



# Fundamentos de Bacos de Dados

Alter table

# O comando ALTER

- A definição de uma tabela ou de outros elementos de esquema nomeados podem ser alterada usando o comando ALTER.
- Para as tabelas, as possíveis **ações de alteração de tabela** incluem acrescentar ou remover uma coluna (atributo), alterar uma definição de coluna e acrescentar ou remover restrições de tabela.
- A sintaxe SQL usada é:

```
ALTER TABLE tbl_name [alter_specification  
                        [, alter_specification] ...]
```

# O comando ALTER

- *A sintaxe SQL usada é:*

```
ALTER TABLE tbl_name [alter_specification  
                        [, alter_specification] ...]
```

- *Algumas especificações mais importantes do alter são:*

**alter\_specification:**

- | ADD
- | ALTER
- | CHANGE
- | MODIFY
- | DROP
- | RENAME [TO|AS]

# ALTER+ADD

- *Acréscendo uma coluna (atributo)*

- *Sintaxe:*

**ALTER TABLE** <nome tabela> **ADD COLUMN** <nome coluna> <tipo coluna>

- *Exemplo:*

*Tabela aluno antes*

	matricula	nome	tipo_aluno	curso
--	-----------	------	------------	-------

# ALTER+ADD

- *Acréscendo uma coluna (atributo)*

- *Sintaxe:*

**ALTER TABLE** <nome tabela> **ADD COLUMN** <nome coluna> <tipo coluna>

- *Exemplo:*

*Tabela aluno antes*

	matricula	nome	tipo_aluno	curso
--	-----------	------	------------	-------

```
ALTER TABLE aluno ADD COLUMN endereco VARCHAR(50);
```

# ALTER+ADD

- *Acréscendo uma coluna (atributo)*

```
ALTER TABLE aluno ADD COLUMN endereco VARCHAR(50);
```

The screenshot shows the MySQL Workbench interface. The top menu bar includes File, Edit, View, Query, Database, Server, Tools, Scripting, and Help. The left sidebar shows the 'Schemas' tab with a list of databases: empresa, information\_schema, performance\_schema, sakila, test, and universidade. The 'universidade' database is selected, showing its tables, views, stored procedures, and functions. The main editor window displays the SQL statement: `ALTER TABLE aluno ADD COLUMN endereco VARCHAR(50);`. The bottom panel shows the 'Output' tab with a table of execution results.

	Time	Action	Message	Duration / Fetch
✓	1 14:10:44	ALTER TABLE aluno ADD COLUMN endereco VARCHAR(50)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.515 sec

# ALTER+ADD

- *Acrescentando uma coluna (atributo)*

```
ALTER TABLE aluno ADD COLUMN endereco VARCHAR(50);
```

- *Exemplo:*

*Tabela aluno depois*

	matricula	nome	tipo_aluno	curso	endereco
--	-----------	------	------------	-------	----------

# ALTER+ADD

- *Também é possível alterar as restrições especificadas sobre uma tabela ao acrescentar ou remover uma restrição nomeada.*
- *A acrescentando uma coluna (atributo)*
  - *Sintaxe primary key:*

**ALTER TABLE** <nome tabela> **ADD CONSTRAINT**<nome constraint>  
**PRIMARY KEY** (<nome coluna>);

- *Exemplo:*

```
ALTER TABLE aluno ADD CONSTRAINT pk_aluno  
PRIMARY KEY(matricula);
```



# ALTER+ADD

- *Acréscendo uma coluna (atributo)*
  - *Sintaxe foreign key:*

```
ALTER TABLE <nome tabela> ADD CONSTRAINT<nome constraint>  
FOREIGN KEY (<nome coluna>)REFERENCES <nome tabela>(<nome coluna>)
```

- *Exemplo:*

---

```
ALTER TABLE aluno ADD CONSTRAINT fk_aluno  
FOREIGN KEY(curso) REFERENCES curso(codigo);
```

# ALTER+CHANGE

- *Alterando o tipo e nome de uma coluna (atributo)*

- *Sintaxe:*

```
ALTER TABLE tbl_name CHANGE [COLUMN] old_col_name new_col_name column_definition
```

- *Exemplo:*

**Table:** *aluno*

**Columns:**

<u>matricula</u>	decimal(4,0) PK
nome	varchar(30)
tipo_aluno	int(11)
curso	varchar(3)
endereco	varchar(50)

```
ALTER TABLE aluno CHANGE COLUMN endereco rua VARCHAR(30);
```

