

Modelagem de Dados e Linguagem SQL Descomplicadas usando o MySQL



<https://www.youtube.com/c/ProgramarIsCool>

Linguagem de Definição de Dados DDL

Introdução

Structured Query Language - SQL

- ❑ No início dos anos 70, o trabalho produtivo do colega de pesquisa da IBM E. F. Codd levou ao desenvolvimento de um produto modelo de dado relacional chamado SEQUEL ou Linguagem de Consulta em Inglês Estruturado (em inglês: Structured English Query Language).
- ❑ SEQUEL ultimamente se transformou em SQL ou Linguagem de Consulta Estruturada (em inglês: Structured Query Language).

Structured Query Language - SQL

- ❑ IBM, junto com outros fornecedores de banco de dados relacionais, queria um método padronizado para acessar e manipular dados em um banco de dados relacional.
- ❑ Através do tempo, SQL se provou popular o suficiente no mercado de trabalho para atrair a atenção do American National Standards Institute (ANSI), que lançou padrões para SQL em 1986, 1989, 1992, 1999, 2003 e 2011.

Tipos de Dados

- ❑ Tipo de dado é a característica da coluna da tabela do banco de dados que determina o tipo de dado que pode ser armazenado.
 - ❑ Alfanumérico;
 - ❑ Numérico;
 - ❑ Data;
 - ❑ Lógica.

Tipos de Dados do MySQL

❑ Tipo alfanumérico:

- ❑ **CHAR (tamanho):** armazena uma quantidade fixa de caracteres. Pode conter letras, números e caracteres especiais. Armazena até 255 caracteres.
- ❑ **VARCHAR(tamanho):** possui o mesmo conteúdo do tipo CHAR. O armazenamento é dinâmico e pode armazenar até 255 caracteres.
- ❑ **TEXT:** armazena até 65.535 caracteres.

Tipos de Dados do MySQL

□ Tipo data:

- **DATE:** o valor referente a uma data no formato 'CCYY-MM-DD'. Por exemplo 1985-11-25 (ano-mês-dia). O 'CC' se refere aos dois dígitos do século (Century, em inglês);
- **TIME:** um valor horário no formato 'hh:mm:ss' (hora:minutos:segundos);
- **TIMESTAMP:** timestamp é uma sequência de caracteres ou informação codificada que identifica uma marca temporal ou um dado momento em que um evento ocorreu. No MySQL, ele tem o formato 'CCYY-MM-DD hh:mm:ss' – neste caso, seguem a padronização ISO 8601;
- **YEAR:** armazena um ano no formato 'CCYY' ou 'YY'.

Tipos de Dados do MySQL

□ Tipo numérico:

Tipo da coluna	Tamanho exigido
TINYINT	1 byte
SMALLINT	2 bytes
MEDIUMINT	3 bytes
INT	4 bytes
INTEGER	4 bytes
BIGINT	8 bytes
FLOAT (X)	4 se $X \leq 24$ ou 8 se $25 \leq X \leq 53$
FLOAT	4 bytes
DOUBLE	8 bytes
DOUBLE PRECISION	8 bytes
REAL	8 bytes
DECIMAL (M,D)	$M+2$ bytes se $D > 0$, $M+1$ bytes se $D = 0$ ($D+2$, se $M < D$)
NUMERIC (M,D)	$M+2$ bytes se $D > 0$, $M+1$ bytes se $D = 0$ ($D+2$, se $M < D$)

Tipo	Bytes	De	Até
TINYINT	1	-128	127
SMALLINT	2	-32768	32767
MEDIUMINT	3	-8388608	8388607
INT	4	-2147483648	2147483647
BIGINT	8	-9223372036854775808	9223372036854775807

Tipos de Dados do MySQL

- ❑ Tipo Lógico (Boolean): esse tipo de variável foi projetado para armazenar os valores lógicos 'TRUE' ou 'FALSE'.
- ❑ No entanto, o MySQL não possui esse tipo de variável. Quando o tipo boolean é criado no Workbench, o MySQL, na verdade, usa uma variável do tipo TINYINT. Neste caso 'TRUE' equivale ao número '1' e 'FALSE' equivale ao número '0'.

