



# **DISCIPLINA: Banco de Dados 1**

Prof. **GIOVANI** Volnei Meinerz

Aula 17 – SQL (cont.)

# Objetivos da Aula

---

→ Entender a estrutura básica de **consulta** em SQL e sua aplicação

# Consulta SQL: Estrutura Básica

- Estrutura básica de uma consulta
  - Sintaxe SQL

```
SELECT A1 [, A2, ... , An]  
FROM T1 [, T2, ... , Tn]  
[WHERE P]
```

- **SELECT** – lista dos atributos desejados no resultado
  - $A_i$  representa um atributo
- **FROM** – lista das tabelas a serem acessadas
  - $T_i$  representa uma relação
- **WHERE** – predicado envolvendo atributos das tabelas
  - $P$  é o predicado.

# Consulta SQL: Estrutura Básica (cont.)

## → Sintaxe SQL

```
SELECT A1 [, A2, ... , An]  
FROM T1 [, T2, ... , Tn]  
[WHERE P]
```

- Toma como entrada as tabelas listadas na cláusula **FROM**
- Opera sobre elas conforme especificado nas cláusulas **WHERE** e **SELECT**

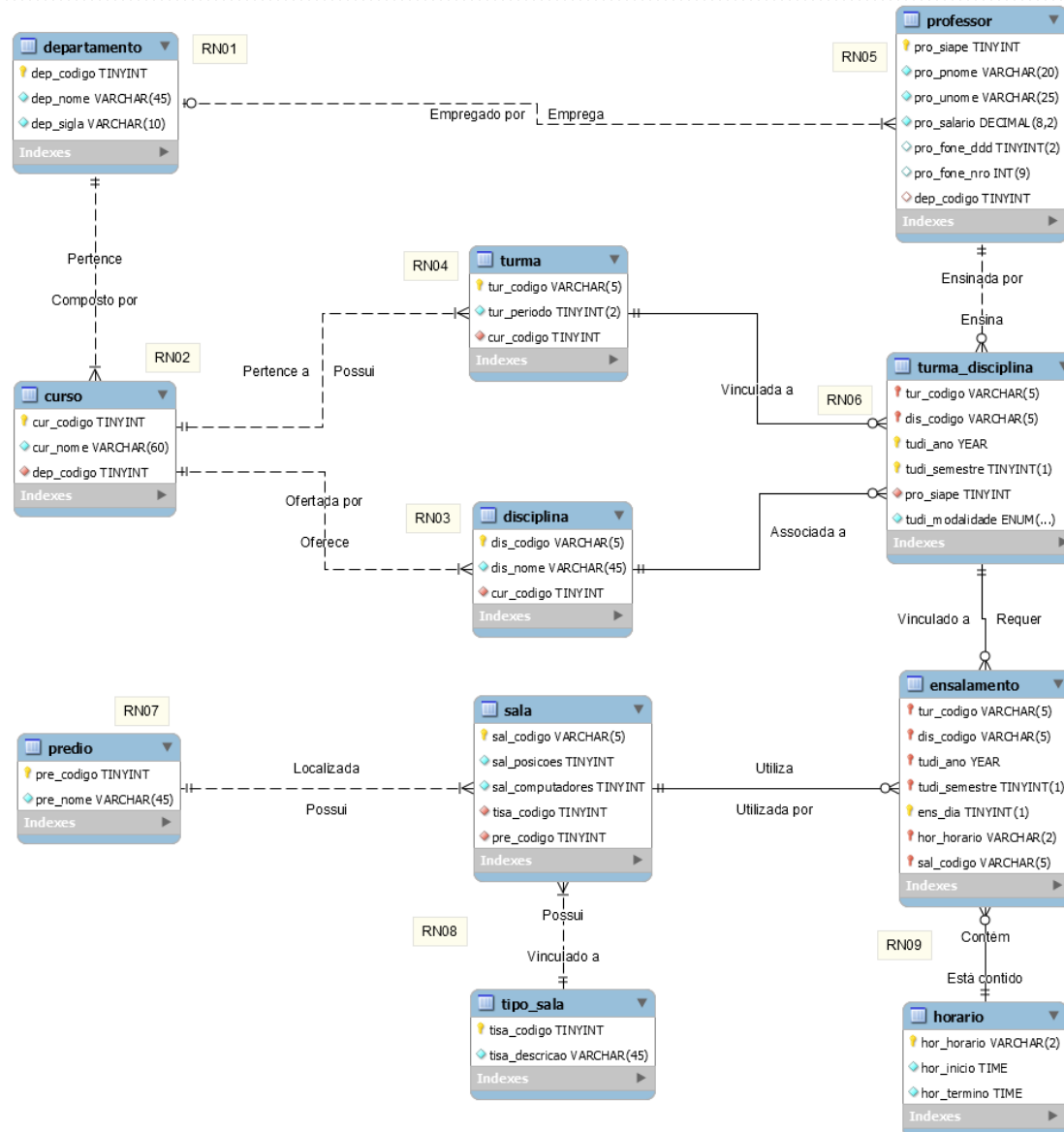
# Consulta SQL: Estrutura Básica (cont.)

