

Discentes: Celio Slomp e Matheus Marchi Moro  
Respostas da AV3-C

Tabelas utilizadas (MYSQL):

```
-- Criar tabela Aluno_log
CREATE TABLE Aluno_log(
  matricula integer,
  cpf integer,
  nome varchar(255),
  email varchar(255),
  numContato varchar(255),
  versao integer
);

-- Criar tabela Aluno
CREATE TABLE Aluno(
  matricula integer,
  cpf integer,
  nome varchar(255),
  email varchar(255),
  numContato varchar(255)
);
```

Trigger de UPDATE:

```
create trigger AtualizaAluno before update on Aluno for each row
begin
  set @versao = coalesce((select max(a.versao) from Aluno_log a where a.matricula=new.matricula), 0) + 1;
  insert into Aluno_log (matricula, cpf, nome, email, numContato, versao)
  values (old.matricula, old.cpf, old.nome, old.email, old.numContato, @versao);
end;
```

Trigger de DELETE:

```
-- Cria trigger de delete
create trigger ExcluiAluno before delete on Aluno for each row
begin
  set @versao = coalesce((select max(a.versao) from Aluno_log a where a.matricula=old.matricula), 0) + 1;
  insert into Aluno_log (matricula, cpf, nome, email, numContato, versao)
  values (old.matricula, old.cpf, old.nome, old.email, old.numContato, @versao);
end;
```

Procedure para usar como se fosse INSERT:

```
create procedure InserirAluno(
  -- Parâmetros
  IN prMatricula INTEGER,
  IN prCpf INTEGER,
  IN prNome VARCHAR(255),
  IN prEmail VARCHAR(255),
  IN prNumContato VARCHAR(255)
)
begin
  -- Corpo da procedure
  -- Deleta a versão antiga de Aluno
  delete from Aluno a where a.matricula = prMatricula;

  -- Insere a versão nova em Aluno
  insert into Aluno (matricula, cpf, nome, email, numContato)
  values (prMatricula, prCpf, prNome, prEmail, prNumContato);
end;
```

Por que usamos uma procedure para o INSERT ao invés de uma trigger? É porque as triggers (pelo menos no MYSQL) não podem alterar os registros da tabela em que estão vinculadas. No nosso caso, uma trigger de INSERT não iria conseguir excluir um registro da tabela Aluno para sobrescrevê-lo. Então, pensamos em criar uma procedure que o usuário utilizaria para fazer as inserções ao invés de usar diretamente o comando de inserção.

## TESTES:

Todos os testes foram feitos com os seguintes dados:

```
insert into Aluno (matricula, cpf, nome, email, numContato)
values
(1, 111111111, 'AAA', 'aaa@email.com', 12345),
(2, 222222222, 'BBB', 'bbb@email.com', 12345),
(3, 333333333, 'CCC', 'ccc@email.com', 12345);
```

OBS.: Os mesmos testes podem ser repetidos para a visualização do incremento do campo “versao” dos registros.

### Teste da trigger de UPDATE

Registros da tabela Aluno antes de efetuar o UPDATE (Aluno\_log está vazia neste momento):

123 matricula	123 cpf	ABC nome	ABC email	ABC numContato
1	111.111.111	AAA	aaa@email.com	12345
2	222.222.222	BBB	bbb@email.com	12345
3	333.333.333	CCC	ccc@email.com	12345

Comando de UPDATE:

```
update Aluno
set matricula = 1, cpf = 111111111, nome = 'JOÃO', email = 'joao@email.com', numContato = 56789
where matricula = 1
```

Registros da tabela Aluno depois de efetuar o UPDATE:

123 matricula	123 cpf	ABC nome	ABC email	ABC numContato
1	111.111.111	JOÃO	joao@email.com	56789
2	222.222.222	BBB	bbb@email.com	12345
3	333.333.333	CCC	ccc@email.com	12345

Registros da tabela Aluno\_log depois de efetuar o UPDATE:

123 matricula	123 cpf	ABC nome	ABC email	ABC numContato	123 versao
1	111.111.111	AAA	aaa@email.com	12345	1

## Teste da trigger de DELETE

Registros da tabela Aluno antes de efetuar o DELETE (Aluno\_log está vazia neste momento):

123 matricula	123 cpf	ABC nome	ABC email	ABC numContato
1	111.111.111	AAA	aaa@email.com	12345
2	222.222.222	BBB	bbb@email.com	12345
3	333.333.333	CCC	ccc@email.com	12345

Comando de DELETE:

```
delete from Aluno
where matricula = 1;
```

Registros da tabela Aluno depois de efetuar o DELETE:

123 matricula	123 cpf	ABC nome	ABC email	ABC numContato
2	222.222.222	BBB	bbb@email.com	12345
3	333.333.333	CCC	ccc@email.com	12345

Registros da tabela Aluno\_log depois de efetuar o DELETE:

123 matricula	123 cpf	ABC nome	ABC email	ABC numContato	123 versao
1	111.111.111	AAA	aaa@email.com	12345	1

## Teste da procedure de INSERT

Registros da tabela Aluno antes de efetuar o INSERT (Aluno\_log está vazia neste momento):

123 matricula	123 cpf	ABC nome	ABC email	ABC numContato
1	111.111.111	AAA	aaa@email.com	12345
2	222.222.222	BBB	bbb@email.com	12345
3	333.333.333	CCC	ccc@email.com	12345

Comando de INSERT (chamando a procedure)

```
call InserirAluno(1, 123456789, 'JOÃO', 'joao@email.com', 56789);
```

Registros da tabela Aluno depois de efetuar o INSERT:

123 matricula	123 cpf	ABC nome	ABC email	ABC numContato
2	222.222.222	BBB	bbb@email.com	12345
3	333.333.333	CCC	ccc@email.com	12345
1	123.456.789	JOÃO	joao@email.com	56789

Registros da tabela Aluno\_log depois de efetuar o INSERT:

123 matricula	123 cpf	ABC nome	ABC email	ABC numContato	123 versao
1	111.111.111	AAA	aaa@email.com	12345	1