

Introdução a Bancos de Dados SQL

Parte 2 – DDL e DML

Clodoveu Davis

Departamento de Ciência da Computação

Universidade Federal de Minas Gerais

clodoveu@dcc.ufmg.br



SQL

DML

Data Management Language

INSERT
DELETE
UPDATE

DDL

Data Definition Language

CREATE TABLE
CREATE INDEX
CREATE VIEW
ALTER TABLE
ALTER INDEX
DROP TABLE
DROP INDEX
DROP VIEW

DCL

Data Control Language

GRANT
REVOKE

DTL

Data Transaction Language

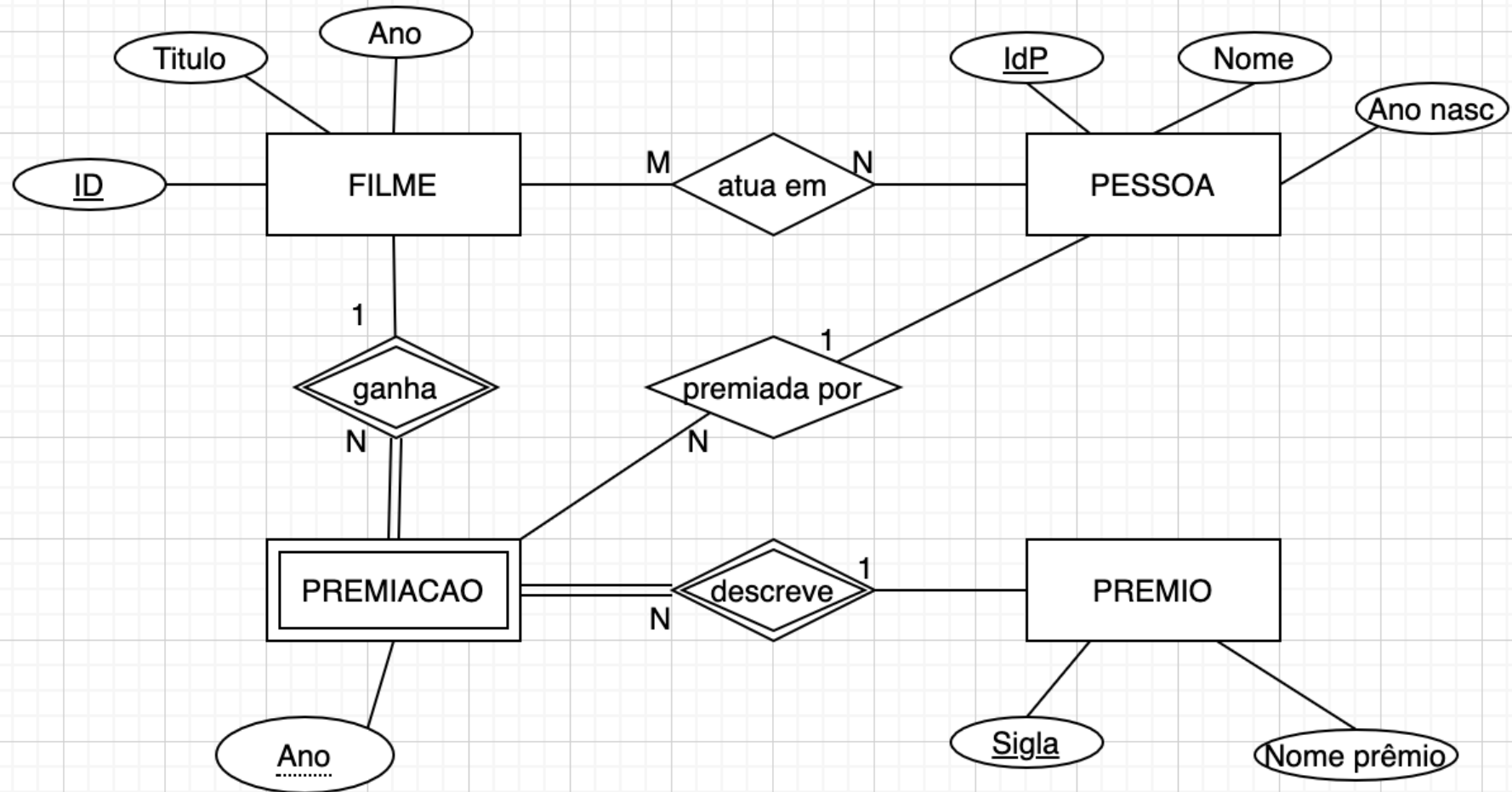
BEGIN TRANSACTION
COMMIT
ROLLBACK

DQL

Data Query Language

SELECT

Exemplo geral



FILME	ID	Titulo	AnoProd
-------	----	--------	---------

FP	ID	IdP
----	----	-----

PESSOA	IdP	Nome	AnoNasc
--------	-----	------	---------

PREMIACAO	ID	Sigla	IdP	AnoPremio
-----------	----	-------	-----	-----------

PREMIO	Sigla	NomePremio
--------	-------	------------

RELAÇÃO	ATRIBUTO	TIPO	PK	FK	ON DELETE	NULL	UNIQUE	DOMINIO
FILME	ID	integer	PK			NOT NULL		
	Titulo	varchar(100)				NOT NULL		
	AnoProd	integer				NOT NULL		>1900
FP	ID	integer	PK	references FILME(ID)	CASCADE			
	IdP	integer	PK	references PESSOA(IdP)	CASCADE			
	Activity	varchar(100)				NOT NULL		
PESSOA	IdP	integer	PK	NOT NULL				
	Nome	varchar(100)		NOT NULL				
	AnoNasc	integer		NOT NULL				
PREMIACAO	ID	integer	PK	references FILME(ID)	CASCADE	NOT NULL		
	Sigla	char(3)	PK	references PREMIO(Sigla)	CASCADE	NOT NULL		
	IdP	integer		references PESSOA(IdP)	CASCADE			
	AnoPremio	integer	PK			NOT NULL		> 1900
PREMIO	Sigla	char(3)	PK			NOT NULL		
	NomePremio	varchar(100)				NOT NULL		

Exemplo

