

Guia de Projeto de Banco de Dados

SUMÁRIO

1.INTRODUÇÃO	2
2.GUIA DE PROJETO DE BANCO DE DADOS	2
2.1.Padrões de Nomenclatura	2
2.1.1.Identificadores de objeto	2
2.1.2.Identificadores de coluna	2
2.1.3.Nomeação de Arquivos	3
2.2.Definindo a solução	4
2.2.1.Evitar duplicidade de tabelas e ou campos	4
2.2.2.Definir chave primária para tabela	4
2.2.3.Manter consistência dos objetos.....	4
2.2.4.Elaborar e manter a documentação.....	4
2.3.Implementação da Solução	4
2.3.1.Criando Diagrama EER no MySQL Workbench	4
2.3.2.Gerando script SQL no MySQL Workbench	6

1. INTRODUÇÃO

Esse guia tem como objetivo direcionar as atividades realizadas pela equipe de Projeto de Banco de Dados.

2. GUIA DE PROJETO DE BANCO DE DADOS

2.1. Padrões de Nomenclatura

2.1.1. Identificadores de objeto

Os identificadores de objeto (tabela/visão) devem obedecer às seguintes regras:

- Devem ser compostos de uma primeira parte, que identifica o tipo do objeto (tb para tabela ou vw para visão), e de uma segunda parte que deve representar o tipo de informação dos registros do objeto.
 - Ex.: tb_cliente
 - Ex.: vw_vendas_por_departamento
 - Ex.: ~~cliente~~
- Devem fazer uso de underscore (“_”) para separar nomes compostos.
 - Ex.: tb_cliente_cnpj
 - Ex.: ~~tb_cliente~~~~cnpj~~
- Devem estar no singular, e não no plural.
 - Ex.: tb_departamento
 - Ex.: ~~tb_departamentos~~
- Devem ter limite máximo de 40 caracteres.
 - Ex.: vw_vendas_por_departamento
 - Ex.: ~~vw_vendas_totais_por_departamento~~
- Devem estar em letras minúsculas.
 - Ex.: tb_consultas
 - Ex.: ~~TB_CONSULTAS~~
- Não devem possuir acentuação gráfica.
 - Ex.: tb_solicitacao
 - Ex.: ~~tb_solicitação~~
- Podem possuir um nome composto, no caso de tabelas de relacionamento.
 - Ex.: tb_funcionario_departamento ou tb_lotacao, para lotação de funcionário em departamento.
- Devem possuir uma terminação “log”, no caso de tabelas de histórico.
 - Ex.: tb_cliente_alteracoes_log, para tabela de registro de alterações no cadastro do cliente.

2.1.2. Identificadores de coluna

Os identificadores de coluna devem obedecer às seguintes regras:

- Devem ser compostos de uma primeira parte, que identifica o tipo do dado da coluna (conforme abreviação contida na Tabela 1), e de uma segunda parte que deve representar o tipo de informação da coluna.
 - Ex.: int_codigo
 - Ex.: ~~codigo~~
- Devem fazer uso de underscore (“_”) para separar nomes compostos.
 - Ex.: vchr_nome_completo
 - Ex.: ~~vchr_nome~~~~completo~~
- Devem estar no singular, e não no plural.
 - Ex.: vchr_endereco
 - Ex.: ~~vchr_enderecos~~
- Devem ter limite máximo de 30 caracteres, com exceção das chaves estrangeiras.
 - Ex.: vchr_endereco_completo
 - Ex.: ~~vchr_endereco_completo_com_estado_e_pais~~
- Devem estar em letras minúsculas.
 - Ex.: dt_nascimento
 - Ex.: ~~DT_NASCIMENTO~~
- Não devem possuir acentuação gráfica.
 - Ex.: ts_emprestimo
 - Ex.: ~~ts_empréstimo~~