# ALTER+CHANGE

```
ALTER TABLE aluno CHANGE COLUMN endereco rua VARCHAR(30);
```

Table: aluno

Columns:

<u>matricula</u>	decimal(4,0) PK
nome	varchar(30)
tipo_aluno	int(11)
curso	varchar(3)
rua	varchar(30)

# ALTER+CHANGE

- *Alterando somente o tipo de uma coluna (atributo)*

- *Sintaxe:*

```
ALTER TABLE tbl_name CHANGE [COLUMN] old_col_name new_col_name column_definition
```

- *Exemplo:*

**Table:** *aluno*

**Columns:**

<u>matricula</u>	decimal(4,0) PK
nome	varchar(30)
tipo_aluno	int(11)
curso	varchar(3)
endereco	varchar(50)

```
ALTER TABLE aluno CHANGE COLUMN tipo_aluno tipo_aluno VARCHAR(40);
```

# ALTER+CHANGE

```
ALTER TABLE aluno CHANGE COLUMN tipo_aluno tipo_aluno VARCHAR(40);
```

Table: **aluno**

Columns:

<u>matricula</u>	decimal(4,0) PK
nome	varchar(30)
tipo_aluno	varchar(40)
curso	varchar(3)
rua	varchar(30)

# ALTER+CHANGE

- Alterando somente o nome de uma coluna (atributo)

- Sintaxe:

```
ALTER TABLE tbl_name CHANGE [COLUMN] old_col_name new_col_name column_definition
```

- Exemplo:

**Table:** aluno

**Columns:**

<u>matricula</u>	decimal(4,0) PK
nome	varchar(30)
tipo_aluno	varchar(40)
curso	varchar(3)
rua	varchar(30)

```
ALTER TABLE aluno CHANGE COLUMN tipo_aluno tp_aluno VARCHAR(40);
```

# ALTER+CHANGE

```
ALTER TABLE aluno CHANGE COLUMN tipo_aluno tp_aluno VARCHAR(40);
```

Table: **aluno**

Columns:

<u>matricula</u>	decimal(4,0) PK
nome	varchar(30)
tp_aluno	varchar(40)
curso	varchar(3)
rua	varchar(30)

# ALTER+MODIFY

- *Modificando uma coluna (atributo)*

- *Sintaxe:*

```
ALTER TABLE tbl_name MODIFY [COLUMN] col_name column_definition
```

- *Exemplo:*

**Table:** **aluno**

**Columns:**

<u>matricula</u>	decimal(4,0) PK
nome	varchar(30)
tp_aluno	varchar(40)
curso	varchar(3)
rua	varchar(30)

```
ALTER TABLE aluno MODIFY COLUMN tp_aluno INT(1);
```



# ALTER+MODIFY

```
ALTER TABLE aluno MODIFY COLUMN tp_aluno INT(1);
```

Table: **aluno**

Columns:

<u>matricula</u>	decimal(4,0) PK
nome	varchar(30)
tp_aluno	int(1)
curso	varchar(3)
rua	varchar(30)

# ALTER+DROP COLUMN

- *Removendo uma coluna (atributo)*
  - *Sintaxe:*

**ALTER TABLE** <nome tabela> **DROP COLUMN** <nome coluna>

- *Exemplo:* **Table: aluno**

**Columns:**

<u>matricula</u>	decimal(4,0) PK
nome	varchar(30)
tp_aluno	int(1)
curso	varchar(3)
rua	varchar(30)

```
ALTER TABLE aluno DROP COLUMN rua;
```

# ALTER+DROP COLUMN

```
ALTER TABLE aluno DROP COLUMN rua;
```

Table: **aluno**

Columns:

<u>matricula</u>	decimal(4,0) PK
nome	varchar(30)
tp_aluno	int(1)
curso	varchar(3)

# ALTER+DROP PRIMARY KEY

- *Removendo uma chave primária*

- *Sintaxe:*

```
ALTER TABLE tbl_name DROP PRIMARY KEY
```

- *Exemplo:*

**Table:** *aluno*

**Columns:**

<u><i>matricula</i></u>	decimal(4,0) PK
<i>nome</i>	varchar(30)
<i>tp_aluno</i>	int(1)
<i>curso</i>	varchar(3)
<i>rua</i>	varchar(30)

```
ALTER TABLE aluno DROP PRIMARY KEY;
```