FILME		
ID	Titulo	AnoProd
001	Amadeus	1984
002	Birdman	2014
003	Casino	1995
004	Die Hard	1988
005	El Cid	1961
006	Fargo	1996
007	Grease	1978
008	Hair	1979
009	It	2017
010	JFK	1991
011	Kundun	1997
012	Lincoln	2012
013	Misery	1990
014	Nixon	1995
015	Outbreak	1995
016	Psycho	1960
017	Quiz Show	1994
018	Rush	2013
019	Shrek	2001
020	Twister	1996
021	Up	2009
022	Vertigo	1958
023	WarGames	1983
024	X-Men	2000
025	Yesterday	2019
026	Zootopia	2016
027	The Godfather	1972
028	The Shawshank Redemption	1994
029	Goodfellas	1990
030	The Godfather Part II	1974

PESSOA		
IdP	Nome	AnoNasc
111	Charlton Heston	1923
114	Dustin Hoffman	1937
115	Lily James	1989
117	Frances McDormand	1957
118	Alfred Hitchcock	1899
151	Shakira	1977
207	Kathy Bates	1948
208	Sharon Stone	1958
215	Hugh Jackman	1968
217	Anthony Perkins	1932
221	Francis Ford Coppola	1939
222	Robert DeNiro	1943
229	James Caan	1940
245	Al Pacino	1940
260	Ralph Fiennes	1962
276	Stephen King	1947
279	Tom Hulce	1953
281	Oliver Stone	1946
298	Morgan Freeman	1937
311	Helen Hunt	1963
324	Daniel Day-Lewis	1957
338	Milos Forman	1932
342	Martin Scorsese	1942
352	Mike Myers	1963
356	Bruce Willis	1955
367	Matthew Broderick	1962
368	Famke Janssen	1964
382	Emma Stone	1988
427	John Travolta	1954
428	Tim Robbins	1958
438	Kevin Costner	1955
439	Chris Hemsworth	1983
446	Anthony Hopkins	1937
453	Edward Asner	1929
456	Olivia Newton-John	1948
495	Michael Keaton	1951
900	Joel Coen	1954
901	F. Murray Abraham	1939
907	Marlon Brando	1924
950	Robert Redford	1936
999	Steven Spielberg	1946