- No caso de a coluna ser chave primária (PRIMARY KEY), o nome deve ter o prefixo “pk”.
 - Ex.: pk_int_codigo
 - Ex.: int_codigo
- No caso de a coluna ser chave estrangeira (FOREIGN KEY), o nome deve ser composto pelo prefixo “fk”, pelo tipo de dado da coluna (conforme Tabela 1), pelo nome da tabela referenciada e pelo nome da coluna na tabela referenciada sem o seu tipo de dado.
 - Ex.: fk_int_tb_departamento_codigo
 - Ex.: fk_int_tb_departamento_int_codigo
 - Ex.: fk_tb_departamento_int_codigo
 - Ex.: fk_tb_departamento_codigo
- Opcionalmente, pode haver um outro nome para diferenciar os nomes de duas chaves estrangeiras para a mesma tabela.
 - Ex.: fk_int_id_mandante_tb_time e fk_int_id_visitante_tb_time, para uma chave estrangeira para o time mandante de um jogo e uma chave estrangeira para o time visitante do mesmo jogo.

Tabela 1 – Abreviações de tipos de dado de coluna.

Abreviação	Tipo do dado da coluna	Exemplo
int	INTEGER	int_codigo
sint	SMALLINT	sint_idade
bint	BIGINT	bint_calculo
srl	SERIAL	srl_id
bsrl	BIGSERIAL	bsrl_id
real	REAL	real_fator
dbl	DOUBLE PRECISION	dbl_indice
num	DECIMAL/NUMERIC	num_salario
vchr	VARCHAR	vchr_nome_completo
chr	CHAR	chr_uf
txt	TEXT	txt_obs
dt	DATE	dt_nascimento
hr	TIME	hr_inicio_consulta
ts	TIMESTAMP	ts_cadastro
itv	INTERVAL	itv_intervalo
bool	BOOLEAN	bool_ativo

2.1.3. Nomeação de Arquivos

Os nomes dos arquivos devem obedecer às seguintes regras:

- Os scripts SQL devem ter extensão .sql.
- Devem ser compostos por “db”, seguido do nome do projeto e da data (ano, mês e dia) e hora (hora, minuto e segundo) de geração do script.
 - Ex.: db_loja_2017_05_10_08_50_32.sql

2.2. Definindo a solução

2.2.1. Evitar duplicidade de tabelas e ou campos

- Antes de criar uma tabela, procurar por tabela já existente.
- Antes de criar uma coluna, verificar se uma coluna com a informação já existe e utilizá-la (se for o caso).

2.2.2. Definir chave primária para tabela

- Cada tabela deve ter uma chave primária.

- A chave primária de uma tabela pode ser composta por mais de uma coluna.

2.2.3. Manter consistência dos objetos

- Avaliar os impactos da criação de objetos e/ou colunas no sistema como um todo.
- Não excluir objetos nem colunas sem prévia autorização do responsável pelo projeto ou por pessoa designada para tal.
- Seguir o padrão de nomenclatura para tabelas e colunas, contido neste documento.

