

Banco de Dados



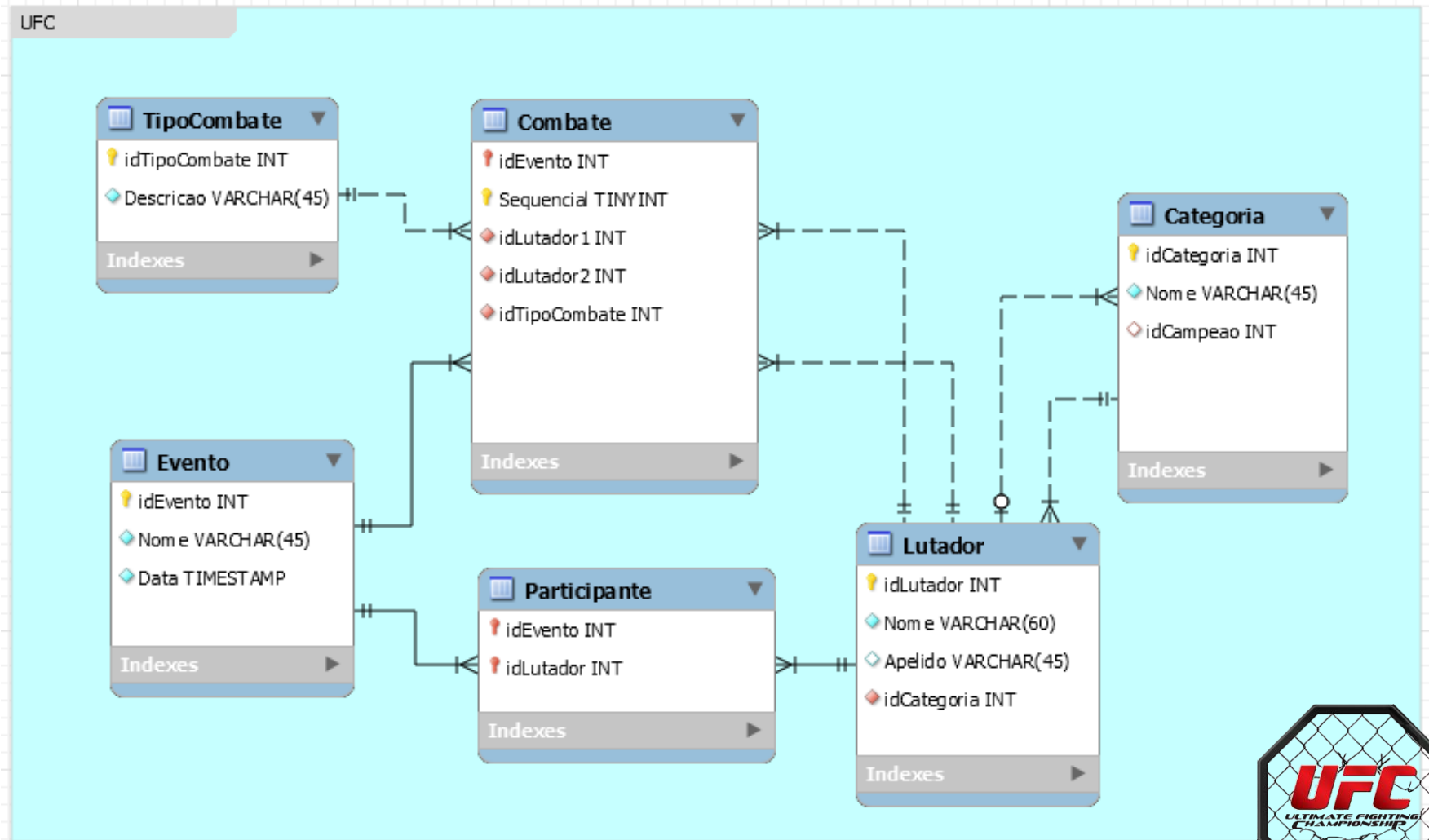
SQL – DML

Associação de Tabelas

***** Joins *****



MySQL – Data Definition Language





MySQL – Data Definition Language



```
CREATE TABLE Categoria (  
    idCategoria INT AUTO_INCREMENT,  
    Nome VARCHAR(45) NOT NULL,  
    idCampeao INT(11),  
    PRIMARY KEY (idCategoria)  
);  
  
CREATE TABLE TipoCombate (  
    idTipoCombate INT AUTO_INCREMENT,  
    Descricao VARCHAR(45) NOT NULL,  
    PRIMARY KEY (idTipoCombate)  
);  
  
CREATE TABLE Evento (  
    idEvento INT AUTO_INCREMENT,  
    Nome VARCHAR(45) NOT NULL,  
    Data TIMESTAMP NOT NULL,  
    PRIMARY KEY (idEvento)  
);  
  
CREATE TABLE Lutador (  
    idLutador INT AUTO_INCREMENT,  
    Nome VARCHAR(60) NOT NULL,  
    Apelido VARCHAR(45) NULL DEFAULT NULL,  
    idCategoria INT NOT NULL,  
    PRIMARY KEY (idLutador) ,  
    CONSTRAINT fkCategoria FOREIGN KEY (idCategoria) REFERENCES Categoria (idCategoria)  
        ON DELETE NO ACTION  
        ON UPDATE NO ACTION  
);
```



MySQL – Data Definition Language



```
ALTER TABLE Categoria
ADD CONSTRAINT fkLutador FOREIGN KEY (idCampeao) REFERENCES Lutador (idLutador)
ON DELETE SET NULL
ON UPDATE SET NULL;
```

```
CREATE TABLE Participante (
    idEvento INT NOT NULL,
    idLutador INT NOT NULL,
    PRIMARY KEY (idEvento, idLutador) ,
    CONSTRAINT fkEvento FOREIGN KEY (idEvento) REFERENCES Evento (idEvento)
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    CONSTRAINT fkLutadorEvento FOREIGN KEY (idLutador) REFERENCES Lutador (idLutador)
