

**UNIVERSIDADE FEDERAL DE VIÇOSA**  
**CAMPUS FLORESTAL**

SAMUEL SENA - 3494

TRABALHO PRÁTICO II

FLORESTAL

2019



## Introdução

O objetivo do trabalho é a implementação de um pequeno banco de dados para fins didáticos. O conteúdo armazenado pelo respectivo banco é composto por informações de diversos consoles e jogos amplamente conhecidos e lançados no mercado até a atualidade.

Para gerar o respectivo banco de dados, os seguintes comandos do MySQL foram utilizados:

```
CREATE DATABASE colecaogames;

CREATE TABLE jogo(
  idjogo INT PRIMARY KEY,
  nomejogo VARCHAR(30),
  descricao VARCHAR(30)
);

CREATE TABLE colecaogames.fabricante(
  idfabricante INT PRIMARY KEY,
  nomefab VARCHAR(30),
  ano_fundacao INT
);

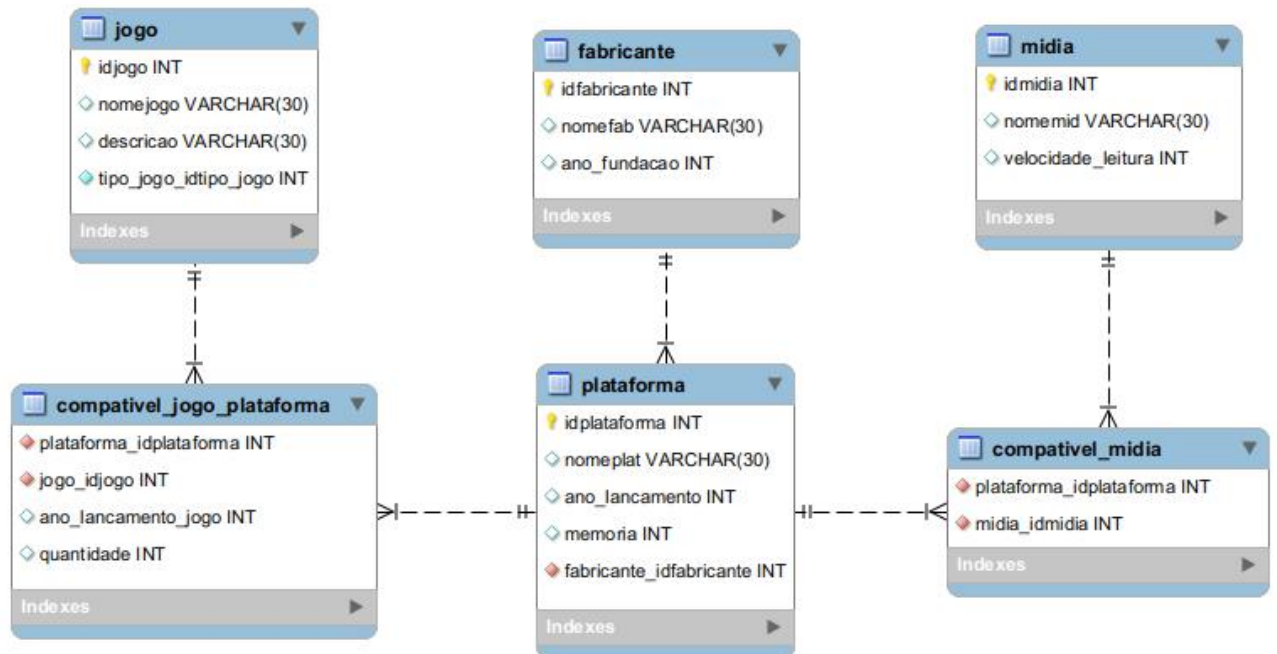
CREATE TABLE colecaogames.midia(
  idmidia INT PRIMARY KEY,
  nomemid VARCHAR(30),
  velocidade_leitura INT
);

CREATE TABLE colecaogames.plataforma(
  idplataforma INT PRIMARY KEY,
  nomeplat VARCHAR(30),
  ano_lancamento INT,
  memoria INT,
  idfabricante INT,
  CONSTRAINT idfabricante FOREIGN KEY (idfabricante) REFERENCES fabricante (idfabricante)
);

CREATE TABLE colecaogames.compativel_midia(
  idplataforma INT,
  CONSTRAINT idplataforma FOREIGN KEY (idplataforma) REFERENCES plataforma (idplataforma),
  idmidia INT,
  CONSTRAINT idmidia FOREIGN KEY (idmidia) REFERENCES midia (idmidia)
);

CREATE TABLE colecaogames.compativel_jogo_plataforma(
  idplataforma INT,
  idjogo INT,
  CONSTRAINT idjogo FOREIGN KEY (idjogo) REFERENCES jogo (idjogo),
  quantidade INT,
  ano_lancamentoJogo INT
);
```

