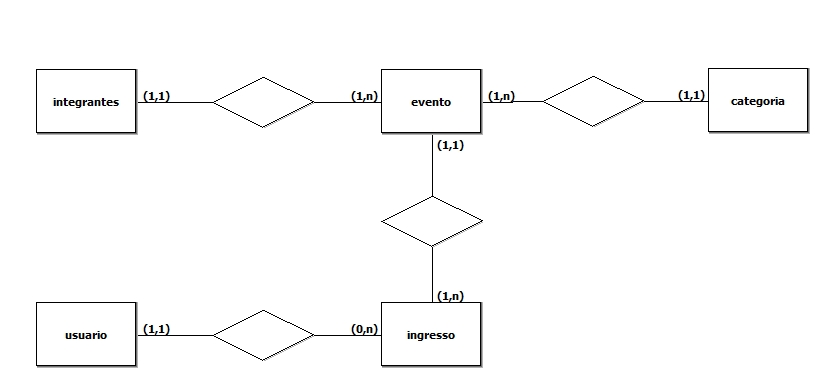
**STORED PROCEDURE – INSERÇÃO / EDIÇÃO – EXERCÍCIOS**

****

**Um evento (peça, espetáculo ou algo assim) poderá ter 1 ou vários integrantes, ou seja, um ou vários profissionais na apresentação/preparação e esse mesmo evento poderá ser classificado em uma ou várias categorias: como comédia, terror, musical...**

**Já o ingresso será vinculado a um evento e a um usuário. Resumindo, o usuário poderá adquirir um ingresso para um evento.**

1) Crie o BD -> create database bdteatro e as tables conforme DER. No mínimo 4 campos por table (isso inclui pk e fk)

2) Após criar as tabelas crie 4 procedures para inserção/edição para as tabelas evento, ingresso, usuario e categoria

3) Faça as chamadas de cada procedure

4) Em uma das chamadas, implemente em um método em qualquer linguagem de programação

**Poste esse documento salvo como pdf e com as respostas**

**1)CRIAR BD**

CREATE TABLE usuario(

id\_usu INT PRIMARY KEY AUTO\_INCREMENT,

cpf\_usu VARCHAR(11) UNIQUE,

nome\_usu VARCHAR(50),

dataNasc\_usu DATE

);

CREATE TABLE integrante(

id\_int INT PRIMARY KEY AUTO\_INCREMENT,

personagem\_int VARCHAR(50),

nome\_int VARCHAR(50),

data\_int DATE,

cpf\_int VARCHAR(11)

);

CREATE TABLE categoria(

id\_cat INT PRIMARY KEY AUTO\_INCREMENT,

nome\_cat VARCHAR(80),

cor\_cat VARCHAR(15),

descricao\_cat VARCHAR(200)

);

CREATE TABLE evento(

id\_eve INT PRIMARY KEY AUTO\_INCREMENT,

nome\_eve VARCHAR(50),

data\_eve DATE,

local\_eve VARCHAR(60),

limite\_eve INT CHECK(limite\_eve >= 0),

id\_cat INT NOT NULL,

id\_int INT NOT NULL,

foreign key (id\_cat) references categoria(id\_cat),

foreign key (id\_int) references integrante(id\_int)

);

CREATE TABLE ingresso(

id\_ing INT,

tipo\_ing VARCHAR(50),

preco\_ing DOUBLE(10,2),

id\_usu INT NOT NULL,

id\_eve INT NOT NULL,

foreign key (id\_usu) references usuario(id\_usu),

foreign key (id\_eve) references evento(id\_eve)

);

**2)Procedures para inserção**

DELIMITER &&

CREATE PROCEDURE salvar\_usuario (IN var\_id INT,

var\_cpf VARCHAR(11),

var\_nome VARCHAR(50),

var\_dataNasc DATE)

BEGIN

IF (EXISTS(SELECT id\_usu FROM usuario WHERE id\_usu = var\_id)) THEN

UPDATE usuario SET cpf\_usu = var\_cpf, nome\_usu = var\_nome, dataNasc\_usu = var\_dataNasc WHERE id\_usu = var\_id;