FP		
ID	N	Activity
001	338	Director
001	279	Actor
001	901	Actor
002	495	Actor
002	382	Actor
003	208	Actor
003	222	Actor
004	356	Actor
005	111	Actor
006	117	Actor
006	900	Director
007	427	Actor
007	456	Actor
008	338	Director
009	276	Writer
010	438	Actor
010	281	Director
011	342	Director
012	324	Actor
012	999	Director
013	207	Actor
013	229	Actor
013	276	Writer
014	446	Actor
015	114	Actor
016	118	Director
016	217	Actor
017	260	Actor
017	950	Director
018	439	Actor
019	352	Actor
020	311	Actor
021	453	Actor
022	118	Director
023	367	Actor
024	215	Actor
024	368	Actor
025	115	Actor
026	151	Actor
027	222	Actor
027	229	Actor
027	245	Actor
027	221	Director
027	907	Actor
028	428	Actor
028	298	Actor
029	222	Actor
029	342	Director
030	245	Actor
030	222	Actor
030	221	Director
030	907	Actor

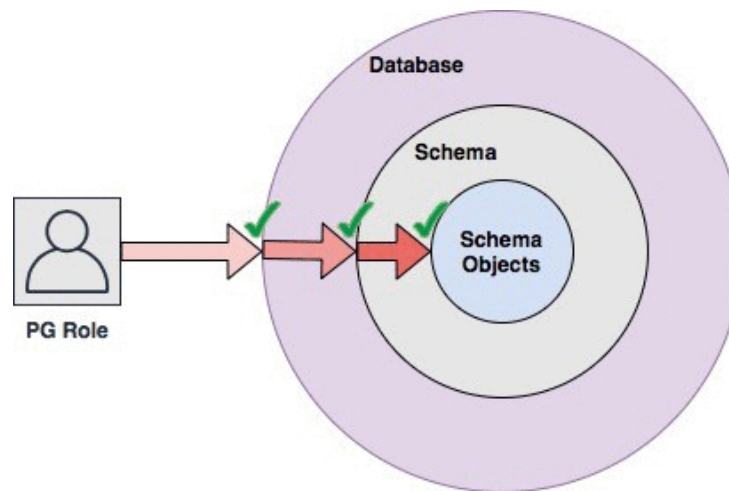
PREMIACAO			
ID	Sigla	IdP	AnoPremio
001	OBP	NULL	1984
001	OBA	901	1984
002	OBP	NULL	2014
002	NBA	495	1984
003	NBA	208	1995
006	NBP	NULL	1996
006	NBD	900	1996
010	NBP	NULL	1991
010	NBD	281	1991
012	NBP	NULL	2012
012	OBA	324	2012
012	NBD	999	2012
013	OBA	207	1990
016	NBD	118	1960
017	NBP	NULL	1994
017	NBD	950	1994
021	NBP	NULL	2009
027	OBP	NULL	1972
027	OBA	907	1972
027	NBD	221	1972
028	NBP	NULL	1994
030	OBP	NULL	1974
030	NBA	245	1974
030	OBD	221	1974

PREMIO	
Sigla	Premio
NBA	Oscar Nominee Best Actor/Actress
NBD	Oscar Nominee Best Director
NBP	Oscar Nominee Best Picture
OBA	Oscar Best Actor
OBD	Oscar Best Director
OBP	Oscar Best Picture

DDL – Data Definition Language

- Criação e alteração de tabelas, índices e visões
 - E mais: criação de BDs, esquemas, restrições, usuários...
- Desde o SQL 2, o catalogo do BD é uma coleção de esquemas
 - No PostgreSQL, p. ex., o esquema default de cada BD é chamado *public*
 - Esquemas são mecanismos para melhor gerenciar os dados, agrupando-os de forma lógica. Isso também facilita o controle de segurança/acesso e o backup

Roles são grupos de usuários no PostgreSQL: definem permissões comuns a um grupo de usuários



Schema objects: tabelas, índices, restrições, tipos de dados, funções, operadores...