---

- Entendendo o significado de uma consulta SQL
  - **Etapa 1** – gera um produto cartesiano das tabelas listadas na cláusula **FROM**
  - **Etapa 2** – Aplica os predicados especificados na cláusula **WHERE** sobre o resultado da **Etapa 1**
  - **Etapa 3** – para cada tupla do resultado da **Etapa 2**, mostra os valores dos atributos especificados na cláusula **SELECT**

# Consultas sobre uma única Tabela

## ✈ Cenário



# Consultas sobre uma única Tabela (cont.)

## → Consulta

- Selecione todos os registros e colunas da tabela **PROFESSOR**
- Os atributos são encontrados na tabela **PROFESSOR**, de forma que colocamos essa tabela na cláusula **FROM**

```
1  -- Listando cada um dos atributos
2  • SELECT pro_siape, pro_pnome, pro_unome, pro_salario, pro_fone_ddd, pro_fone_nro, dep_codigo
3     FROM professor;
4
5  -- Substituindo todos os atributos pelo caractere "*"
6  • SELECT *
7     FROM professor;
```

| pro_siape | pro_pnome | pro_unome  | pro_salario | pro_fone_ddd | pro_fone_nro | dep_codigo |
|-----------|-----------|------------|-------------|--------------|--------------|------------|
| 1         | Hanka     | Ruebbert   | 60000.00    | 43           | 35239000     | 6          |
| 2         | Tilo      | Gerhold    | 65000.00    | NULL         | NULL         | 2          |
| 3         | Ekkehart  | Schubbert  | 55000.00    | 11           | 754210000    | 3          |
| 4         | Gerhard   | Huettig    | 85000.00    | 14           | 33448888     | 2          |
| 5         | Anoela    | Lehmann    | 75000.00    | 43           | 35237777     | 2          |
| 6         | Lisa      | Reimann    | 95000.00    | NULL         | NULL         | 1          |
| 7         | Corinna   | Engellmann | 130000.00   | 55           | 995554500    | 1          |
| 8         | Manfred   | Schubbert  | 79000.00    | 43           | 998456587    | 2          |
| 9         | Lena      | Reimann    | 145000.00   | 14           | 997465544    | 3          |
| 10        | Giovani   | Meinerz    | 84000.00    | 55           | 999838457    | 1          |
| 11        | Luiz      | Marengo    | 67000.00    | 55           | 996814596    | NULL       |
| 12        | Mano      | Lima       | 81000.00    | 55           | 975824684    | NULL       |
| 13        | Cenair    | Maicá      | 89000.00    | 55           | 997896341    | NULL       |
| NULL      | NULL      | NULL       | NULL        | NULL         | NULL         | NULL       |

# Consultas sobre uma única Tabela (cont.)

## → Consulta

- Selecione o nome e o salário de todos os professores

```
1 -- Selecione o nome e o salário de todos os professores
2 • SELECT pro_pnome, pro_unome, pro_salario
3 FROM professor;
```

