

Modelagem de banco de dados com MySQL Workbench



Após o termino desse curso, você estará apto à:

Teoria:

- Reconhecer os principais tipos de dados suportados pelo MySQL 5.0
- Otimizar espaço pela escolha certo do tipo de dados.
- Entender relacionamentos 1:1, 1:n, n:n.

Prática:

- Modelar um banco usando Mysql WorkBench.
- Criar relacionamentos entre as tabelas.
- Exportar código SQL para gerar o banco.

Apresentação



Leonardo Chaves de Freitas - leonardo@phprime.com.br

Profissional MYSQL/PHP há 4 anos

Profissional Postgres-SQL há 2 anos

Bacharel em Ciência da Computação pela UFMG

- Instrutor dos cursos MySQL e PHP Essentials da PHPrime
- Sócio-Diretor da UaiTI

Tipo de dados



O que são os tipos de dados???

Tipo de dados



O MySQL permite o armazenamento em diversos tipos de dados. É importante conhecê-los para conseguirmos criar nossas tabelas de forma correta e otimizada para o tipo de dados que ela armazenará.

Podemos agrupar os tipos da seguinte forma:

- Numéricos
- Strings (textos)
- Temporais

Espaço de armazenamento



O que é esse espaço de armazenamento?

Espaço em disco que cada valor de um determinado tipo irá ocupar.

- O somatório de bytes de todas as colunas de um registro (tupla ou linha) pode nos auxiliar na projeção de crescimento do nosso banco.

Ex: Usar tipo INT para campo de usuário ativo (1 – ativo , 0 – inativo) - Desperdício de espaço.

Tipos Numéricos



Para a escolha de um tipo numérico devemos ter em mente:

- O intervalo de valores que o tipo de dados representa.
- A quantidade de espaço de armazenamento que os valores das colunas exigem.
- O número máximo de caracteres apresentados no retorno de uma consulta.
- A coluna de precisão e escala de ponto flutuante e ponto fixo de valores

Inteiros



Tipo	Armazenamento requerido	Intervalo	Intervalo sem sinal
TINYINT	1 byte	-128 to 127	0 to 255
SMALLINT	2 bytes	-32,768 to 32,767	0 to 65,535
MEDIUMINT	3 bytes	-8,388,608 to 8,388,607	0 to 16,777,215
INT	4 bytes	-2,147,683,648 to 2,147,483,647	0 to 4,294,967,295
BIGINT	8 bytes	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807	0 to 18,446,744,073,709,551,615