Nome “completo”:
database.schema.tablename

CREATE TABLE

CREATE TABLE <nome>

(<nomeColuna> <tipoColuna> [<restriçãoAtributo>]

[, <nomeColuna> <tipoColuna>

[<restriçãoAtributo>]]

[, restriçãoTabela [, restriçãoTabela]])

Tipos de colunas

- Tipos de dados e domínios
 - Numéricos: INTEGER, INT, SMALLINT, FLOAT, REAL, DOUBLE PRECISION, DECIMAL(i,j), DEC(i,j), NUMERIC(i,j), NUMBER(i,j)
 - Strings: CHAR(n), CHARACTER(n), VARCHAR(n), CHAR VARYING(n)
 - Strings de bits: BIT(n), BIT VARYING(n)
 - Data e hora: DATE, TIME, TIME(i), TIME WITH TIME ZONE, TIMESTAMP, TIMESTAMP WITH TIME ZONE, INTERVAL
- Variam um pouco entre SGBDs
- Alguns permitem criar novos tipos

Restrições de colunas

- Restrições de atributo
 - NOT NULL
 - PRIMARY KEY
 - UNIQUE
 - Chave alternativa
 - REFERENCES <tabela>(<atributo>)
ON DELETE [SET NULL | DEFAULT | CASCADE]
ON UPDATE [SET NULL | DEFAULT | CASCADE]
 - Chave estrangeira
 - DEFAULT <valor>

Restrições de tabela

- Pode-se dar nome a uma restrição
 - CONSTRAINT <nome> PRIMARY KEY (<atributo>)
 - CONSTRAINT <nome> FOREIGN KEY (<atributo>)
REFERENCES <tabela>(<atributo>)
ON DELETE [SET NULL | DEFAULT | CASCADE]
ON UPDATE [SET NULL | DEFAULT | CASCADE]
 - CONSTRAINT <nome> UNIQUE (<atributo>)
- Dar um nome é boa prática, e é conveniente para que se possa posteriormente alterar ou cancelar a restrição

```
create table filme (
    id integer not null,
    titulo varchar(100) not null,
    anoprod integer check (anoprod > 1900),
    constraint pk_filme primary key (id)
);
```

```
create table pessoa (
    idp integer not null,
    nome varchar(100) not null,
    anonasc integer not null,
    constraint pk_pessoa primary key (idp)
);
```

```
create table premio (
    sigla char(3) not null,
    nomepremio varchar(100) not null,
    constraint pk_premio primary key (sigla)
);
```

```
create table fp (
    id integer not null,
    idp integer not null,
    activity varchar(100) not null,
    constraint pk_fp primary key(id, idp),
    constraint fk_fp_filme foreign key (id)
        references filme(id) on delete cascade,
    constraint fk_fp_pessoa foreign key (idp)
        references pessoa(idp) on delete cascade
);
```

```
create table premiacao (
    id integer not null,
    sigla char(3) not null,
    idp integer,
    anopremio integer not null,
    constraint pk_premiacao primary key (id, sigla, anopremio),
    constraint fk_premiacao_filme foreign key (id)
        references filme(id) on delete cascade,
    constraint fk_premiacao_premio foreign key (sigla)
        references premio(sigla) on delete cascade,
    constraint fk_premiacao_pessoa foreign key (idp)
        references pessoa(idp) on delete cascade,
    constraint ck_anopremio check (anopremio > 1900)
);
```

Alternativas

```
create table filme (  
    id integer not null primary key,  
    titulo varchar(100) not null,  
    anoprod integer check (anoprod > 1900)  
);
```

```
create table fp (  
    id integer not null references filme(id) on delete cascade,  
    idp integer not null references pessoa(idp) on delete cascade,  
    activity varchar(100) not null,  
    constraint pk_fp primary key(id, idp),  
);
```

