

11208012 Caio Rodrigues
11208030 Felipe Furquim
10843552 Stefany Ramos
10723836 Renan Nakazawa

ARTEFATO A (DUAS REGRAS DE NEGÓCIOS):

- **Trigger 1:** Toda vez que uma avaliação de usuário é adicionada, a nota média dada pelo nosso site é recalculada.
- **Solução SQL padrão:**

```
CREATE TRIGGER obra_nota_media AFTER INSERT, DELETE ON avaliacao  
  
BEGIN  
    UPDATE obra  
    SET nota_media = (  
        SELECT COALESCE(avg(nota), 0)  
        FROM avaliacao a WHERE obra_id = 1  
    )  
    WHERE obra.id = new.obra_id;  
  
END;
```

- **Solução Postgres:**

```
CREATE FUNCTION obra_nota_media() RETURNS TRIGGER AS $$  
BEGIN  
    UPDATE obra  
    SET nota_media = (  
        SELECT COALESCE(avg(nota), 0)  
        FROM avaliacao a WHERE obra_id = 1  
    )  
    WHERE obra.id = new.obra_id;  
    RETURN NULL;  
END;  
$$ LANGUAGE plpgsql;  
  
CREATE TRIGGER trigger_obra_nota_media  
AFTER INSERT  
ON avaliacao  
FOR EACH ROW  
EXECUTE PROCEDURE obra_nota_media();
```

(*nota: esse mesmo trigger, porém agora para deleção, também foi implementado*)

- **Transcrição do código:**

Criando o trigger “obra_nota_media”, que é ativado depois de qualquer inserção na tabela “avaliacao”. São declaradas duas variáveis numéricas: “contador” e “divisor”. Some o valor de todas as notas de uma certa obra em “contador”, depois conte quantas avaliações distintas da mesma obra em “divisor”. Divida o contador pelo divisor, ou seja, a nota total dividida pela quantidade de notas, e atribua à obra em sua entrada na tabela “obra” em seu campo “nota_media”.

- **Casos de Teste:**

- **Caso 1:**

Suponha que no banco de dados já estejam cadastrados dois usuários e um filme. Queremos adicionar a primeira avaliação do filme presente do SGBD.

tabela obra

id [PK] integer	estudio_id integer	diretores_id integer	pais_id integer	nome text	descricao text	data_lancamento date	nota_media numeric (2)
1	1	1	1	Pirarucu	[null]	[null]	0

```
INSERT INTO avaliacao (id, usuario_id, obra_id, comentario, nota)
VALUES (5, 1, 1, "Peixe", 10);
```

tabela avaliacao

id [PK] integer	usuario_id integer	obra_id integer	comentario text	nota integer	data_avaliacao date
5	1	1	Peixe	10	

Ao adicionar a primeira avaliação desta obra, o trigger “obra_nota_media” será acionado e o campo “nota_media” da tabela “obra” será atualizado com a nota do usuário.

tabela obra

id [PK] integer	estudio_id integer	diretores_id integer	pais_id integer	nome text	descricao text	data_lancamento date	nota_media numeric (2)
1	1	1	1	Pirarucu	[null]	[null]	10

- **Caso 2:**

Suponha que no banco de dados já estejam cadastrados dois usuários e um filme. Considere também que o filme já tem 2 avaliações cadastradas, uma avaliação com nota 3 e outra com nota 5.

tabela obra

id [PK] integer	estudio_id integer	diretores_id integer	pais_id integer	nome text	descricao text	data_lancamento date	nota_media numeric (2)
1	1	1	1	Click	[null]	[null]	4

tabela filme

obra_id integer	duracao integer
1	120

tabela avaliacao

id [PK] integer	usuario_id integer	obra_id integer	comentario text	nota integer	data_avaliacao date
1	1	1	mais ou menos	3	2021-06-24
3	1	1	gostei	5	2021-06-06

Quando o usuário inserir uma nova avaliação no banco de dados, a trigger “obra_nota_media” será ativada:

```
INSERT INTO avaliacao (id, usuario_id, obra_id, comentario, nota)
VALUES (5, 2, 1, "Filme cringe, não gostei", 2);
```