Recapitulando



Quantidade de caracteres

Sem sinal

```
CREATE TABLE `vendedores` (  
  `id`          mediumint(8) unsigned NOT NULL auto_increment,  
  `nome`       varchar(200) default NULL,  
  `celular_empresa` varchar(10) default NULL,  
  `celular_pessoal` varchar(10) default NULL,  
  `telefone`   varchar(10) default NULL,  
  `nascimento` date default NULL,  
  `observacoes` text,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Ponto Flutuante



- **FLOAT[(M,D)]**

Um número de ponto flutuante pequeno (precisão simples).

- **DOUBLE[(M,D)]**

Um número de ponto flutuante de tamanho normal (dupla-precisão).

O **M** é a largura do display e o **D** é número de casa decimais

peso **FLOAT(10,3)**

valor_medio **DOUBLE(20,7)**

Decimal



•DECIMAL[(M[,D])]

Decimal é usado para valores exatos que possuem um parte inteira, uma fracionada ou ambas;

O **M** é a largura do display e o **D** é número de casa decimais

Bit



bit_coluna1 BIT(4)
bit_coluna2 BIT(20)

Strings



- CHAR
- VARCHAR
- TEXT
- BINARY
- VARBINARY
- BLOB
- ENUM
- SET

CHAR, VARCHAR, TEXT



PHP MG

Tipo	Armazenamento	Tamanho máximo
CHAR(M)	M caracteres	255 caracteres
VARCHAR(M)	L caracteres + 1 ou 2 bytes	65,535 caracteres
TINYTEXT	L caracteres + 1 byte	255 caracteres
TEXT	L caracteres + 2 bytes	65,535 caracteres
MEDIUMTEXT	L caracteres + 3 bytes	16,777,215 caracteres
LONGTEXT	L caracteres + 4 bytes	4,294,967,295 caracteres

M representa o tamanho máximo da coluna.

L representa o tamanho atual do valor.

CHAR



CHAR(M)

Uma string de **tamanho fixo** que é sempre preenchida a direita com espaços até o tamanho especificado quando armazenado.

VARCHAR



VARCHAR(M)

Uma string de tamanho variável.

TEXT



Usados para textos maiores e de tamanhos variáveis.

- TINYTEXT
- TEXT
- MEDIUMTEXT
- LONGTEXT

BINARY, VARBINARY, BLOB



BINARY, VARBINARY, BLOB são tipos de dados para armazenamento de strings binárias e é equivalente aos tipos não binários CHAR, VARCHAR, e TEXT.

ENUM e SET



ENUM('valor1','valor2',...):

Uma enumeração. Uma string que só pode ter um valor, selecionado da lista de valores 'valor1', 'valor2'.