        ON DELETE RESTRICT
        ON UPDATE RESTRICT
);
```

```
CREATE TABLE Combate (
    idEvento INT NOT NULL,
    Sequencial TINYINT NOT NULL,
    idLutador1 INT NOT NULL,
    idLutador2 INT NOT NULL,
    idTipoCombate INT NOT NULL,
    PRIMARY KEY (idEvento, Sequencial) ,
    CONSTRAINT fkEventoCombate FOREIGN KEY (idEvento) REFERENCES Evento (idEvento)
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    CONSTRAINT fkLutador1 FOREIGN KEY (idLutador1) REFERENCES Lutador (idLutador),
    CONSTRAINT fkLutador2 FOREIGN KEY (idLutador2) REFERENCES Lutador (idLutador),
    CONSTRAINT fkTipoCombate FOREIGN KEY (idTipoCombate) REFERENCES TipoCombate (idTipoCombate)
);
```



MySQL – Data Definition Language



Aula 14 - Script SQL - Parte 2.sql



MySQL – Data Manipulation Language



```
SELECT * FROM evento;
```

	idEvento	Nome	Data
▶	1	UFC 194	2015-12-12 00:00:00
	2	UFC 187	2015-05-23 00:00:00
	3	UFC 168	2013-12-28 00:00:00
	4	UFC 175	2014-07-05 00:00:00
	5	UFC 152	2012-09-22 00:00:00
	6	UFC 140	2011-12-10 00:00:00
	7	UFC 165	2013-09-21 00:00:00
	8	UFC 179	2014-10-25 00:00:00
	9	UFC 193	2015-11-20 00:00:00
	10	UFC 190	2015-08-01 00:00:00
	11	UFC 126	2011-02-05 00:00:00

```
SELECT * FROM combate;
```

	idEvento	Sequencial	idLutador1	idLutador2	idTipoCombate
	1	1	6	7	1
	1	2	11	12	1
	2	1	12	13	1
	4	1	12	14	1
	4	2	25	23	1
	5	1	17	13	1
	5	2	1	2	1
	6	1	17	15	1
	7	1	17	18	1
	8	1	7	9	1
	10	1	25	26	1
	11	1	14	13	1
	11	2	9	27	2

Quais eventos não possuem combates vinculados?