Criando Esquemas

Criando Esquemas

- ❑ Um esquema é uma abstração para armazenar um conjunto de objetos de banco de dados que possuem um relação semântica.
- ❑ `CREATE SCHEMA meu_primeiro_esquema`

Criando Usuários

Criando Usuários

- ❑ `CREATE user Aluno01@'%' identified by 'senha@01'`
- ❑ `CREATE user Aluno02@'localhost' identified by 'senha@02'`

Criando Tabelas

Criando Tabelas

- ❑ Criação de Tabelas:

```
Create Table NOME_TABELA(  
    Col1      TIPO_COLUNA    [Not Null],  
    Col2      TIPO_COLUNA    [Not Null],  
    Col3      TIPO_COLUNA    [Not Null]  
)
```

- ❑

```
CREATE TABLE Aluno(CPF          INT(11)          NOT NULL,  
                    Nome        VARCHAR(30)        NOT NULL  UNIQUE,  
                    PRIMARY KEY (CPF));
```


Criando Constraints

Criando Constraints

- ❑ Na criação de tabelas, é possível especificar vários tipos de restrições:
 - ❑ Chave Primária: PRIMARY KEY ;
 - ❑ Chave Estrangeira: FOREIGN KEY;
 - ❑ Restrição de Unicidade: UNIQUE;
 - ❑ Restrição de Domínio: CHECK.
 - ❑ Restrição de Vazio: NOT NULL.

Criando Constraints

- ❑ CREATE TABLE Pessoa(CPF VARCHAR(11) **NOT NULL,**
 Nome VARCHAR(30) **NOT NULL UNIQUE,**
 Sexo CHAR(1) **CHECK (Sexo IN('M', 'F')),**
 PRIMARY KEY (CPF));

- ❑ CREATE TABLE Piloto(P_CPF VARCHAR(11) **NOT NULL,**
 Licenca VARCHAR(5) **NOT NULL UNIQUE,**
 PRIMARY KEY (P_CPF),
 FOREIGN KEY (P_CPF)
 REFERENCES Pessoa (CPF));

Criando Índices

Criando Índices

- São estruturas de acesso adicionais auxiliares usadas para aumentar a velocidade de recuperação de registros na resposta a certas condições de busca.

Tabela Ordenada

CPF	Nome	Sexo
111	Lia	F
444	Caio	M
777	Ana	F
999	Beto	M

Tabela Não Ordenada

CPF	Nome	Sexo
777	Ana	F
111	Lia	F
999	Beto	M
444	Caio	M

Índice Secundário

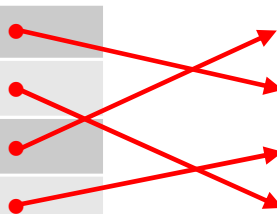
- ❑ Fornece um meio secundário de acesso a um arquivo.
- ❑ O índice secundário pode ser usado sobre um campo que é uma chave candidata e possui um valor único em cada registro, ou um campo que não é chave e possui valores duplicados.

Índices Secundários

CPF	Ponteiro
111	•
444	•
777	•
999	•

Tabela Não Ordenada

CPF	Nome	Sexo
777	Ana	F
111	Lia	F
999	Beto	M
444	Caio	M



Índice Secundário

Tabela Ordenada

222 Edu M



CPF	Nome	Sexo
111	Lia	F
444	Caio	M
777	Ana	F
999	Beto	M

Índice Secundário

Índices Secundários

CPF	Ponteiro
111	
222	
444	
777	
999	

Tabela Não Ordenada

CPF	Nome	Sexo
777	Ana	F
111	Lia	F
999	Beto	M
444	Caio	M
222	Edu	M

Criando Índice Secundário

- ❑ CREATE INDEX **indice_nome** ON **Tabela_Nome**(**Coluna_Nome**);
- ❑ CREATE INDEX **ind_aluno_cpf** ON **Aluno**(CPF);

Alterando e Excluindo Objetos do BD

Alterando Objetos do BD

- ❑ Alteração de Tabelas:
 - ❑ Incluir novas colunas em uma tabela;
 - ❑ Excluir colunas existentes em uma tabela;
 - ❑ Adicionar a definição de uma restrição em uma tabela;
 - ❑ Excluir a definição de uma restrição existente em uma tabela;
 - ❑ Modificar uma coluna.

Alterando Objetos do BD

- ❑ ALTER TABLE Aluno ADD COLUMN Telefone INT;
- ❑ ALTER TABLE Aluno ADD COLUMN Tipo_Telefone VARCHAR(5);
- ❑ ALTER TABLE Aluno MODIFY Tipo_Telefone VARCHAR(10);
- ❑ ALTER TABLE Aluno ADD CONSTRAINT UK_Telefone UNIQUE (Telefone);
- ❑ ALTER TABLE Aluno DROP CONSTRAINT UK_Telefone;
- ❑ ALTER TABLE Aluno DROP COLUMN Tipo_Telefone;

Alterando Objetos do BD

- ❑ Renomear Tabela:

- ❑ `RENAME TABLE Pessoa TO Pessoa_Nova;`

Excluindo Objetos do BD

- ❑ Índice:
 - ❑ Tabela:
 - ❑ Esquema:
-
- ❑ `DROP INDEX ind_aluno_cpf ON Aluno;`
 - ❑ `DROP TABLE Aluno;`
 - ❑ `DROP SCHEMA meu_primeiro_esquema;`

Fim