SET('valor1','valor2',...)

Um conjunto. Um objeto string que pode ter zero ou mais valores, cada um deve ser selecionado da lista de valores 'valor1', 'valor2',

Temporais



- **DATE:**

Uma data. A faixa suportada é entre '1000-01-01' e '9999-12-31'.
Formato 'AAAA-MM-DD'

- **TIME:**

Uma hora. A faixa é entre '-838:59:59' e '838:59:59'. Formato
'HH:MM:SS'

- **DATETIME:**

Um combinação de hora e data. A faixa suportada é entre '1000-01-01
00:00:00' e '9999-12-31 23:59:59'.

- **YEAR[(2|4)]:**

Um ano no formato de 2 ou 4 dígitos (padrão são 4 dígitos)

Temporais



- **TIMESTAMP:**

Similar ao datetime com algumas particularidades.

Formato: 'AAAA-MM-DD HH:MM:SS'.

- Padrão é NOT NULL.
- No insert e update ele é atualizado automaticamente.



MySQL Workbench



- Ferramenta de design
- Veio substituir o DBDesigner4
- Link para Download

<http://dev.mysql.com/downloads/workbench/5.1.html>



Quê mais?

Mysql Model

MySQL Model EER Diagram

EER Diagrams

PHYSICAL Add Diagram PHYSICAL EER Diagram

Physical Schemata

sakila
MySQL Schema

Tables (16 items)

Add Table actor address category city
country customer film film_actor film_category
film_text inventory language payment rental
staff store

Views (7 items)

Add View actor_info customer_list film_list nicer_but_slower_fil...
sales_by_film_category sales_by_store staff_list

Routines (6 items)

Add Routine film_in_stock film_not_in_stock get_customer_balance inventory_held_by_c...
inventory_in_stock rewards_report

Routine Groups (2 items)

customer

Name: The name of the table. It is recommended to use only alpha-numeric characters. Spaces should be avoided and be replaced by _

Collation: The charset/collation specifies which language specific characters can be stored in the table and their sort order. Common choices are Latin1 or UTF8.

Engine: The database engine that is used for the table. This option affects performance, data consistency and much more.

Comments:
Basic information about the customer like first and last name are stored in the table itself. Same for the date the record was created and when the information was last updated.