MySQL – Data Manipulation Language



```
SELECT e.idEvento, e.Nome  
FROM evento e  
WHERE NOT EXISTS (SELECT * FROM combate c  
                  WHERE c.idEvento = e.idEvento);
```

	idEvento	Nome
	3	UFC 168
	9	UFC 193



MySQL – Data Definition Language



```
SELECT * FROM evento;
```

	idEvento	Nome	Data
▶	1	UFC 194	2015-12-12 00:00:00
	2	UFC 187	2015-05-23 00:00:00
	3	UFC 168	2013-12-28 00:00:00
	4	UFC 175	2014-07-05 00:00:00
	5	UFC 152	2012-09-22 00:00:00
	6	UFC 140	2011-12-10 00:00:00
	7	UFC 165	2013-09-21 00:00:00
	8	UFC 179	2014-10-25 00:00:00
	9	UFC 193	2015-11-20 00:00:00
	10	UFC 190	2015-08-01 00:00:00
	11	UFC 126	2011-02-05 00:00:00

```
SELECT * FROM combate;
```

	idEvento	Sequencial	idLutador1	idLutador2	idTipoCombate
	1	1	6	7	1
	1	2	11	12	1
	2	1	12	13	1
	4	1	12	14	1
	4	2	25	23	1
	5	1	17	13	1
	5	2	1	2	1
	6	1	17	15	1
	7	1	17	18	1
	8	1	7	9	1
	10	1	25	26	1
	11	1	14	13	1
	11	2	9	27	2

Quais eventos que possuem combates vinculados?



MySQL – Data Manipulation Language



```
SELECT e.idEvento, e.Nome  
FROM evento e  
WHERE EXISTS (SELECT * FROM combate c  
              WHERE c.idEvento = e.idEvento);
```

	idEvento	Nome
	1	UFC 194
	2	UFC 187
	4	UFC 175
	5	UFC 152
	6	UFC 140
	7	UFC 165
	8	UFC 179
	10	UFC 190
	11	UFC 126

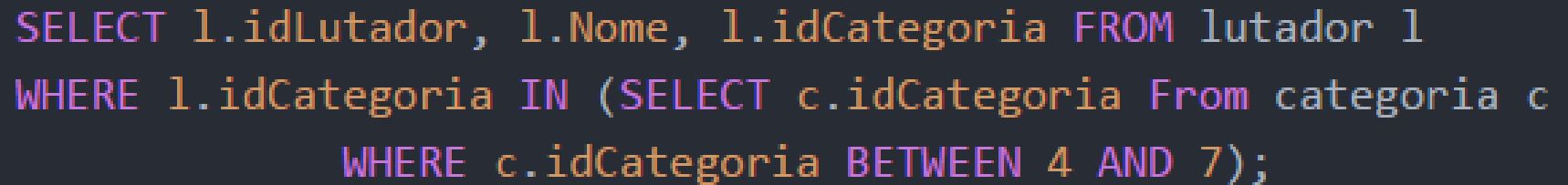


MySQL – Data Manipulation Language



Imagine que você desconhece as categorias cadastradas entre as categorias registradas com código entre 4 e 7, mas precise identificar todos os lutadores que pertencem a essas categorias.

O que fazer?