O seguinte diagrama corresponde a implementação das tabelas descritas no código MySQL acima:



Os tópicos a seguir correspondem as respostas para cada pergunta realizada na descrição do trabalho:

1)

```

SELECT DISTINCT nomeplat
FROM (plataforma AS P NATURAL JOIN compativel_midia) NATURAL JOIN midia AS M
WHERE velocidade_leitura > 100;
    
```

nomeplat
Atari 2600
NES
SNES

2)

```
SELECT nomeplat, ano_lancamento
FROM plataforma NATURAL JOIN fabricante
WHERE ano_fundacao > 1970;
```

nomeplat	ano_lancamento
Xbox360	2005
Atari 2600	1977

3)

```
SELECT nomemid, velocidade_leitura
FROM midia
WHERE velocidade_leitura BETWEEN 10 AND 30
ORDER BY velocidade_leitura DESC, nomemid ASC;
```

nomemid	velocidade_leitura
DVD	20
CD	10

4)

```
INSERT INTO `colecaogames`.`plataforma` (`idplataforma`, `nomeplat`, `ano_lancamento`, `memoria`, `idfabricante`)
VALUES (7, 'WiiU', 2012, '214748364',
(SELECT idfabricante
FROM fabricante
WHERE nomefab = 'Nintendo'));
SELECT * FROM plataforma;
```

idplataforma	nomeplat	ano_lancamento	memoria	idfabricante
1	Atari 2600	1977	128	4
2	NES	1985	2048	1
3	SNES	1990	131072	1
4	Wii	2006	92274688	1
5	Xbox360	2005	536870912	3
6	PlayStation3	2006	536870912	2
7	WiiU	2012	214748364	1

5)

```
SELECT nomeplat, nomefab, SUM(quantidade) AS numeroJogos
FROM (compativel_jogo_plataforma AS CPP NATURAL JOIN plataforma) NATURAL JOIN fabricante
GROUP BY CPP.idplataforma
ORDER BY nomeplat;
```

nomeplat	nomefab	numeroJogos
Atari 2600	Atari	4
NES	Nintendo	2
PlayStation3	Sony	6
SNES	Nintendo	1
Wii	Nintendo	4
Xbox360	Microsoft	4

6)

```
SELECT nomejogo
FROM compativel_jogo_plataforma NATURAL JOIN jogo
GROUP BY nomejogo
HAVING COUNT(*) > 2;
```

nomejogo
FIFA12
Street Fighter IV

7)

```
SELECT ano_lancamentoJogo, SUM(quantidade) AS jogosLancados
FROM compativel_jogo_plataforma
GROUP BY ano_lancamentoJogo
ORDER BY ano_lancamentoJogo ASC;
```

ano_lancamentoJogo	jogosLancados
1982	2
1983	3
1985	1
1990	1
2006	1
2007	2
2008	4
2010	1
2011	4
2012	2

8)

```
SELECT nomejogo, ano_lancamentoJogo, nomeplat
FROM (compativel_jogo_plataforma NATURAL JOIN jogo) NATURAL JOIN plataforma
WHERE ano_lancamentoJogo <= ALL (SELECT ano_lancamentoJogo FROM compativel_jogo_plataforma);
```

nomejogo	ano_lancamentoJogo	nomeplat
River Raid	1982	Atari 2600

9)

```
SELECT nomejogo, truncate(AVG(memoria),0) AS memoria_utilizada
FROM (compativel_jogo_plataforma NATURAL JOIN plataforma) NATURAL JOIN jogo
GROUP BY nomejogo;
```

nomejogo	memoria_utilizada
River Raid	128
Mario Bros	1088
F-Zero	46202880
Super Mario Bros	46138368
Wii Sports	92274688
FIFA12	388672170
God of War 3	536870912
Forza MotorSport	536870912
Gran Turismo 5	536870912
Street Fighter IV	536870912



