

Banco de Dados : Restrições e Relacionamentos

Prof. Márcio Funes



Plano de aula

Engines

Restrições

Relacionamentos

Chaves Estrangeiras

Exercícios

Engines

Engines

- Uma **Engine** ou **Storage Engine** é um componente de software do MySQL responsável **gravar as tabelas fisicamente**.
- A engine **InnoDB** é utilizada como padrão pelo MySQL.
- Junto ao comando create table, é possível especificar a engine utilizada.
- Cada engine suporta diferentes recursos de armazenamento, como integridade referencial e índices.

Sintaxe:

```
create table Tabela (  
    campo1      tipo,  
    campo2      tipo,  
    campo3      tipo  
)engine=NomeEngine;
```

Engines

- Uma **Engine** ou **Storage Engine** é um componente de software do MySQL responsável **gravar as tabelas fisicamente**.
- A engine **InnoDB** é utilizada como padrão pelo MySQL.
- Junto ao comando `create table`, é possível especificar a engine utilizada.
- Cada engine suporta diferentes recursos de armazenamento, como integridade referencial e índices.

```
create table Professor (  
    id int,  
    nome varchar(60)  
)engine=InnoDB;
```

```
create table Musica(  
    id int,  
    titulo varchar(40)  
)engine=CSV;
```

```
create table Produto(  
    cod int,  
    nome varchar(20)  
)engine=MEMORY;
```

Restrições

Restrições (Constraints)

- Impõe regras para inserção e alteração de dados.
- Caso uma tentativa de inserção viole uma restrição, os dados não são inseridos.
- Podem ser aplicadas a campos e tabelas. [link](#)
- Podem ser especificadas junto ao comando `create table` ou com o comando `alter table`.

Restrições (Constraints)

Restrição	Descrição
not null	Não permite valor nulo
unique	Não permite valores repetidos. O valor deve ser único
check(<i>exp</i>)	Impõe regras para os valores. (<i>exp</i>) é uma expressão que define a regra
primary key	Especifica como identificador
foreign key	Determina que os valores dessa coluna devem corresponder a valores de uma coluna especificada como chave primária.
default	Atribui um valor padrão quando nenhum for especificado.

Outros comandos relacionados (Próprios do MySQL):

as(<i>exp</i>)	Terá seus valores gerados por <i>exp</i> . Por padrão será um campo calculado, não armazenado.
[generated always] as(<i>exp</i>)	
auto_increment	Usado para gerar automaticamente valores únicos a cada inserção.

Restrições (Constraints)

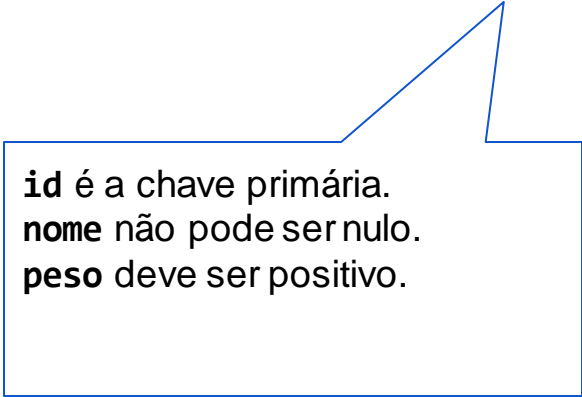
Sintaxe simplificada para restrições de atributos:

```
create table Tabela (  
    campo1      tipo      restrição,  
    campo2      tipo      restrição,  
    campo3      tipo      restrição  
);
```


Restrições (Constraints)

Exemplos:

```
create table Pets (  
    id            int            primary key      auto_increment,  
    nome          varchar(60)    not null,  
    peso          decimal(5,2)   check (peso > 0)  
)engine=InnoDB;
```



id é a chave primária.
nome não pode ser nulo.
peso deve ser positivo.

Restrições (Constraints)- MySQL

Exemplos:

```
create table Produtos (  
    id            int            primary key      auto_increment,  
    nome          varchar(60)    not null        unique,  
    preco_custo   decimal(9,2),  
    preco_venda   decimal(9,2),  
    lucro         decimal(9,2)   as (preco_venda - preco_custo),  
  
    check(preco_custo < preco_venda)  
)engine=InnoDB;
```

id é a chave primária e a cada inserção um valor único será gerado.
nome não pode ser nulo e cada registro deve ter um valor único.
lucro é sempre a diferença entre **preco_venda** e **preco_custo**.
preco_custo deve ser menor que **preco_venda**.