2.2.4. Elaborar e manter a documentação

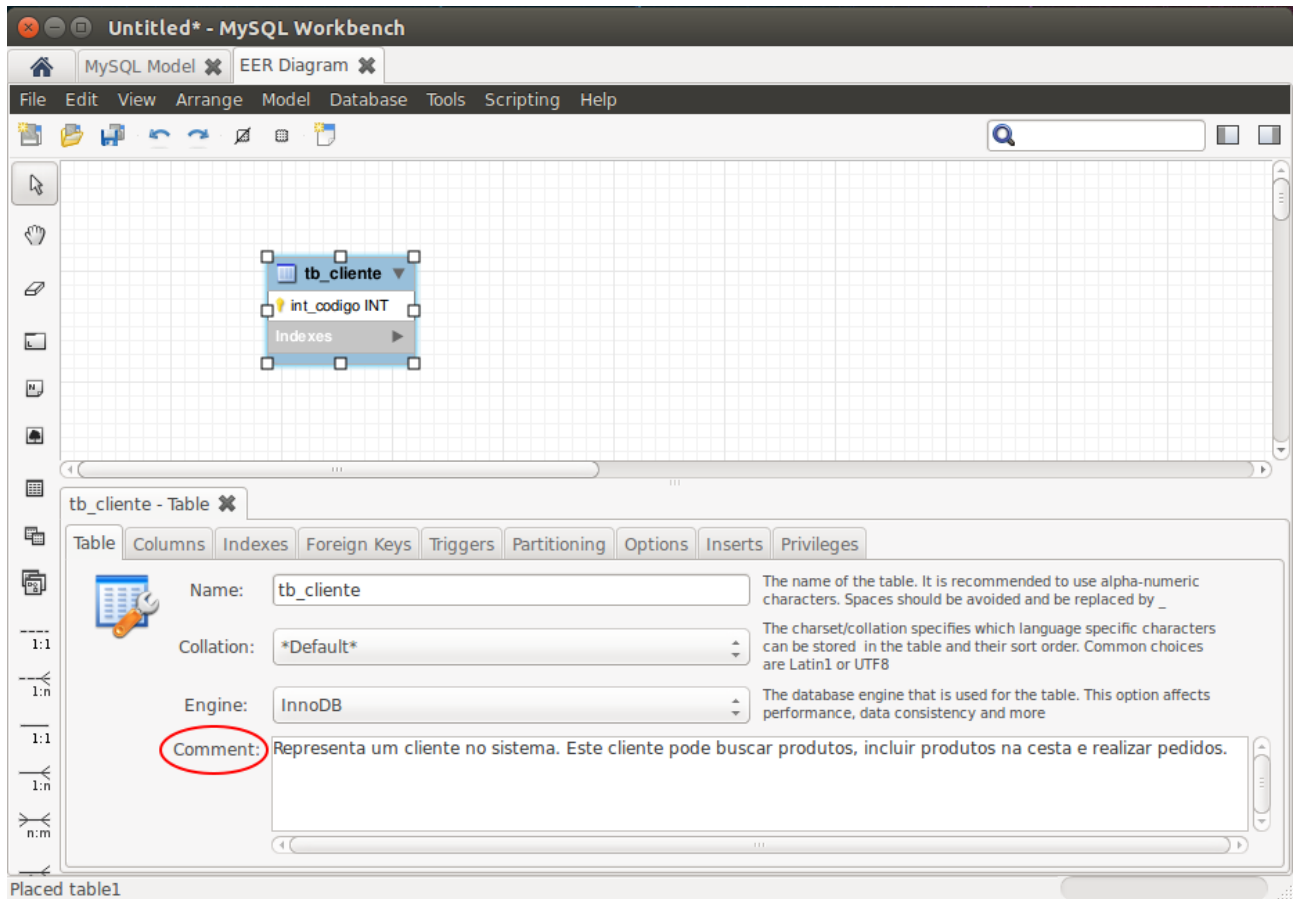
- Ao criar um objeto (tabela/visão), documentar com pelo menos as seguintes informações: breve descrição do objeto.
- Ao criar uma coluna em um objeto, documentar com pelo menos as seguintes informações: breve descrição da coluna, domínio dos dados da coluna (por exemplo, intervalo de valores possíveis) e exemplos de valores válidos para a coluna (pelo menos 3).
 - Ex.: Coluna int_idade na tabela tb_cliente
 - Breve descrição: representa a idade do cliente.
 - Domínio: inteiros não negativos.
 - Exemplos: 0 (idade indeterminada), 1, 2, 90.
- Ao fazer qualquer alteração no projeto do banco de dados, verificar e manter a documentação do objeto e/ou coluna afetada.

