Regra Geral 7](#_Toc203191540)

[6.2 Sintaxe 7](#_Toc203191541)

[6.3 Exemplos 7](#_Toc203191542)

[7 STORED PROCEDURES 7](#_Toc203191543)

[7.1 Regra Geral 7](#_Toc203191544)

[7.2 Sintaxe 7](#_Toc203191545)

[7.3 Exemplos 8](#_Toc203191546)

[8 TRIGGER 8](#_Toc203191547)

[8.1 Regra Geral 8](#_Toc203191548)

[8.2 Sintaxe 8](#_Toc203191549)

[8.3 Exemplos 9](#_Toc203191550)

[9 CHECK 9](#_Toc203191551)

[9.1 Regra Geral 9](#_Toc203191552)

[9.2 Sintaxe 9](#_Toc203191553)

[9.3 Exemplos 9](#_Toc203191554)

[10 SEQUENCE 9](#_Toc203191555)

[10.1 Regra Geral 9](#_Toc203191556)

[10.2 Sintaxe 9](#_Toc203191557)

[10.3 Exemplos 10](#_Toc203191558)

1. Introdução

O maior esforço no desenvolvimento de software é dispensado às atividades de manutenção, desta forma, quanto mais fácil for o entendimento do código do sistema, mais produtiva será a equipe de desenvolvimento. Freqüentemente as pessoas que escrevem o código não são as mesmas que o mantém e, quando são, geram uma dependência com o código desenvolvido que dificilmente é dissolvida. Um padrão de codificação/nomenclatura visa minimizar esses problemas, pois estabelece regras, definindo como o código deve ser escrito para favorecer a impessoalidade do artefato.

Este documento tem como objetivo definir um padrão de nomenclatura para definição de bancos de dados que, quando usado, garante um melhor entendimento por qualquer pessoa que conheça e siga o mesmo.

1. BANCO DE DADOS
   1. Regra Geral

O nome do banco de dados deverá identificar o negócio que está sendo automatizado ou deverá refletir a sigla da aplicação.

* 1. Sintaxe

{[A..Z][a..z]} → Xxxxx, onde:

Xxxxx – indica o nome do banco de dados.

* Primeira letra deverá ser maiúscula e as demais minúsculas. Para cada palavra interna, primeira letra em maiúsculo.
* Usar palavras no singular, sem acentuações ou caracteres especiais.
  1. Exemplos

Banco

Financeiro

Caixa

ContasPagar

SAP

1. TABELAS
   1. Regra Geral

O nome das tabelas deve ser sugestivo e refletir os dados armazenados nesta. O nome deve ser pré-fixado pela palavra TBL.

* 1. Sintaxe

TBL\_{[A..Z]} → TBL\_XXXXX..., onde:

XXXXX – indica o nome da tabela no banco de dados.

* Todas as letras deverão ser maiúsculas. Para cada palavra interna, utilizar um *underline* antes da palavra.
* Não usar preposições.
* Limite-se a usar 30 caracteres para definição do nome da tabela – padrão da maioria dos bancos de dados.
* Usar palavras no singular, sem acentuações ou caracteres especiais.
* Evite usar abreviações, se necessário use as conhecidas.
  1. Exemplos

TBL\_PESSOA – Tabela de Pessoas

TBL\_FUNCIONARIO – Tabela de Funcionários

TBL\_COMPRA\_IMOVEL – Tabela de Compra de Imóveis

TBL\_CARTAO\_CREDITO – Tabela de Cartões de Crédito

1. ATRIBUTOS
   1. Regra Geral

O nome dos atributos devem identificar a coluna de maneira clara e descritiva.

* 1. Sintaxe

{[A..Z][a..z]} → Xxxxx, onde:

Xxxxx – indica o nome do atributos de uma determinada tabela do banco de dados.

* Primeira letra deverá ser maiúscula e as demais minúsculas. Para cada palavra interna, primeira letra em maiúsculo.
* Usar palavras no singular, sem acentuações ou caracteres especiais.
* Não usar preposições.
* Evite usar abreviações, se necessário use as conhecidas. Abreviações limitadas a 4 caracteres.
  1. Exemplos

*Sem Abreviações*

Nome

NomeFuncionario

DataNascimento

ValorSalario

*Com Abreviações*

CodFuncionario

DescProduto

NomeFuncResp

DtNascimento

1. VIEWS
   1. Regra Geral

Deve-se usar a mesma semântica utilizadas para as tabelas. O nome deve ser pré-fixado pela palavra VW.

* 1. Sintaxe

VW\_{[A..Z]} → VW\_XXXXX, onde:

XXXXX – indica o nome da view no banco de dados.

* Todas as letras deverão ser maiúsculas. Para cada palavra interna, utilizar um *underline* antes da palavra.
* Não usar preposições.
* Limite-se a usar 30 caracteres para definição do nome da View.
* Usar palavras no singular, sem acentuações ou caracteres especiais.
* Evite usar abreviações, se necessário use as conhecidas.
  1. Exemplos

VW\_PESSOA – Visão da Tabela de Pessoas

VW\_FUNCIONARIO – Visão da Tabela de Funcionários

VW\_IMOVEL\_VENDIDO\_ANO – Visão de uma consulta dos imóveis vendidos por ano

1. ÍNDICES
   1. Regra Geral

Deve-se usar a mesma semântica utilizadas para os atributos. O nome deve ser pré-fixado pela palavra IDX.

* 1. Sintaxe

{[A..Z][a..z]} → Xxxxx, onde:

Xxxxx - indica o nome do índice no banco de dados.