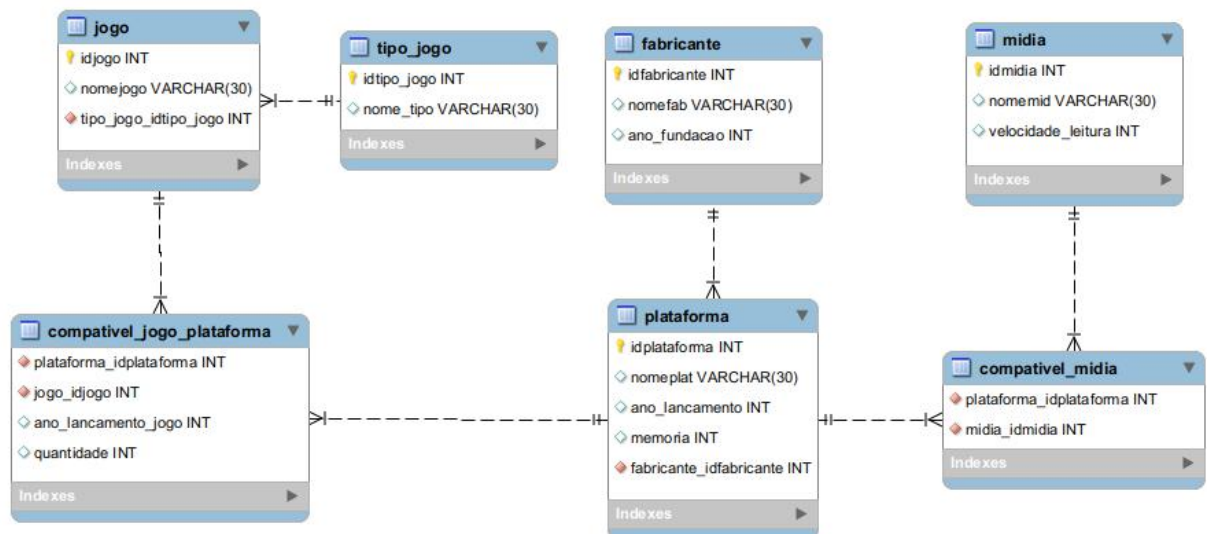
10)

```
SELECT nomeplat
FROM compativel_midia NATURAL JOIN plataforma
GROUP BY nomeplat
HAVING COUNT(*) > 1
ORDER BY nomeplat ASC;
```

nomeplat
NES
PlayStation3
Wii
Xbox360

11 e 12)

Ao final do trabalho, foi pedido que uma alteração no banco de dados fosse efetuada, de forma que o banco fosse expandido. O novo diagrama do banco (já com a modificação efetuada) corresponde a seguinte imagem abaixo:



Como é possível podemos ver, a alteração escolhida foi retirar o campo descrição da tabela jogo, em seguida criamos tipo\_jogo para assim evitarmos a repetição que acontecia ao digitar o mesmo tipo do jogo em tuplas diferentes. Dessa forma reduzimos o espaço necessário de armazenagem dos dados de cada jogo. Para não perdermos os dados já inseridos na tabela jogo, antes de remover a coluna *descricao*, será realizada uma consulta que irá adicionar o código do tipo do jogo de acordo com sua descrição atual, após isso, a retirada da coluna poderá ser realizada normalmente.