Dessa forma, o trigger irá recalculer a nota média do filme e a tabela obra será modificada:

tabela obra

id [PK] integer	estudio_id integer	diretores_id integer	pais_id integer	nome text	descricao text	data_lancamento date	nota_media numeric (2)
1	1	1	1	Click	[null]	[null]	3

■ Caso 3:

Considerando o cenário anterior, inserir diversas avaliações de uma vez também irá ativar a trigger que irá recalculer a nota média normalmente e modificar o campo nota_media da tabela obra:

```
INSERT INTO avaliacao (id, usuario_id, obra_id, comentario, nota) VALUES
(42, 1, 1, 'Muito bomm', 10),
(56, 2, 1, 'Amei!!', 8);
```

tabela obra

id [PK] integer	estudio_id integer	diretores_id integer	pais_id integer	nome text	descricao text	data_lancamento date	nota_media numeric (2)
1	1	1	1	Click	[null]	[null]	7

- **Trigger 2:** Caso um usuário seja deletado, delete todas as suas avaliações de usuário, e recalculer a nota de todos os filmes afetados.
- Solução SQL padrão:

```
CREATE TRIGGER trigger_deletar_avaliacoes_usuario
AFTER UPDATE ON usuario
FOR EACH ROW
WHEN (NEW.deletado = TRUE)
BEGIN
```

```

DELETE FROM avaliacao a
WHERE a.usuario_id = NEW.id;
END;

```

- Solução Postgres:

```

CREATE FUNCTION deletar_avaliacoes_usuario()
RETURNS TRIGGER
LANGUAGE PLPGSQL
AS $$
BEGIN
DELETE FROM avaliacao a
WHERE a.usuario_id = NEW.id;
END;
RETURN NULL;

CREATE TRIGGER trigger_deletar_avaliacoes_usuario
AFTER UPDATE ON usuario
FOR EACH ROW
WHEN (NEW.deletado = TRUE)
EXECUTE PROCEDURE deletar_avaliacoes_usuario()
deletar_avaliacoes_usuario()

```

- Transcrição do código:

Criando o trigger “deletar_avaliacoes_usuario”, que é ativado toda vez que uma linha é deletada. Dado o ID da linha “deletada” (não é verdadeiramente deletada) através de NEW, vá na tabela “avaliacao”, e remova todas as entradas que possuam esse ID. É esperado que o outro trigger deste exercício também aconteça quando estas entradas forem derrubadas.

- Casos de Teste:

- caso 1:

Considerando o cenário do trigger passado, suponha que o usuário com id=2 deseja excluir sua conta:

tabela usuario

id [PK] integer	nickname text	email text	data_nascimento date	deletado boolean
1	Roberto	cavalo@gmail.com	2000-10-03	false
2	clebinho	clebinho.bolo@hotmail.com	1999-05-05	false

tabela avaliacao

id [PK] integer	usuario_id integer	obra_id integer	comentario text	nota integer	data_avaliacao date
1	1	1	mais ou menos	3	2021-06-24
3	1	1	gostei	5	2021-06-06
5	2	1	Filme cringe, não gostei	2	[null]
42	1	1	Muito bom	10	[null]
56	2	1	Amei	8	[null]

```
DELETE FROM usuario
WHERE usuario.id = 2
```

Com essa deleção, o trigger “trigger_deletar_avaliacoes_usuario” será acionado e todas as avaliações deste usuário serão deletadas da tabela “avaliacao”:

Agora, como avaliações foram deletadas, isso irá acionar o trigger “obra_nota_media” que vai recalcular a nota média das obras afetadas, nesse caso, apenas um filme foi cadastrado.

tabela avaliacao

id [PK] integer	usuario_id integer	obra_id integer	comentario text	nota integer	data_avaliacao date
1	1	1	mais ou menos	3	2021-06-24
3	1	1	gostei	5	2021-06-06
42	1	1	Muito bom	10	[null]