

---

# **Introdução às Bases de Dados**

Modelo Relacional - IV

---

FCUL, Departamento de Informática

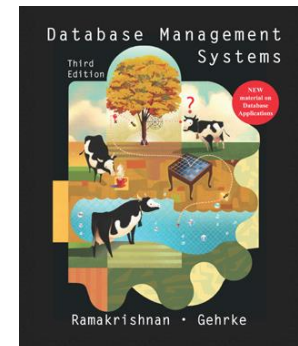
Ano Letivo 2021/2022

Ana Paula Afonso

# Sumário e Referências

---

- Sumário
  - Comandos SQL
    - Alteração de tabelas
    - Remoção de tabelas
    - Pesquisas de dados em tabelas (forma básica)
  - Vistas
    - Criação de vistas
    - Independência dos dados e privacidade
    - Atualização de dados
- Referências
  - R. Ramakrishnan (**capítulo 3, secção 3.4, 3.6 e 3.7**)



# Síntese de Comandos SQL

---

- **SQL-DDL**: *Data Definition Language*

operações sobre a estrutura das tabelas e gestão de restrições de integridade

- CREATE TABLE
- DROP TABLE
- ALTER TABLE
- CREATE VIEW

- **SQL-DML**: *Data Manipulation Language*

operações sobre os **dados** das tabelas

- INSERT INTO
- DELETE FROM
- UPDATE
- SELECT

# Tabelas de Exemplo

*Students* (***sid***: *string*, *name*: *string*, *login*: *string*, *age*: *integer*, *gpa*: *real*)

*Enrolled* (***studid***: *string*, *cid*: *string*, *grade*: *string*)

Foreign key			Primary key				
<u>cid</u>	<u>grade</u>	<u>studid</u>	<u>sid</u>	<u>name</u>	<u>login</u>	<u>age</u>	<u>gpa</u>
Carnatic101	C	53831	50000	Dave	dave@cs	19	3.3
Reggae203	B	53832	53666	Jones	jones@cs	18	3.4
Topology112	A	53650	53688	Smith	smith@ee	18	3.2
History105	B	53666	53650	Smith	smith@math	19	3.8
			53831	Madayan	madayan@music	11	1.8
			53832	Guldu	guldu@music	12	2.0

Enrolled (Referencing relation)

Students (Referenced relation)

# Alteração de Tabelas

---

- Alteração de tabelas com o comando ALTER TABLE
  - Adição, alteração, e remoção de **colunas**
- **Adição de coluna**
  - Linhas que já existem ficam a *NULL* nessa coluna
  - `ALTER TABLE Students ADD COLUMN maiden-name CHAR(10)`
- **Alteração de coluna**
  - Novo domínio deverá abranger valores já existentes nessa coluna
  - `ALTER TABLE Students MODIFY COLUMN maiden-name CHAR(20)`
- **Remoção de uma coluna**
  - `ALTER TABLE Students DROP COLUMN maiden-name`

# Alteração de Tabelas

---

- Alteração de tabelas com o comando ALTER TABLE
  - Adição e remoção de **restrições de integridade**
- **Adição de restrição de integridade**
  - Apenas se todos os dados já existentes cumprirem a nova regra
  - `ALTER TABLE Enrolled ADD CONSTRAINT nn_enrolled_grade CHECK (grade IS NOT NULL)`
- **Remoção de restrição de integridade**
  - Tem efeito permanente
  - `ALTER TABLE Enrolled DROP CONSTRAINT nn_enrolled_grade`

# Pesquisas de Dados

---

- Pesquisas de dados com o comando **SELECT**
  - Seleção de linhas e colunas de uma ou mais tabelas

- Exemplos com uma tabela

```
SELECT * FROM Students
```

<i>sid</i>	<i>name</i>	<i>login</i>	<i>age</i>	<i>gpa</i>
50000	Dave	dave@cs	19	3.2
53666	Jones	jones@cs	18	3.3
53688	Smith	smith@ee	18	3.2
53650	Smith	smith@math	19	3.7
53831	Madayan	madayan@music	11	1.8
53832	Guldu	guldu@music	12	2.0

```
SELECT name, login  
FROM Students  
WHERE age >= 18
```

<i>name</i>	<i>login</i>
Dave	dave@cs
Jones	jones@cs
Smith	smith@ee
Smith	smith@math

# Pesquisas de Dados

- Exemplo que combina informação de duas tabelas
  - Nome dos estudantes e nome da disciplina para os quais  $\text{grade} = 'A'$  e  $S.\text{sid} = E.\text{studid}$

```
SELECT S.name, E.cid  
FROM Students S, Enrolled E  
WHERE S.sid = E.studid AND  
      grade = 'A'
```

Foreign key			Primary key				
cid	grade	studid	sid	name	login	age	gpa
Carnatic101	C	53831	50000	Dave	dave@cs	19	3.3
Reggae203	B	53832	53666	Jones	jones@cs	18	3.4
Topology112	A	53650	53688	Smith	smith@ee	18	3.2
History105	B	53666	53650	Smith	smith@math	19	3.8
			53831	Madayan	madayan@music	11	1.8
			53832	Guldu	guldu@music	12	2.0

Enrolled (Referencing relation)

Students (Referenced relation)