Mais comandos DDL

- DROP TABLE <nome> [CASCADE | RESTRICT]
 - RESTRICT: a tabela só será excluída se ela não for referenciada em nenhuma restrição nem em visões
- CREATE SCHEMA <nome>
- DROP SCHEMA <nome> [CASCADE | RESTRICT]
 - RESTRICT: só é excluído o esquema se ele não contiver nenhum elemento

Mais comandos DDL

- ALTER TABLE: *evolução de esquemas*
 - ALTER TABLE <nome> ADD <atributo> <tipo>
 - Não é permitido configurar NOT NULL neste caso
 - ALTER TABLE <nome> DROP <atributo> [CASCADE | RESTRICT]
 - ALTER TABLE <nome> ALTER <atributo> [SET DEFAULT <valor> | DROP DEFAULT]
 - ALTER TABLE <nome> ADD CONSTRAINT
 - ALTER TABLE <nome> DROP CONSTRAINT <nome> [CASCADE | RESTRICT]
 - Só se a restrição tiver recebido nome no CREATE TABLE
- Geradores de comandos DDL a partir da modelagem conceitual costumam preferir gerar CREATE TABLEs simples, seguidos de comandos ALTER TABLE ... ADD CONSTRAINT.

Visões

- Visão (View) é uma tabela “virtual” que é derivada de outras tabelas
 - As tabelas armazenadas fisicamente são chamadas de tabelas de base
- Uma visão não necessariamente existe fisicamente
 - Se existir é dita uma visão materializada
- Como é virtual, algumas operações de atualização podem ser limitadas ou bloqueadas
 - P. ex., se atualizar a visão causar violação de integridade em alguma tabela de base
- Visões são um meio de deixar disponíveis consultas às quais é necessário fazer referência constantemente

Visões

- **CREATE VIEW**

- `CREATE VIEW <nome>`
`AS SELECT... FROM... WHERE...`
- `CREATE VIEW <nome>`
`AS SELECT... FROM... WHERE... GROUP BY... HAVING...`

- **Acesso à visão: com SELECT, como em uma tabela comum**

- `DROP VIEW <nome>`

Exemplo

```
create view oscarwinners as (  
    select distinct titulo, anoprod, x.nome, q.nomepremio  
    from filme f, premiacao p, premio q, pessoa x  
    where f.id = p.id  
    and q.sigla = p.sigla  
    and x.idp = p.idp  
    and q.nomepremio LIKE '%Oscar Best%'  
    UNION  
    select distinct titulo, anoprod, NULL, 'Best Picture'  
    from filme f, premiacao p, premio q  
    where f.id = p.id  
    and q.sigla = p.sigla  
    and p.idp IS NULL  
    and q.nomepremio LIKE '%Oscar Best%'  
)
```

select *
from oscarwinners
order by titulo

titulo character varying (100)	anoprod integer	nome character varying	nomepremio character varying
Amadeus	1984	F. Murray Abraham	Oscar Best Actor
Amadeus	1984	[null]	Best Picture
Birdman	2014	[null]	Best Picture
Lincoln	2012	Daniel Day-Lewis	Oscar Best Actor
Misery	1990	Kathy Bates	Oscar Best Actor
The Godfather	1972	Marlon Brando	Oscar Best Actor
The Godfather	1972	[null]	Best Picture
The Godfather Part II	1974	[null]	Best Picture
The Godfather Part II	1974	Francis Ford Coppola	Oscar Best Director

select *
from oscarwinners
where anoprod > 2000
order by anoprod desc

titulo character varying (100)	anoprod integer	nome character varying	nomepremio character varying
Birdman	2014	[null]	Best Picture
Lincoln	2012	Daniel Day-Lewis	Oscar Best Actor