ELSE

INSERT INTO usuario VALUES (null, var\_cpf, var\_nome, var\_dataNasc);

END IF;

END

&&

CREATE PROCEDURE salvar\_integrante (IN var\_id INT,

var\_personagem VARCHAR(50),

var\_nome VARCHAR(50),

var\_data DATE,

var\_cpf VARCHAR(11))

BEGIN

IF (EXISTS(SELECT id\_int FROM integrante WHERE id\_int = var\_id)) THEN

UPDATE integrante SET personagem\_int = var\_personagem, nome\_int = var\_nome, data\_int = var\_data, cpf\_int = var\_cpf WHERE id\_int = var\_id;

ELSE

INSERT INTO integrante VALUES (null, var\_personagem, var\_nome, var\_data, var\_cpf);

END IF;

END

&&

CREATE PROCEDURE salvar\_categoria (IN var\_id INT,

var\_nome VARCHAR(50),

var\_cor VARCHAR(15),

var\_descricao VARCHAR(200))

BEGIN

IF (EXISTS(SELECT id\_cat FROM categoria WHERE id\_cat = var\_id)) THEN

UPDATE categoria SET nome\_cat = var\_nome, cor\_cat = var\_cor, descricao\_cat = var\_descricao WHERE id\_cat = var\_id;

ELSE

INSERT INTO categoria VALUES (null, var\_nome, var\_cor, var\_descricao);

END IF;

END

&&

CREATE PROCEDURE salvar\_evento (IN var\_id INT,

var\_nome VARCHAR(50),

var\_data DATE,

var\_local VARCHAR(60),

var\_limite INT,

var\_id\_cat INT,

var\_id\_int INT)

BEGIN

IF (EXISTS(SELECT id\_eve FROM evento WHERE id\_eve = var\_id)) THEN

UPDATE evento SET nome\_eve = var\_nome, data\_eve = var\_data, local\_eve = var\_local,

limite\_eve = var\_limite, id\_cat = var\_id\_cat, id\_int = var\_id\_int WHERE id\_eve = var\_id;

ELSE

INSERT INTO evento VALUES (null, var\_nome, var\_data,var\_local, var\_limite, var\_id\_cat, var\_id\_int);

END IF;

END

&&

CREATE PROCEDURE salvar\_ingresso (IN var\_id INT,

var\_tipo VARCHAR(50),

var\_preco DOUBLE(10,2),

var\_id\_usu INT,

var\_id\_eve INT)

BEGIN

IF (EXISTS(SELECT id\_ing FROM ingresso WHERE id\_ing = var\_id)) THEN

UPDATE ingresso SET tipo\_ing = var\_tipo, preco\_ing = var\_preco, id\_usu = var\_id\_usu, id\_eve = var\_id\_eve WHERE id\_ing = var\_id;

ELSE

INSERT INTO ingresso VALUES (null, var\_tipo, var\_preco, var\_id\_usu, var\_id\_eve);

END IF;

END&&

DELIMITER ;

**3) Chamada de cada procedure**

--Salvar

CALL salvar\_usuario(null, '12345678901', 'Alice', '17/08/2001');

CALL salvar\_integrante(null, 'Arvore', 'Maria', '04/05/2023' , '12345678906');

CALL salvar\_categoria(null, 'Terror', 'preto', null);

CALL salvar\_evento(null,'A calça do terror','04/05/2023','Indaiaclube',68,1,1);

CALL salvar\_ingresso(null, 'Meia', 85.00, 1,1);

--Editar

CALL salvar\_usuario(1, '12345678901', 'Alice', '17/08/2005');

--Deletar procdedure

--DROP PROCEDURE salvar\_evento;

--DROP PROCEDURE salvar\_ingresso;

--DROP PROCEDURE salvar\_integrante;

--DROP PROCEDURE salvar\_usuario;

--DROP PROCEDURE salvar\_categoria;

**4) Implementar em uma linguagem**