* Primeira letra deverá ser maiúscula e as demais minúsculas. Para cada palavra interna, primeira letra em maiúsculo.
* Usar palavras no singular, sem acentuações ou caracteres especiais.
* Não usar preposições.
* Evite usar abreviações, se necessário use as conhecidas. Abreviações limitadas a 4 caracteres.
  1. Exemplos

IDX\_Codigo – Índice da Tabela Pessoa pelo atributo Código.

IDX\_Nome – Índice da Tabela Funcionário pelo atributo Nome.

IDX\_MatriculaImovel – Índice da Tabela Imóvel pelo atributo Matrícula Imóvel.

1. STORED PROCEDURES
   1. Regra Geral

Deve-se usar a mesma semântica utilizadas para as tabelas. O nome deve ser pré-fixado pela palavra SP seguido por uma sigla para indicar a operação principal realizada.

* 1. Sintaxe

STP\_{[A..Z]}\_{[A..Z]} → STP\_XXX\_XXXXX, onde:

XXX – identifica a ação principal da stored procedure: Inserção, Exclusão, atualização e consulta a dados.

XXXXX – indica o nome da Stored Procedure no banco de dados.

* Todas as letras deverão ser maiúsculas. Para cada palavra interna, utilizar um *underline* antes da palavra.
* Não usar preposições.
* Usar palavras no singular, sem acentuações ou caracteres especiais.
* Evite usar abreviações, se necessário use as conhecidas.
* Endentar de modo a tornar a codificação clara e facilitar o trabalho de manutenção. Quando disponível utilizar ferramentas de endentação automática do banco de dados.
* Inserir comentários sempre que necessário. Comentários que descrevem procedimentos óbvios devem ser evitados para evitar poluição do código desenvolvido.
* Evitar aninhamento excessivo de comandos, o que dificulta a manutenção do código.
  1. Exemplos

SP\_INSERE\_PESSOA – Stored Procedure para atualização de dados de Pessoa.

SP\_ATUALIZA\_SALDOCLIENTE – Stored Procedure para atualização de saldo do Cliente.

SP\_CONSULTA\_HISTORICOFUNC – Stored Procedure para consulta do histórico do Funcionário.

1. TRIGGER
   1. Regra Geral

Deve-se usar a mesma semântica utilizadas para as tabelas. O nome deve ser pré-fixado pela palavra TRG seguido por uma sigla para indicar a operação principal realizada.

* 1. Sintaxe

TRG\_{[A..Z]}\_{[A..Z]} → TRG\_XXX\_XXXXX, onde:

XXX – identifica a ação principal da trigger: Inserção, Exclusão, atualização e consulta a dados.

XXXXX – indica o nome da trigger no banco de dados.

* Todas as letras deverão ser maiúsculas. Para cada palavra interna, utilizar um *underline* antes da palavra.
* Não usar preposições.
* Usar palavras no singular, sem acentuações ou caracteres especiais.
* Evite usar abreviações, se necessário use as conhecidas.
* Endentar de modo a tornar a codificação clara e facilitar o trabalho de manutenção. Quando disponível utilizar ferramentas de endentação automática do banco de dados.
* Inserir comentários sempre que necessário. Comentários que descrevem procedimentos óbvios devem ser evitados para evitar poluição do código desenvolvido.
* Evitar aninhamento excessivo de comandos, o que dificulta a manutenção do código.
  1. Exemplos

TRG\_REMOVE\_PESSOA – Trigger para excluir dados de Pessoa.

TRG\_ATUALIZA\_SALDOCLIENTE – Trigger para atualização de saldo do Cliente.

TRG\_CONSULTA\_HISTORICOFUNC – Trigger para consultar o histórico do Funcionário.

1. CHECK
   1. Regra Geral

Especifica que uma condição deve ser atendida por cada tupla da tabela.

* 1. Sintaxe

CHK \_{[A..Z]}\_{[A..Z]} → CHK \_XXX\_XXXXX, onde:

XXX\_XXXXX – indica o nome da tabela seguido pelo nome da coluna onde haverá o check.

* Todas as letras deverão ser maiúsculas. Para cada palavra interna, utilizar um *underline* antes da palavra.
* Não usar preposições.
* Usar palavras no singular, sem acentuações ou caracteres especiais.
* Evite usar abreviações, se necessário use as conhecidas.
  1. Exemplos

CHK\_PESSOA\_CPF – Check no campo CPF da tabela Pessoa.

CHK\_CLIENTE\_SEXO – Check no campo sexo da tabela Cliente.

1. SEQUENCE
   1. Regra Geral

Uma sequence é um objeto do banco de dados criado pelo usuário, que pode ser compartilhado por vários usuários para gerar números seqüenciais inteiros exclusivos. Normalmente, as sequences são utilizadas para criar um valor de PRIMARY KEY, que deve ser exclusivo para cada linha. Os números da sequence são armazenados e gerados de modo independente das tabelas. Portanto, a mesma sequence pode ser usada para várias tabelas.

* 1. Sintaxe

SQC\_{[A..Z]} → SQC\_XXXXX, onde:

XXXXX – indica o nome do campo que utilizará a Sequence.

* Todas as letras deverão ser maiúsculas. Para cada palavra interna, utilizar um *underline* antes da palavra.
* Não usar preposições.
* Usar palavras no singular, sem acentuações ou caracteres especiais.
* Evite usar abreviações, se necessário use as conhecidas.
  1. Exemplos

SQC\_INSCRICAO – Sequence para o campo inscrição

SQC\_SOLICITACAO – Sequence para o campo solicitação.