DML: Data Manipulation Language

- Composta basicamente de três comandos:
 - INSERT
 - DELETE
 - UPDATE

Inserção

- **INSERT**

- `INSERT INTO <tabela>
VALUES (lista);` -- na ordem das colunas
- `INSERT INTO <tabela>(atribos)
VALUES (lista);`
- `INSERT INTO <tabela>(atribos)
SELECT... FROM... WHERE`
-- resultados do select precisam casar com a lista de atributos

```
insert into filme(id, titulo, anoprod) values (001, 'Amadeus', 1984);
insert into filme(id, titulo, anoprod) values (002, 'Birdman', 2014);
insert into filme(id, titulo, anoprod) values (003, 'Casino', 1995);
insert into filme(id, titulo, anoprod) values (004, 'Die Hard', 1988);
insert into filme(id, titulo, anoprod) values (005, 'El Cid', 1961);
insert into filme(id, titulo, anoprod) values (006, 'Fargo', 1996);
insert into filme(id, titulo, anoprod) values (007, 'Grease', 1978);
insert into filme(id, titulo, anoprod) values (008, 'Hair', 1979);
insert into filme(id, titulo, anoprod) values (009, 'It', 2017);
insert into filme(id, titulo, anoprod) values (010, 'JFK', 1991);
insert into filme(id, titulo, anoprod) values (011, 'Kundun', 1997);
insert into filme(id, titulo, anoprod) values (012, 'Lincoln', 2012);
insert into filme(id, titulo, anoprod) values (013, 'Misery', 1990);
```

→ Default: mesma ordem do CREATE TABLE

```
insert into pessoa(idp, nome, anonasc) values (111, 'Charlton Heston', 1923);
insert into pessoa(idp, nome, anonasc) values (114, 'Dustin Hoffman', 1937);
insert into pessoa(idp, nome, anonasc) values (115, 'Lily James', 1989);
insert into pessoa(idp, nome, anonasc) values (117, 'Frances McDormand', 1957);
insert into pessoa(idp, nome, anonasc) values (118, 'Alfred Hitchcock', 1899);
insert into pessoa(idp, nome, anonasc) values (151, 'Shakira', 1977);
insert into pessoa(idp, nome, anonasc) values (207, 'Kathy Bates', 1948);
insert into pessoa(idp, nome, anonasc) values (208, 'Sharon Stone', 1958);
insert into pessoa(idp, nome, anonasc) values (215, 'Hugh Jackman', 1968);
```

select * from oscarwinners
order by titulo

titulo character varying (100)	anoprod integer	nome character varying	nomepremio character varying
Amadeus	1984	F. Murray Abraham	Oscar Best Actor
Amadeus	1984	[null]	Best Picture
Birdman	2014	[null]	Best Picture
Lincoln	2012	Daniel Day-Lewis	Oscar Best Actor
Misery	1990	Kathy Bates	Oscar Best Actor
The Godfather	1972	Marlon Brando	Oscar Best Actor
The Godfather	1972	[null]	Best Picture
The Godfather Part II	1974	[null]	Best Picture
The Godfather Part II	1974	Francis Ford Coppola	Oscar Best Director

select *
from oscarwinners
where anoprod > 2000
order by anoprod desc

titulo character varying (100)	anoprod integer	nome character varying	nomepremio character varying
Birdman	2014	[null]	Best Picture
Lincoln	2012	Daniel Day-Lewis	Oscar Best Actor

insert into oscarwinners(titulo, anoprod, nome, nomepremio)
values ('Parasite', 2019, NULL, 'BestPicture')

ERROR: cannot insert into view "oscarwinners"
DETAIL: Views containing UNION, INTERSECT, or EXCEPT are not automatically updatable.
HINT: To enable inserting into the view, provide an INSTEAD OF INSERT trigger or an unconditional ON INSERT DO INSTEAD rule.
SQL state: 55000

```
create view simple_filme as (
    select id as codfilme, titulo as nomefilme
    from filme
)
```

```
select * from simple_filme
```

codfilme integer	nomefilme character varying (100)
1	Amadeus
2	Birdman
3	Casino
4	Die Hard
5	El Cid
6	Fargo
7	Grease
8	Hair
9	It
10	JFK
11	Kundun
12	Lincoln
13	Misery

```
insert into simple_filme(codfilme, nomefilme)
values (99, 'Parasite');
```

```
select * from filme
```

26	26	Zootopia	2016
27	27	The Godfather	1972
28	28	The Shawshank Redemption	1994
29	29	Goodfellas	1990
30	30	The Godfather Part II	1974
31	99	Parasite	[null]

UAI!