Restrições (Constraints)- SQL Server

Exemplos:

```
create table Produtos(  
    id int primary key identity,  
    nome varchar(60) not null unique,  
    precoCusto decimal(9,2),  
    precoVenda decimal(9,2),  
    lucro as precoVenda - precoCusto,  
    check (precoCusto < precoVenda)  
);
```

id é a chave primária e a cada inserção um valor único será gerado.
nome não pode ser nulo e cada registro deve ter um valor único.
lucro é sempre a diferença entre **preco_venda** e **preco_custo**.
preco_custo deve ser menor que **preco_venda**.

Restrições (Constraints)

Exemplos:

```
create table Fornecedores (  
    id            int            primary key      auto_increment,  
    nome          varchar(60)    not null        unique,  
    cidade        varchar(40),  
    estado        char(2)        default 'SP'  
)engine=InnoDB;
```

id é a chave primária e a cada inserção um valor único será gerado.
nome não pode ser nulo e cada registro deve ter um valor único.
estado será igual a 'SP' se nenhum outro for atribuído.

Restrições (Constraints)- SQLServer

```
CREATE TABLE persons
(
person_id INT PRIMARY KEY IDENTITY,
first_name NVARCHAR(100) NOT NULL,
last_name NVARCHAR(100) NOT NULL,
dob      DATE,
age_round AS CONVERT(INT, ROUND(DATEDIFF(HOUR, dob, GETDATE()) / 8766.0, 0)),
age_truncate as DATEDIFF(HOUR, dob, GETDATE())/8766,
age_decimal as DATEDIFF(HOUR, dob, GETDATE())/8766.0
);

INSERT INTO persons (first_name, last_name, dob) VALUES ('name 01', 'last name 01', '1983-05-15')
INSERT INTO persons (first_name, last_name, dob) VALUES ('name 02', 'last name 02', '1990-11-30')
INSERT INTO persons (first_name, last_name, dob) VALUES ('name 03', 'last name 03', '2005-02-25')
INSERT INTO persons (first_name, last_name, dob) VALUES ('name 04', 'last name 04', '1999-06-19')

select * from persons
```

Relacionamentos

Relacionamentos

- Em um relacionamento, um linha de uma tabela está relacionada a uma ou muitas outras linhas.
- Nos bancos de dados relacionais, é preciso **definir as relações por meio dos dados**.
- Veja o exemplo de **Alunos** que **moram** em uma **Cidade**:

Aluno

RA	Nome
1	Marie Currie
2	Alan Turing
3	Claude Shannon
4	Charles Darwin
5	Lynn Margulis

Cidade

Id	Nome	População
1	Sorocaba	120000
2	Tatuí	70000
3	São Paulo	300000
4	Itú	50000

Relacionamentos

- Sabendo que:
 - **Marie Curie** e **Charles Darwin** moram em **São Paulo**,
 - **Alan Turing** em **Sorocaba**,
 - **Claude Shannon** em **Itú** e
 - **Lynn Margulis** em **Tatuí**,
- como registrar isso no banco de dados?

Aluno

RA	Nome
1	Marie Currie
2	Alan Turing
3	Claude Shannon
4	Charles Darwin
5	Lynn Margulis

Cidade

Id	Nome	População
1	Sorocaba	120000
2	Tatuí	70000
3	São Paulo	300000
4	Itú	50000

Relacionamentos

- Nos SGBDR, a solução é criar uma **nova coluna** na tabela **Aluno** que **referencia** a coluna **Id** de **Cidade**.
- A coluna **Id** de **Cidade** deve ser uma **chave primária**, e a nova coluna **Cidade_id** de **Aluno** é chamada de **chave estrangeira**.
- Cada linha de **Aluno** pode referenciar uma linha de **Cidade** por meio das **chaves**.