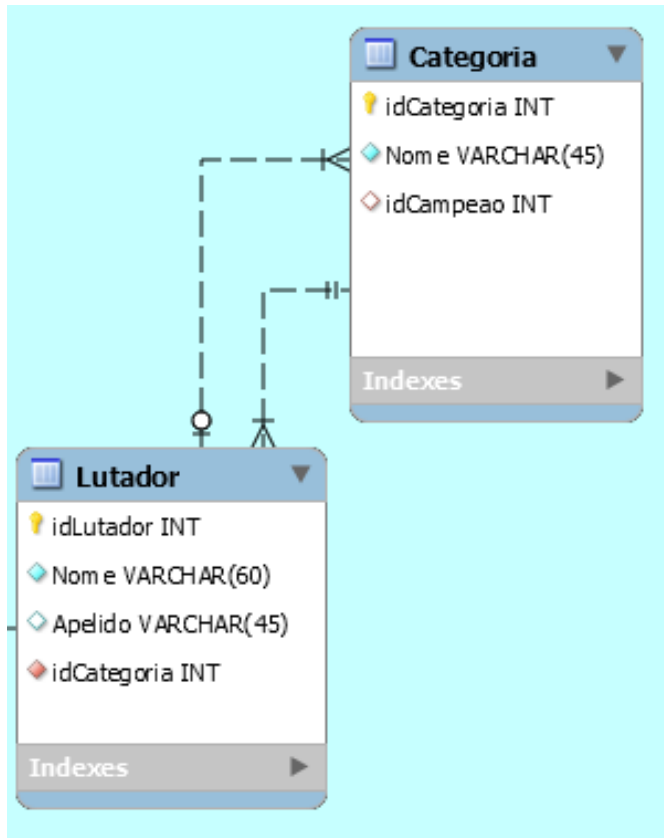




MySQL – Data Manipulation Language



```
SELECT c.idCategoria, c.Nome, c.idCampeao
FROM categoria c
ORDER BY c.idCategoria;
```



	idCategoria	Nome	idCampeao
	1	Mosca	1
	2	Peso Galo	4
	3	Peso Pena	6
	4	Peso Leve	10
	5	Peso Meio-Médio	NULL
	6	Peso Médio	11
	7	Peso Meio-Pesado	17
	8	Peso Pesado	19
	9	Peso Palha - Feminino	NULL
	10	Peso Galo - Feminino	23



MySQL – Data Manipulation Language



```
SELECT c.idCategoria, c.Nome, c.idCampeao, l.Nome
FROM categoria c
INNER JOIN lutador l ON l.idLutador = c.idCampeao
ORDER BY c.idCategoria;
```

	idCategoria	Nome	idCampeao
	1	Mosca	1
	2	Peso Galo	4
	3	Peso Pena	6
	4	Peso Leve	10
	5	Peso Meio-Médio	NULO
	6	Peso Médio	11
	7	Peso Meio-Pesado	17
	8	Peso Pesado	19
	9	Peso Palha - Feminino	NULO
	10	Peso Galo - Feminino	23

	idCategoria	Nome	idCampeao	Nome
	1	Mosca	1	Demetrious Johnson
	2	Peso Galo	4	Dominick Cruz
	3	Peso Pena	6	Conor McGregor
	4	Peso Leve	10	Rafael Dos Anjos
	6	Peso Médio	11	Luke Rockhold
	7	Peso Meio-Pesado	17	Jon Jones
	8	Peso Pesado	19	Fabricio Werdum
▶	10	Peso Galo - Feminino	23	Miesha Tate



MySQL – Data Manipulation Language



```
SELECT c.idCategoria, c.Nome, c.idCampeao, l.Nome
FROM categoria c
LEFT JOIN lutador l ON l.idLutador = c.idCampeao
ORDER BY c.idCategoria;
```

	idCategoria	Nome	idCampeao
	1	Mosca	1
	2	Peso Galo	4
	3	Peso Pena	6
	4	Peso Leve	10
	5	Peso Meio-Médio	NULL
	6	Peso Médio	11
	7	Peso Meio-Pesado	17
	8	Peso Pesado	19
	9	Peso Palha - Feminino	NULL
	10	Peso Galo - Feminino	23

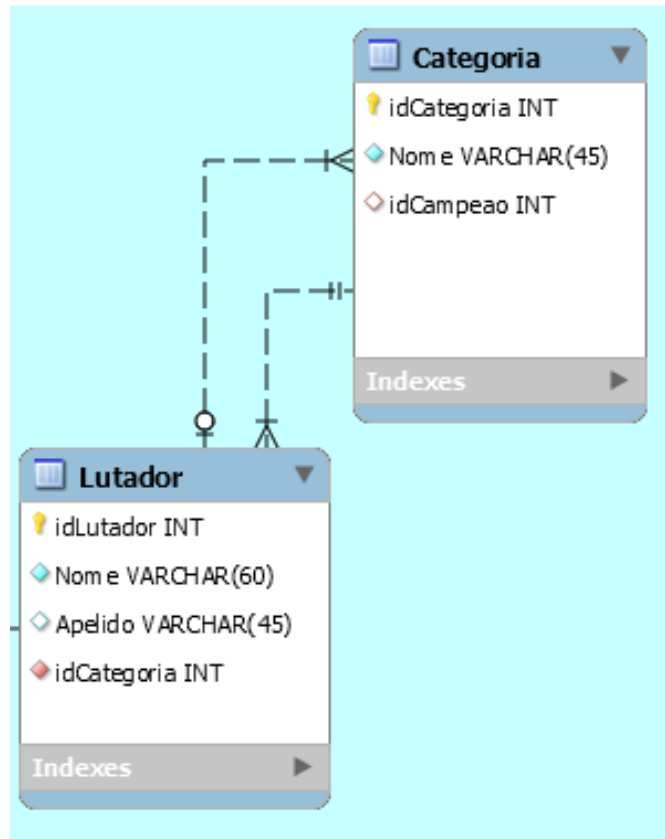
	idCategoria	Nome	idCampeao	Nome
	1	Mosca	1	Demetrious Johnson
	2	Peso Galo	4	Dominick Cruz
	3	Peso Pena	6	Conor McGregor
	4	Peso Leve	10	Rafael Dos Anjos
	5	Peso Meio-Médio	NULL	NULL
	6	Peso Médio	11	Luke Rockhold
	7	Peso Meio-Pesado	17	Jon Jones
	8	Peso Pesado	19	Fabricio Werdum
	9	Peso Palha - Feminino	NULL	NULL
▶	10	Peso Galo - Feminino	23	Miesha Tate