2.3. Implementação da Solução

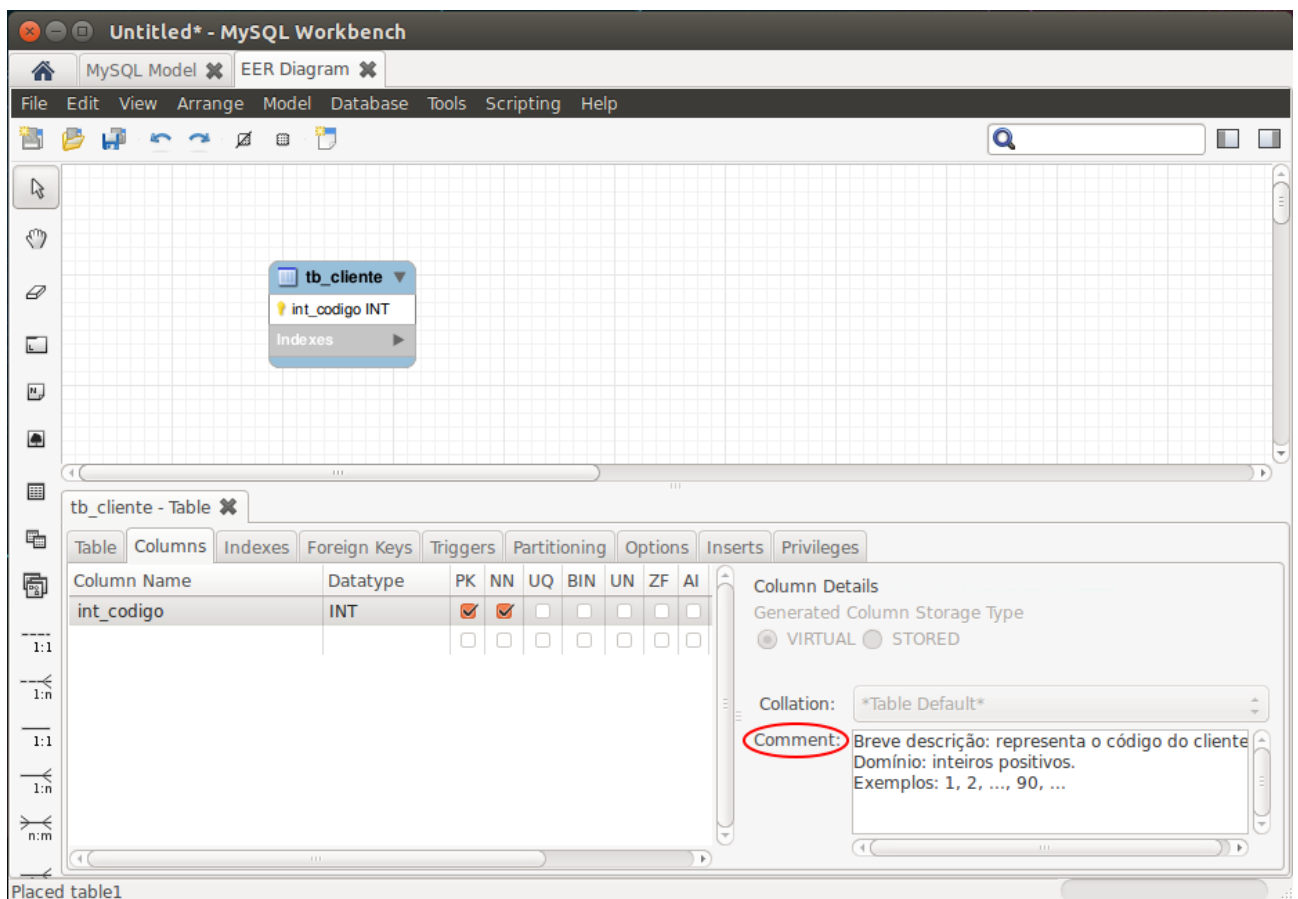
Uma vez definida a solução, seguem os passos para implementá-la.

2.3.1. Criando Diagrama EER no MySQL Workbench

- Executar o MySQL Workbench (versão 6.3).
- Criar um modelo, selecionando o menu File > New Model.
- Criar um diagrama, selecionando o menu File > Model > Add Diagram.
- As tabelas devem ser documentadas no campo Comment, conforme figura a seguir.



- As colunas devem ser documentadas no campo Comment da coluna, conforme figura a seguir.



2.3.2. Gerando script SQL no MySQL Workbench

- Exportar o modelo, selecionando o menu File > Export > Forward Engineer SQL CREATE Script...
- Em Filter Objects, selecionar as opções Export MySQL Table Objects e Export MySQL View Objects.