Aluno

RA	Nome	Cidade_id
1	Marie Currie	3
2	Alan Turing	1
3	Claude Shannon	4
4	Charles Darwin	3
5	Lynn Margulis	2

Chave Estrangeira

Cidade

Id	Nome	População
1	Sorocaba	120000
2	Tatuí	70000
3	São Paulo	300000
4	Itú	50000

Chave Primária

Chaves Estrangeiras

Relacionamentos

- Uma chave estrangeira é um campo com a restrição foreign key.
- Ela sempre deverá referenciar uma chave primária.
- Deve ser do mesmo tipo que a chave primária a qual referencia.
- Pode ser definida na criação da tabela

Sintaxe simplificada para restrições foreign key:

```
create table Tabela1 (  
    campo_PK tipo          restrição,  
    campo2    tipo          restrição,  
    campo3    tipo          restrição,  
  
    primary key(campo_PK)  
);  
  
create table Tabela2 (  
    campo1    tipo          restrição,  
    campo2    tipo          restrição,  
    campo_FK  tipo          restrição  
  
    foreign key(campo_FK) references Tabela1(campo_PK);  
);
```

Relacionamentos

Exemplo:

```
create table Cidades (  
    id            int,  
    nome          varchar(60),  
    populacao     int,  
  
    primary key(id)  
)engine=InnoDB;
```

```
create table Alunos (  
    ra            int,  
    nome          varchar(60),  
    cidade_id     int ,  
  
    primary key(ra),  
    foreign key(cidade_id) references Cidades(id)  
)
```

Relacionamentos - MySQL

Exemplo:

```
create table Cidades (  
    id            int            primary key,  
    nome          varchar(60),  
    populacao     int  
)engine=InnoDB;  
  
create table Alunos (  
    ra            int            primary key,  
    nome          varchar(60),  
    cidade_id     int            references Cidades(id)  
)engine=InnoDB;
```

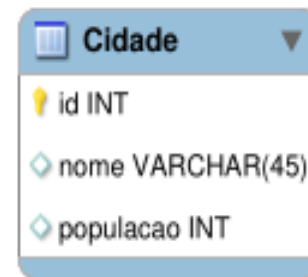
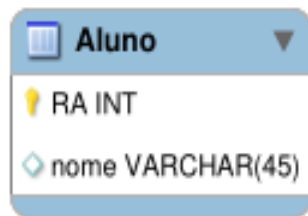
Relacionamentos – SQL Server

```
create database automovel;  
use automovel
```

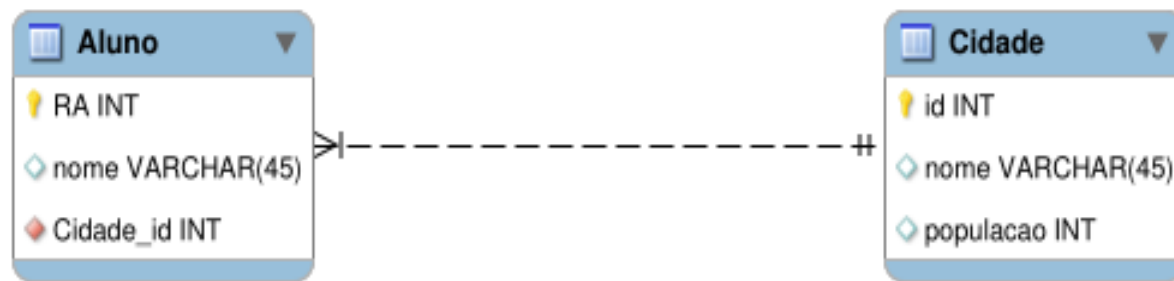
```
CREATE TABLE Pessoa  
(  
    ID_Pessoa integer PRIMARY KEY identity,  
    Nome varchar(255),  
    Endereco varchar(255),  
    Cidade varchar(255)  
);
```

```
CREATE TABLE Carro  
(  
    ID_Carro integer PRIMARY KEY identity,  
    Nome varchar(255),  
    Marca varchar(255),  
    ID_Pessoa integer,  
    CONSTRAINT fk_PesCarro  
    FOREIGN KEY (ID_Pessoa) REFERENCES Pessoa (ID_Pessoa)  
);
```

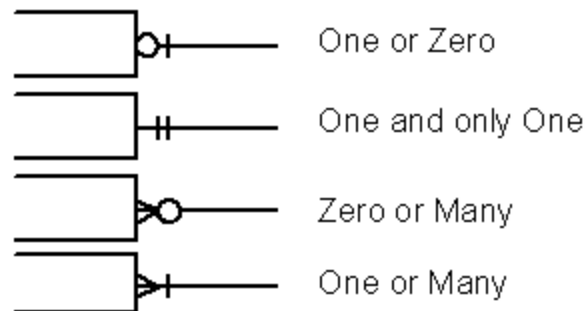
Relacionamentos - Notação do modelo Lógico