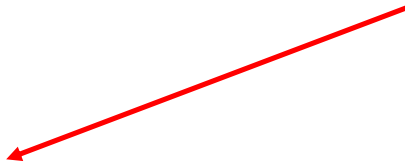
```
create table filme (
    id integer not null,
    titulo varchar(100) not null,
    anoprod integer check (anoprod > 1900),
    constraint pk_filme primary key (id)
);
```

NULL não é comparável com 1900, e não foi definido NOT NULL

```
create view filme_sem_codigo as (  
    select titulo as nomefilme, anoprod  
    from filme  
)
```

```
insert into filme_sem_codigo(nomefilme, anoprod)  
values ('Parasite', 2019);
```

Na tabela de base
FILME



```
ERROR: null value in column "id" violates not-null constraint  
DETAIL: Failing row contains (null, Parasite, 2019).  
SQL state: 23502
```


Exclusão

- DELETE

- DELETE FROM <tabela>
WHERE <condição>;
- DELETE FROM <tabela>
WHERE <atrib> IN (SELECT... FROM ... WHERE)
- DELETE FROM <tabela>
-- Deleta TODAS AS LINHAS da tabela!!

```
delete from pessoa  
where nome = 'Francis Ford Coppola'
```

Como estabelecemos ON DELETE CASCADE nas chaves estrangeiras, a exclusão de uma linha de PESSOA vai causar a exclusão das linhas correspondentes de PREMIACAO e FP em relação a PESSOA

Exemplo

FILME		
ID	Titulo	AnoProd
001	Amadeus	1984
002	Birdman	2014
003	Casino	1995
004	Die Hard	1988
005	El Cid	1961
006	Fargo	1996
007	Grease	1978
008	Hair	1979
009	It	2017
010	JFK	1991
011	Kundun	1997
012	Lincoln	2012
013	Misery	1990
014	Nixon	1995
015	Outbreak	1995
016	Psycho	1960
017	Quiz Show	1994
018	Rush	2013
019	Shrek	2001
020	Twister	1996
021	Up	2009
022	Vertigo	1958
023	WarGames	1983
024	X-Men	2000
025	Yesterday	2019
026	Zootopia	2016
027	The Godfather	1972
028	The Shawshank Redemption	1994
029	Goodfellas	1990
030	The Godfather Part II	1974

PESSOA		
IdP	Nome	AnoNasc
111	Charlton Heston	1923
114	Dustin Hoffman	1937
115	Lily James	1989
117	Frances McDormand	1957
118	Alfred Hitchcock	1899
151	Shakira	1977
207	Kathy Bates	1948
208	Sharon Stone	1958
215	Hugh Jackman	1968
217	Anthony Perkins	1932
221	Francis Ford Coppola	1939
222	Robert DeNiro	1943
229	James Caan	1940
245	Al Pacino	1940
260	Ralph Fiennes	1962
276	Stephen King	1947
279	Tom Hulce	1953
281	Oliver Stone	1946
298	Morgan Freeman	1937
311	Helen Hunt	1963
324	Daniel Day-Lewis	1957
338	Milos Forman	1932
342	Martin Scorsese	1942
352	Mike Myers	1963
356	Bruce Willis	1955
367	Matthew Broderick	1962
368	Famke Janssen	1961
382	Emma Stone	1988
427	John Travolta	1954
428	Tim Robbins	1958
438	Kevin Costner	1955
439	Chris Hemsworth	1983
446	Anthony Hopkins	1937
453	Edward Asner	1929
456	Olivia Newton-John	1948
495	Michael Keaton	1951
900	Joel Coen	1954
901	F. Murray Abraham	1939
907	Marlon Brando	1924
950	Robert Redford	1936
999	Steven Spielberg	1946