# ALTER+DROP COLUMN

```
ALTER TABLE aluno DROP PRIMARY KEY;
```

Table: **aluno**

Columns:

matricula	decimal(4,0)
nome	varchar(30)
tp_aluno	int(1)
curso	varchar(3)

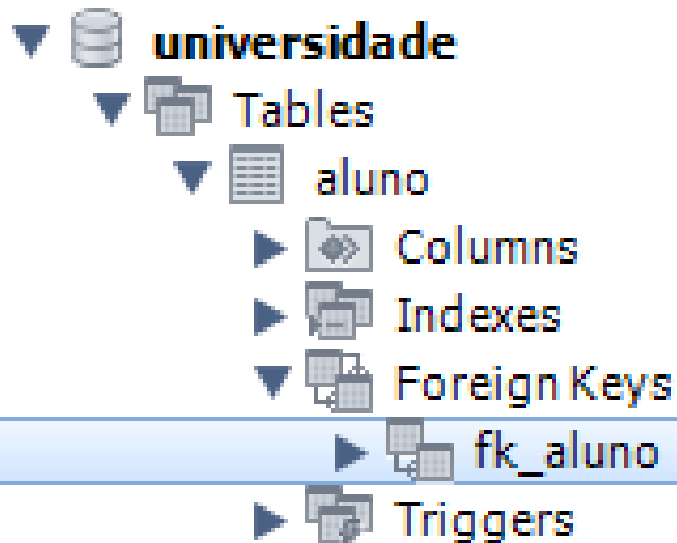
# ALTER+DROP FOREIGN KEY

- *Removendo uma chave estrangeira*

- *Sintaxe:*

```
ALTER TABLE tbl_name DROP FOREIGN KEY fk_symbol
```

- *Exemplo:*



Foreign Key: **fk\_aluno**

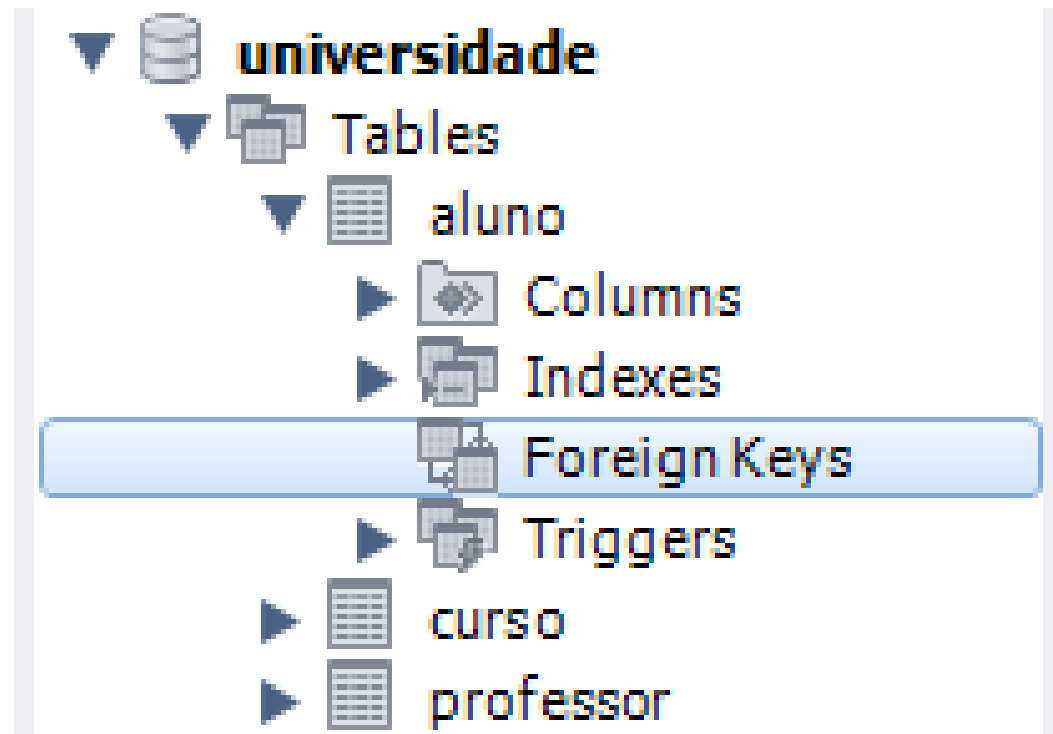
Definition:

Target	curso (curso → codigo)
On Update	RESTRICT
On Delete	RESTRICT

```
ALTER TABLE aluno DROP FOREIGN KEY fk_aluno;
```

# ALTER+DROP FOREIGN KEY

```
ALTER TABLE aluno DROP FOREIGN KEY fk_aluno;
```



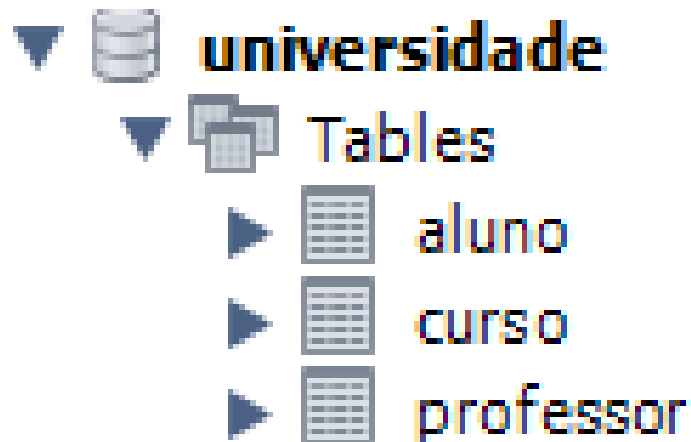
# ALTER+RENAME

- *Renomeando uma tabela*

- *Sintaxe:*

```
ALTER TABLE tbl_name RENAME [TO|AS] new_tbl_name
```

- *Exemplo:*

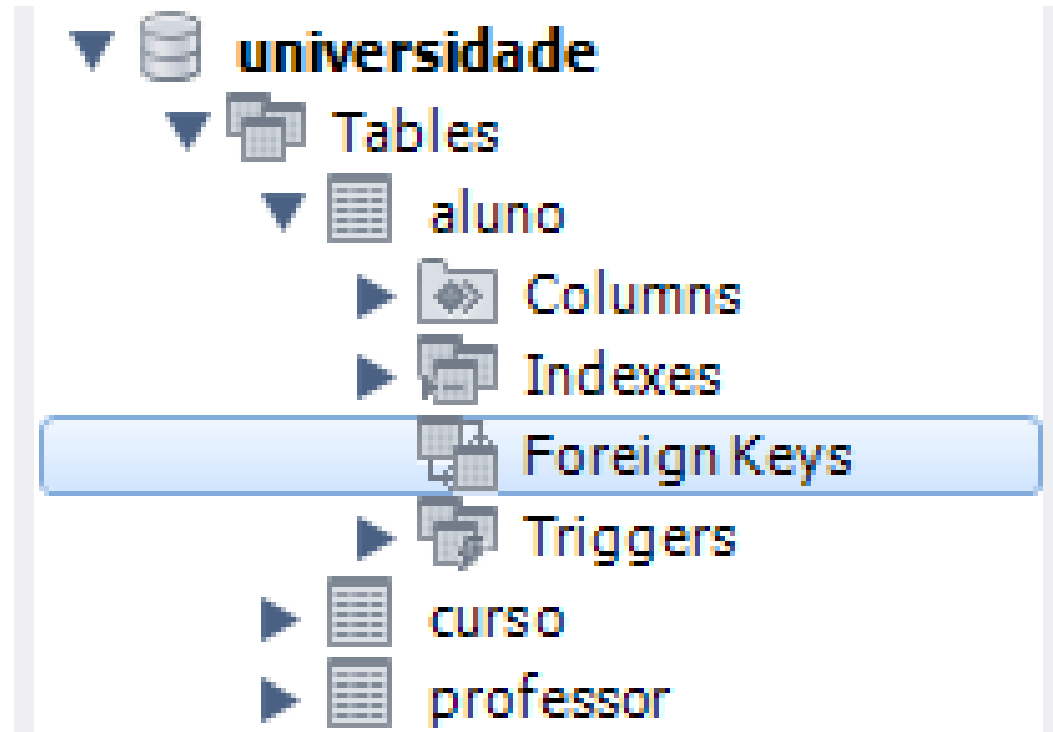


```
ALTER TABLE aluno RENAME al;
```



# ALTER+RENAME

```
ALTER TABLE aluno DROP FOREIGN KEY fk_aluno;
```



# ALTER+RENAME

```
ALTER TABLE aluno RENAME al;
```