MySQL – Data Manipulation Language



```
SELECT l.idLutador, l.Nome, l.idCategoria, c.Nome  
FROM lutador l  
INNER JOIN categoria c ON c.idCategoria = l.idCategoria;
```



	idLutador	Nome	idCategoria	Nome
	1	Demetrious Johnson	1	Mosca
	2	Joseph Benavidez	1	Mosca
	3	Henry Cejudo	1	Mosca
	4	Dominick Cruz	2	Peso Galo
	5	Renan Barão	2	Peso Galo
	6	Conor McGregor	3	Peso Pena
	7	José Aldo	3	Peso Pena
	8	Frankie Edgar	3	Peso Pena
	9	Chad Mendes	3	Peso Pena
	27	Michihiro Omigawa	3	Peso Pena
	10	Rafael Dos Anjos	4	Peso Leve
	11	Luke Rockhold	6	Peso Médio



MySQL – Data Manipulation Language



```
SELECT e.idEvento, e.Nome, p.idLutador, l.Nome
FROM Evento e
INNER JOIN Participante p ON p.idEvento = e.idEvento
INNER JOIN Lutador l ON l.idLutador = p.idLutador
WHERE e.idEvento = 1;
```

	idEvento	Nome	idLutador	Nome
	1	UFC 194	6	Conor McGregor
	1	UFC 194	7	José Aldo
	1	UFC 194	11	Luke Rockhold
▶	1	UFC 194	12	Chris Weidman



MySQL – Data Manipulation Language



```
✓ SELECT e.idEvento, e.Nome, c.idLutador1, l1.Nome,  
        c.idLutador2, l2.Nome, c.idTipoCombate, tc.Descricao  
FROM Evento e  
INNER JOIN combate c ON c.idEvento = e.idEvento  
INNER JOIN lutador l1 ON l1.idLutador = c.idLutador1  
INNER JOIN lutador l2 ON l2.idLutador = c.idLutador2  
INNER JOIN tipoCombate tc ON tc.idTipoCombate = c.idTipoCombate  
WHERE e.idEvento = 11;
```

	idEvento	Nome	idLutador1	Nome	idLutador2	Nome	idTipoCombate	Descricao
▶	11	UFC 126	14	Anderson Silva	13	Vitor Belfort	1	Card Principal
	11	UFC 126	9	Chad Mendes	27	Michihiro Omigawa	2	Card Preliminar



MySQL – Data Manipulation Language



```
SELECT l.idCategoria, c.Nome, COUNT(l.idCategoria)
FROM lutador l
INNER JOIN categoria c ON c.idCategoria = l.idCategoria
GROUP BY l.idCategoria
HAVING COUNT(l.idCategoria) > 2
ORDER BY COUNT(l.idCategoria) DESC;
```

	idCategoria	Nome	COUNT(l.idCategoria)
	3	Peso Pena	5
	6	Peso Médio	5
	8	Peso Pesado	4
	10	Peso Galo - Feminino	4
	1	Mosca	3
▶	7	Peso Meio-Pesado	3