# Vistas sobre Dados

---

- Uma vista **é uma tabela virtual**
  - Cujas linhas não são explicitamente armazenadas
  - Conteúdo determinado na criação da view por um comando SELECT
  - Pode mostrar apenas algumas colunas e linhas da(s) tabela(s) de base
- Essencial para independência e privacidade de dados
  - Pode abstrair alterações na tabela de base
  - Cada tipo de utilizador pode ter acesso a vistas específicas

# Vistas sobre Dados

---

- Exemplo

```
CREATE VIEW Old_students (name, login, age) AS  
SELECT name, login, age  
FROM students  
WHERE age >= 18
```

<i>sid</i>	<i>name</i>	<i>login</i>	<i>age</i>	<i>gpa</i>
50000	Dave	dave@cs	19	3.2
53666	Jones	jones@cs	18	3.3
53688	Smith	smith@ee	18	3.2
53650	Smith	smith@math	19	3.7
53831	Madayan	madayan@music	11	1.8
53832	Guldu	guldu@music	12	2.0

Consulta da *view* executa a interrogação SQL associada

```
SELECT * FROM Old_students
```

# Vistas sobre Datos

---

- Exemplo

```
CREATE VIEW B-Students (name, sid, course) AS
  SELECT S.sname, S.sid, E.cid
  FROM Students S, Enrolled E
  WHERE S.sid = E.studid AND E.grade = 'B'
```

<i>name</i>	<i>sid</i>	<i>course</i>
Jones	53666	History105
Guldu	53832	Reggae203

# Vistas sobre Dados

---

- SQL distingue entre vistas
  - Cujas linhas podem ser modificadas (*updatable views*)
  - E vistas onde novas linhas podem ser inseridas (*insertable views*)
  - Tem de existir uma relação de um para um entre as linhas da vista e das linhas das respetivas tabelas

# Inserção de Dados nas Views

VIEW **good\_student**      TABLE **student**

```
CREATE VIEW good_student (sid, gpa) AS  
  SELECT sid, gpa FROM student  
  WHERE gpa >= 3
```

```
INSERT INTO good_student VALUES (10000, 3)  
INSERT INTO good_student VALUES (11000, 1.8)
```

sid	gpa
50000	3.3
53650	3.8
53666	3.4
53688	3.2

sid	gpa
10000	3.0
50000	3.3
53650	3.8
53666	3.4
53688	3.2

sid	name	login	age	gpa
50000	Dave	dave@cs	19	3.2
53666	Jones	jones@cs	18	3.3
53688	Smith	smith@ee	18	3.2
53650	Smith	smith@math	19	3.7
53831	Madayan	madayan@music	11	1.8
53832	Guldu	guldu@music	12	2.0

sid	name	login	age	gpa
10000			0	3.0
11000			0	1.8
50000	Dave	dave@cs	19	3.3
53650	Smith	smith@math	19	3.8
53666	Jones	jones@cs	18	3.4
53688	Smith	smith@ee	18	3.2
53831	Madayan	madayan@music	11	1.8
53832	Guldu	guldu@music	12	2.0

# Inserção de Dados nas Views

VIEW **good\_student**      TABLE **student**

```
CREATE VIEW good_student (sid, gpa) AS
  SELECT sid, gpa FROM student
 WHERE gpa >= 3
WITH CHECK OPTION
```

sid	gpa
10000	3.0
50000	3.3
53650	3.8
53666	3.4
53688	3.2

sid	name	login	age	gpa
10000			0	3.0
11000			0	1.8
50000	Dave	dave@cs	19	3.3
53650	Smith	smith@math	19	3.8
53666	Jones	jones@cs	18	3.4
53688	Smith	smith@ee	18	3.2
53831	Madayan	madayan@music	11	1.8
53832	Guldu	guldu@music	12	2.0

```
INSERT INTO good_student values (12000, 3)
INSERT INTO good_student values (13000, 1.8)
```

sid	name	login	age	gpa
10000			0	3.0
11000			0	1.8
12000			0	3.0
50000	Dave	dave@cs	19	3.3
53650	Smith	smith@math	19	3.8
53666	Jones	jones@cs	18	3.4
53688	Smith	smith@ee	18	3.2
53831	Madayan	madayan@music	11	1.8
53832	Guldu	guldu@music	12	2.0

# Atualização de Dados nas Views

---

```
CREATE VIEW authors_CA AS  
    SELECT * FROM Authors WHERE state='CA'
```

```
UPDATE authors_CA SET state='NJ'
```

--problema/desafio: deixa de pertencer à vista

```
CREATE VIEW authors_CA AS  
    SELECT * FROM Authors WHERE state='CA'  
WITH CHECK OPTION
```

```
UPDATE authors_CA SET state='NJ'
```

# Remoção de Tabelas e Vistas

---

- Remoção de tabela e dos seus dados
  - As chaves estrangeiras para esta tabela têm de ser removidas antes
  - `DROP TABLE Student`

`DROP TABLE Students RESTRICT`

- Apaga a tabela exceto se existirem *views* ou restrições de integridade referencial

`DROP TABLE Students CASCADE`

- Apaga a tabela e todas as *views* e restrições de integridade referencial

- Remoção de view
  - `DROP VIEW Good_students`



# MySQL observations

---

- DROP TABLE IF EXISTS *não é standard*
  - Se a tabela não existir não dá erro
- SET foreign\_key\_checks = 0;
  - Lista de tabelas: drop table ...
- SET foreign\_key\_checks = 1;