FP		
ID	N	Activity
001	338	Director
001	279	Actor
001	901	Actor
002	495	Actor
002	382	Actor
003	208	Actor
003	222	Actor
004	356	Actor
005	111	Actor
006	117	Actor
006	900	Director
007	427	Actor
007	456	Actor
008	338	Director
009	276	Writer
010	438	Actor
010	281	Director
011	342	Director
012	324	Actor
012	999	Director
013	207	Actor
013	229	Actor
013	276	Writer
014	446	Actor
015	114	Actor
016	118	Director
016	217	Actor
017	260	Actor
017	950	Director
018	439	Actor
019	352	Actor
020	311	Actor
021	453	Actor
022	118	Director
023	367	Actor
024	215	Actor
024	368	Actor
025	115	Actor
026	151	Actor
027	222	Actor
027	229	Actor
027	245	Actor
027	221	Director
027	907	Actor
028	428	Actor
028	298	Actor
029	222	Actor
029	342	Director
030	245	Actor
030	222	Actor
030	221	Director
030	907	Actor

PREMIACAO			
ID	Sigla	IdP	AnoPremio
001	OBP	NULL	1984
001	OBA	901	1984
002	OBP	NULL	2014
002	NBA	495	1984
003	NBA	208	1995
006	NBP	NULL	1996
006	NBD	900	1996
010	NBP	NULL	1991
010	NBD	281	1991
012	NBP	NULL	2012
012	OBA	324	2012
012	NBD	999	2012
013	OBA	207	1990
016	NBD	118	1960
017	NBP	NULL	1994
017	NBD	950	1994
021	NBP	NULL	2009
027	OBP	NULL	1972
027	OBA	907	1972
027	NBD	221	1972
028	NBP	NULL	1994
030	OBP	NULL	1974
030	NBA	245	1974
030	OBD	221	1974

PREMIO	
Sigla	Premio
NBA	Oscar Nominee Best Actor/Actress
NBD	Oscar Nominee Best Director
NBP	Oscar Nominee Best Picture
OBA	Oscar Best Actor
OBD	Oscar Best Director
OBP	Oscar Best Picture

```
alter table fp drop constraint fk_fp_pessoa;
```

```
alter table fp add constraint fk_fp_pessoa foreign key (idp) references pessoa (idp) on delete restrict;
```

```
alter table premiacao drop constraint fk_premiacao_pessoa;
```

```
alter table premiacao add constraint fk_premiacao_pessoa foreign key (idp)  
references pessoa (idp) on delete restrict;
```

```
delete from pessoa  
where nome = 'Francis Ford Coppola'
```

```
ERROR:  update or delete on table "pessoa" violates foreign key constraint "fk_fp_pessoa" on table "fp"  
DETAIL:  Key (idp)=(221) is still referenced from table "fp".  
SQL state: 23503
```

Atualização

- **UPDATE**

- UPDATE <tabela>
SET <atributo>=<valor> [, <atributo>=<valor>]
WHERE <condição>;
- UPDATE <tabela>
SET <atributo>=<valor> [, <atributo>=<valor>]
WHERE <atributo> IN (SELECT... FROM... WHERE...);

```
update filme  
set anoprod = 1756 -- ano de nascimento de Mozart  
where titulo = 'Amadeus'
```

```
ERROR:  new row for relation "filme" violates check constraint "filme_anoprod_check"  
DETAIL:  Failing row contains (1, Amadeus, 1756).  
SQL state: 23514
```

```
update filme  
set titulo = upper(titulo)  
where id in (select id from filme where anoprod < 1960)
```

Para praticar

- **GeoSQL: não permite CREATE, INSERT, DELETE, UPDATE**
- SQL Fiddle: você entra com tabelas e dados, e executa operações; permite CREATE, INSERT, DELETE, UPDATE, mas tem limitações
 - <http://sqlfiddle.com>
- w3schools.com/sql: tutorial de muitas funções, com exemplos interativos
 - <http://w3schools.com/sql>
- No seu ambiente: instale MySQL ou PostgreSQL, por exemplo.
 - Um dataset está disponível na página da disciplina