Relacionamentos - Notação do modelo Lógico



Summary of Crow's Foot Notation



Relacionamentos

- A inserção dos dados deve respeitar a restrição de chave estrangeira, ou seja,
- Todo valor na coluna chave estrangeira **deve estar contido na coluna chave primária**.

```
insert into Cidades values
```

```
(1, 'Sorocaba' , 120000),  
(2, 'Tatuí'    ,  70000),  
(3, 'São Paulo', 300000),  
(4, 'Itú'      ,  50000);
```

```
insert into Alunos values
```

```
(1, 'Marie Curie'   , 3), -- mora em São Paulo  
(2, 'Alan Turing'   , 1), -- mora em Sorocaba  
(3, 'Claude Shannon', 4), -- mora em Itú  
(4, 'Charles Darwin', 3), -- mora em São Paulo  
(5, 'Linn Margulis' , 2); -- mora em Tatuí
```

Relacionamentos

- A inserção dos dados deve respeitar a restrição de chave estrangeira, ou seja,
- Todo valor na coluna chave estrangeira **deve estar contido na coluna chave primária**.
- Caso haja uma tentativa de referenciar um valor que **não** existe, causará um **erro**, e os dados **não** serão inseridos.

```
insert into Cidades values
  (1, 'Sorocaba' , 120000),
  (2, 'Tatuí'    ,  70000),
  (3, 'São Paulo', 300000),
  (4, 'Itú'      ,  50000);
```

```
insert into Alunos values
  (1, 'Marie Curie'   , 3), -- mora em São Paulo
  (2, 'Alan Turing'   , 1), -- mora em Sorocaba
  (3, 'Claude Shannon', 4), -- mora em Itú
  (4, 'Charles Darwin', 3), -- mora em São Paulo
  (5, 'Linn Margulis' , 2); -- mora em Tatuí
```

```
insert into Alunos values
  (6, 'Nicola Tesla' , 5); -- mora onde???
```

Erro
↓

insert into Alunos values (6, 'Nicola Tesla' , 5) Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails

Para saber mais...

https://www.w3schools.com/sql/sql_constraints.asp

<https://imasters.com.br/banco-de-dados/voce-sabe-o-que-sao-storage-engines-mysql>

<https://dev.mysql.com/doc/refman/8.0/en/storage-engines.html>

<https://dev.mysql.com/doc/refman/8.0/en/innodb-storage-engine.html>

https://www.w3schools.com/sql/sql_constraints.asp

<https://dev.mysql.com/doc/refman/5.7/en/create-table-generated-columns.html>

<https://dev.mysql.com/doc/refman/5.7/en/innodb-auto-increment-handling.html>

<https://docs.oracle.com/javadb/10.8.3.0/ref/rrefsqlj13590.html>

https://www.w3schools.com/sql/sql_check.asp

<https://dev.mysql.com/doc/refman/5.6/en/create-table-foreign-keys.html>

https://www.w3schools.com/sql/sql_foreignkey.asp

<https://www.mysqltutorial.org/mysql-foreign-key/>

<https://www.devmedia.com.br/criando-uma-chave-estrangeira-no-mysql/20299>

https://www.cs.uct.ac.za/mit_notes/database/htmls/chp06.html

Exercícios

Exercícios

1- Crie um banco de dados para armazenar dados de Animais e Espécies. Um animal tem seu, id nome, data_nasc e peso.

Uma espécie tem um nome e uma descrição, o peso deve ser positivo.

Insira pelo menos 8 animais e 3 espécies.

2- Crie um banco de dados para registrar dados de Produtos e Marcas. Um produto deve ter, id, nome, preço de custo, preço de venda, data de validade e marca. Uma marca deve ter, id, nome, site oficial e telefone.

Aplique as restrições que ache corretas para o contexto dessa tabela.

Insira pelo menos 8 Produtos e 3 Marcas.

3 - Crie um banco de dados para registrar dados de Filmes e Categorias. Um filme tem seu título, sinopse, estudio e categoria. Uma categoria deve ter nome e público alvo.

Aplique as restrições que ache corretas para o contexto dessa tabela.

Insira pelo menos 8 Filmes e 3 Categorias.