Table Columns Indexes Foreign Keys Triggers Partitioning Options Inserts Privileges

Catalog

sakila
Tables
actor
address
category
city
country
customer
film
film_actor
film_category
film_text
inventory
language
payment

Description

customer: MySQL Table

Table storing all customers. Holds foreign keys to the address table and the store table where this customer is registered.

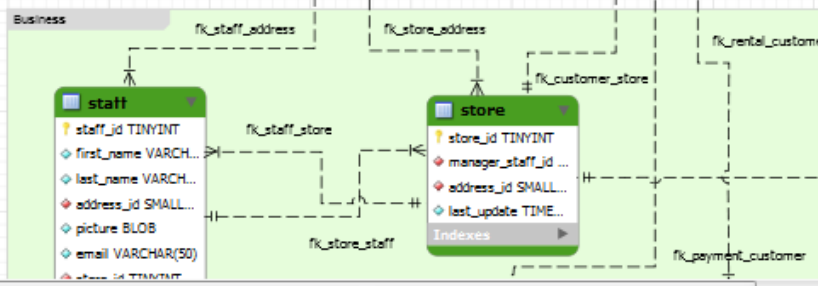
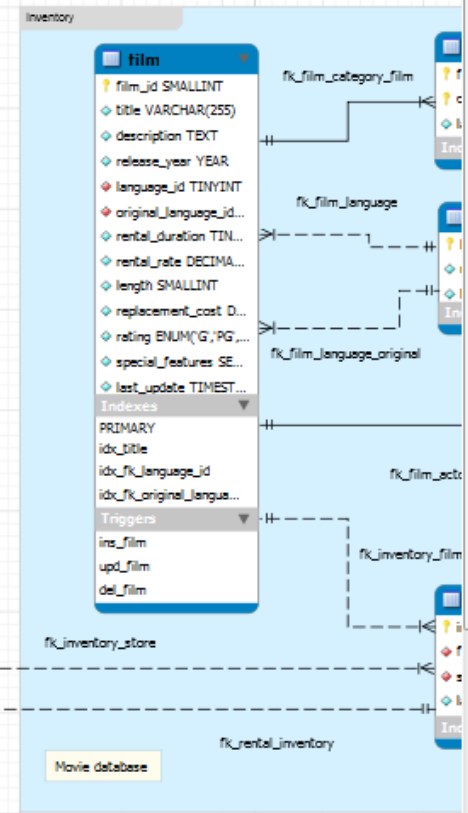
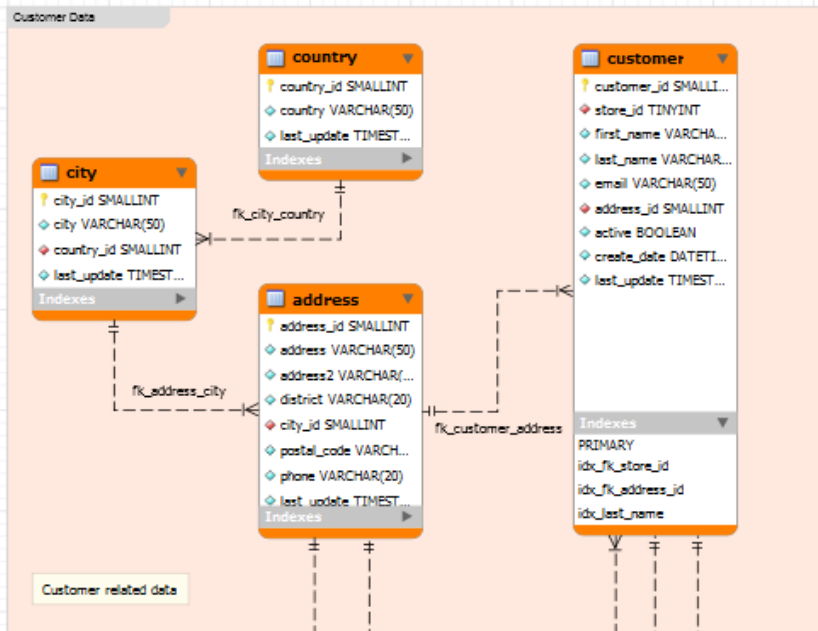
Basic information about the customer like first and last name are stored in the table itself. Same for the date the record was created and when the information was last updated.

Descripi... Properties History

Press [Ctrl]+[Tab] to switch to the next tab.



MySQL Model EER Diagram



- Catalog**
- sakila
 - Tables
 - actor
 - address
 - category
 - city
 - country
 - customer
 - film
 - film_actor
 - film_category
 - film_text
 - inventory
 - language
 - payment

Description

No Selection

Ferramenta de modelagem

The screenshot shows the MySQL Workbench EER Diagram interface. Two tables are visible: **product** and **product_part**.

- product** table columns: idproduct (INT), name (VARCHAR(45)), ean (CHAR(16)).
- product_part** table columns: idproduct_part (INT), idproduct (INT), weight (DECIMAL(8,4)).

A dialog box titled "Foreign Key Columns" is open, showing "idproduct INT" and a "Pick Referenced Columns" button. The catalog pane on the right shows the database structure: mydb > Tables > product, product_part.

The bottom pane shows the column details for the selected column "idproduct":

Column Name	Datatype	NN	AI	Default
idproduct	INT	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
name	VARCHAR(45)	<input type="checkbox"/>	<input type="checkbox"/>	
ean	CHAR(16)	<input type="checkbox"/>	<input type="checkbox"/>	

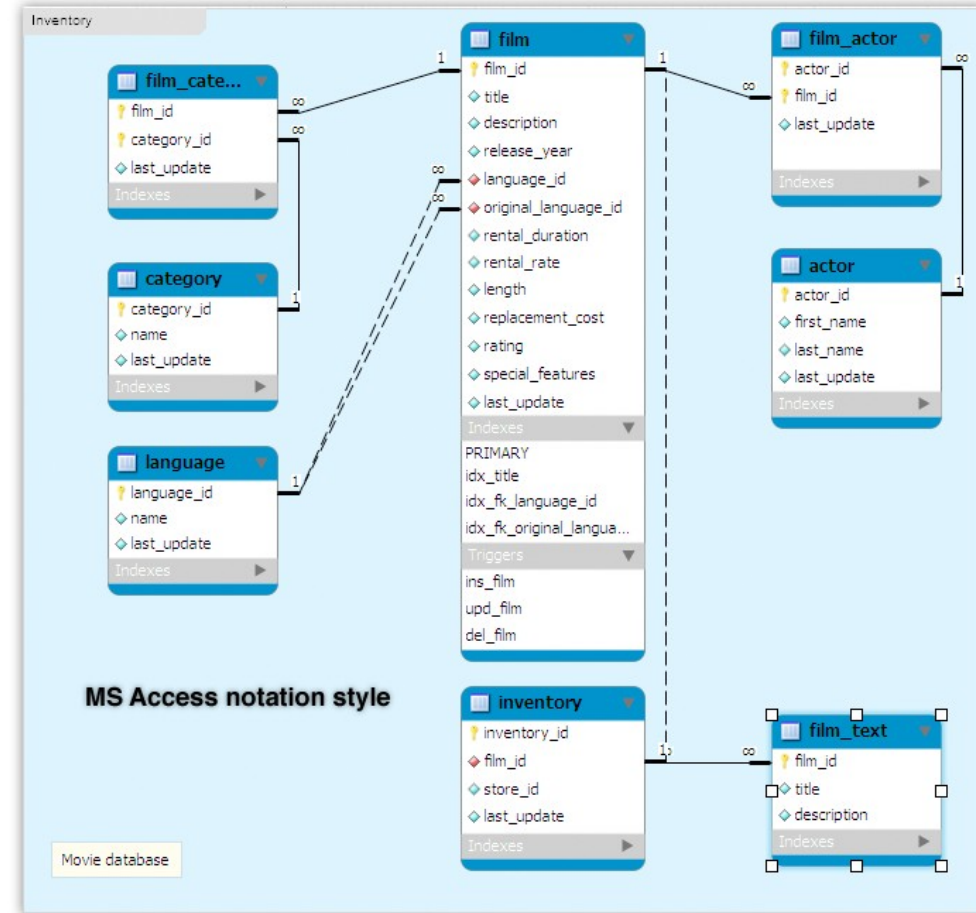
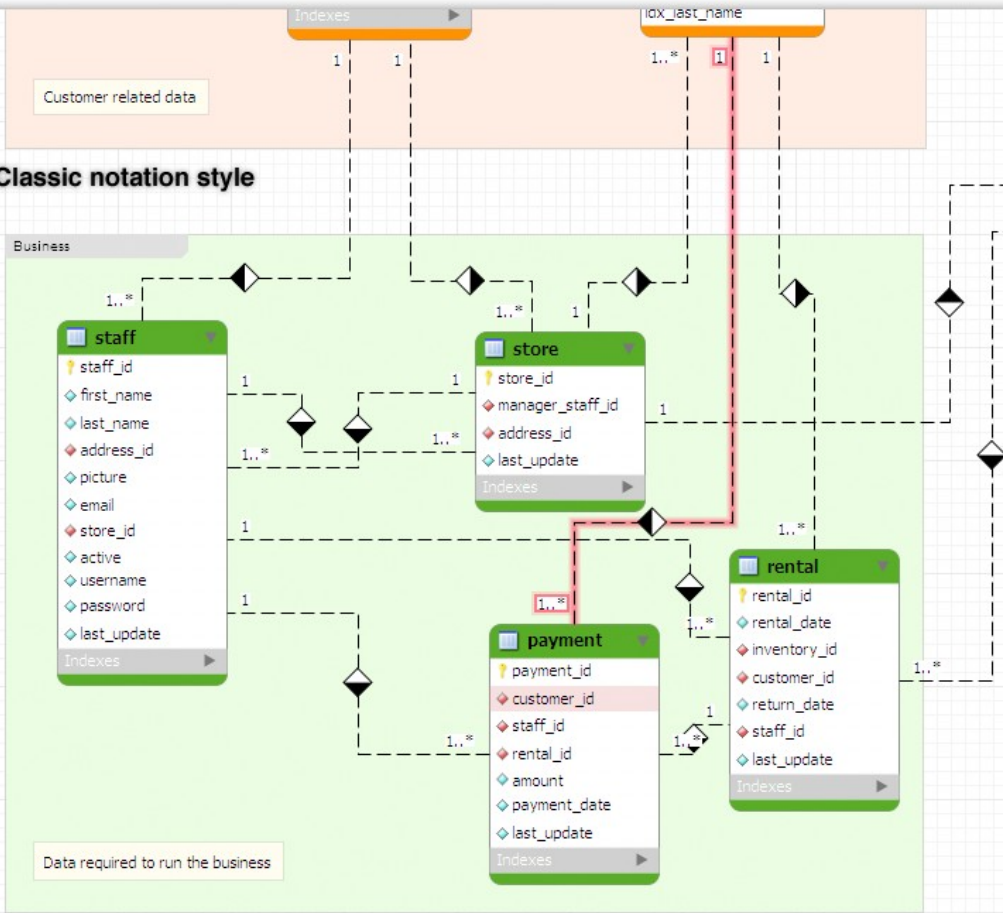
Column Details: Collation: Table Default, Flags: PRIMARY KEY

Column Comment: [Empty text area]

Bottom status bar: Column 'idproduct' selected.

Notações

Classic notation style



Vamos brincar um pouco