Tendo isso em vista, os seguintes comandos foram realizados:

Para criar a nova tabela:

```
CREATE TABLE tipo_jogo(  
    idTipo_jogo INT PRIMARY KEY,  
    nome_tipo VARCHAR(30)  
);  
  
ALTER TABLE jogo ADD idTipo_jogo INT DEFAULT 1;  
  
INSERT INTO `colecaogames`.`tipo_jogo` (`idTipo_jogo`, `nome_tipo`) VALUES (1, 'Tiro');  
INSERT INTO `colecaogames`.`tipo_jogo` (`idTipo_jogo`, `nome_tipo`) VALUES (2, 'Aventura');  
INSERT INTO `colecaogames`.`tipo_jogo` (`idTipo_jogo`, `nome_tipo`) VALUES (3, 'Corrida');  
INSERT INTO `colecaogames`.`tipo_jogo` (`idTipo_jogo`, `nome_tipo`) VALUES (4, 'Esportes variados');  
INSERT INTO `colecaogames`.`tipo_jogo` (`idTipo_jogo`, `nome_tipo`) VALUES (5, 'Futebol');  
INSERT INTO `colecaogames`.`tipo_jogo` (`idTipo_jogo`, `nome_tipo`) VALUES (6, 'Ação');  
INSERT INTO `colecaogames`.`tipo_jogo` (`idTipo_jogo`, `nome_tipo`) VALUES (7, 'Corrida Realista');  
INSERT INTO `colecaogames`.`tipo_jogo` (`idTipo_jogo`, `nome_tipo`) VALUES (8, 'Luta');
```

Para realizar as alterações estruturais, sem perda de informação:

```
UPDATE jogo  
SET idTipo_jogo = 2  
WHERE descricao = 'Aventura';  
  
UPDATE jogo  
SET idTipo_jogo = 3  
WHERE descricao = 'Corrida';  
  
UPDATE jogo  
SET idTipo_jogo = 4  
WHERE descricao = 'Esportes Variados';  
  
UPDATE jogo  
SET idTipo_jogo = 5  
WHERE descricao = 'Futebol';  
  
UPDATE jogo  
SET idTipo_jogo = 6  
WHERE descricao = 'Ação';  
  
UPDATE jogo  
SET idTipo_jogo = 7  
WHERE descricao = 'Corrida Realista';  
  
UPDATE jogo  
SET idTipo_jogo = 8  
WHERE descricao = 'Luta';
```



E por fim, para definir a *foreign key* e remover a coluna não mais utilizada:

```
ALTER TABLE jogo ADD FOREIGN KEY (idTipo_jogo) REFERENCES tipo_jogo (idTipo_jogo);  
ALTER TABLE jogo DROP COLUMN descricao;
```

idTipo_jogo	nome_tipo
1	Tiro
2	Aventura
3	Corrida
4	Esportes variados
5	Futebol
6	Ação
7	Corrida Realista
8	Luta

idjogo	nomejogo	idTipo_jogo
1	River Raid	1
2	Mario Bros	2
3	F-Zero	3
4	Super Mario Bros	2
5	Wii Sports	4
6	FIFA12	5
7	God of War 3	6
8	Forza MotorSport	7
9	Gran Turismo 5	7
10	Street Fighter IV	8

13)

1 - Realize uma consulta que mostre o nome de cada jogo e o nome de seu respectivo tipo.

```
SELECT nomejogo, nome_tipo  
FROM jogo NATURAL JOIN tipo_jogo;
```

nomejogo	nome_tipo
River Raid	Tiro
Mario Bros	Aventura
Super Mario Bros	Aventura
F-Zero	Corrida
Wii Sports	Esportes variados
FIFA12	Futebol
God of War 3	Ação
Forza MotorSport	Corrida Realista
Gran Turismo 5	Corrida Realista
Street Fighter IV	Luta

2 - Realize uma consulta que além de mostrar o nome de cada tipo de jogo, mostre em uma coluna chamada “quantidadeJogos” que mostre o número de jogos de cada tipo.

```
SELECT nome_tipo, COUNT(idTipo_jogo) AS quantidadeJogos
FROM tipo_jogo NATURAL JOIN jogo
GROUP BY nome_tipo;
```

nome_tipo	quantidadeJogos
Tiro	1
Aventura	2
Corrida	1
Esportes variados	1
Futebol	1
Ação	1
Corrida Realista	2
Luta	1