Table: professor

### Columns:

|                  |               |
|------------------|---------------|
| <u>pro_siape</u> | tinyint(4) PK |
| pro_pnome        | varchar(20)   |
| pro_unome        | varchar(25)   |
| pro_salario      | decimal(8,2)  |
| pro_fone_ddd     | tinyint(2)    |
| pro_fone_nro     | int(9)        |
| dep_codigo       | tinyint(4)    |

| pro_pnome | pro_unome  | pro_salario |
|-----------|------------|-------------|
| Hanka     | Ruebbert   | 60000.00    |
| Tilo      | Gerhold    | 65000.00    |
| Ekkehart  | Schubbert  | 55000.00    |
| Gerhard   | Huettia    | 85000.00    |
| Angela    | Lehmann    | 75000.00    |
| Lisa      | Reimann    | 95000.00    |
| Corinna   | Endellmann | 130000.00   |
| Manfred   | Schubbert  | 79000.00    |
| Lena      | Reimann    | 145000.00   |
| Giovani   | Meinerz    | 84000.00    |
| Luiz      | Marenco    | 67000.00    |
| Mano      | Lima       | 81000.00    |
| Cenair    | Maicá      | 89000.00    |



# Eliminar Linhas Repetidas

- ➔ Para eliminar linhas repetidas do resultado, pode-se utilizar a cláusula **DISTINCT**

- ➔ Sintaxe SQL

```
SELECT DISTINCT A1 [, A2, ... , An]  
FROM T1 [, T2, ... , Tn]  
[WHERE P]
```

- ➔ A cláusula **DISTINCT** só pode ser colocada imediatamente após o **SELECT**

# Eliminar Linhas Repetidas (cont.)

## → Cenário

- Selecione as turmas/disciplinas que foram ensinadas, mostrando apenas turma, disciplina, ano e semestre

```
1 -- Selecione as turmas/disciplinas que foram ensinadas,
2 -- mostrando apenas turma, disciplina, ano e semestre
3 • SELECT tur_codigo, dis_codigo, tudi_ano, tudi_semestre
4 FROM turma_disciplina;
```

Table: **turma\_disciplina**

Columns:

|                      |   |
|----------------------|---|
| <u>tur_codigo</u>    | varchar(5) PK                                   |
| <u>dis_codigo</u>    | varchar(5) PK                                   |
| <u>tudi_ano</u>      | year(4) PK                                      |
| <u>tudi_semestre</u> | tinyint(1) PK                                   |
| pro_siape            | tinyint(4)                                      |
| tudi_modalidade      | enum('Presencial','Distância','Semipresencial') |

| tur_codigo | dis_codigo | tudi_ano | tudi_semestre |
|------------|------------|----------|---------------|
| C41        | EC34D      | 2017     | 2             |
| C51        | EC35B      | 2017     | 1             |
| C51        | EC35B      | 2017     | 2             |
| E21        | MA35B      | 2017     | 2             |
| ES21       | IF62H      | 2017     | 1             |
| ES21       | IF62H      | 2017     | 2             |
| ES31       | IF63C      | 2016     | 1             |
| M31        | EM33H      | 2017     | 2             |
| N12A       | AN32C      | 2017     | 2             |
| N12B       | AN32C      | 2017     | 2             |
| N12B       | AN32C      | 2018     | 1             |
| N12SP      | AN32C      | 2017     | 2             |
| NULL       | NULL       | NULL     | NULL          |

# Eliminar Linhas Repetidas (cont.)

## → Cenário

- Encontre as turmas/disciplinas, mostrando apenas o seu código, desconsiderando turmas/disciplinas que tenham sido ministradas mais de uma vez

```
1 -- Selecione as turmas/disciplinas que foram ensinadas,  
2 -- mostrando apenas turma, disciplina, ano e semestre  
3 • SELECT tur_codigo, dis_codigo, tudi_ano, tudi_semestre  
4 FROM turma_disciplina;
```

| tur_codigo | dis_codigo | tudi_ano | tudi_semestre |
|------------|------------|----------|---------------|
| C41        | EC34D      | 2017     | 2             |
| C51        | EC35B      | 2017     | 1             |
| C51        | EC35B      | 2017     | 2             |
| E21        | MA35B      | 2017     | 2             |
| ES21       | IF62H      | 2017     | 1             |
| ES21       | IF62H      | 2017     | 2             |
| ES31       | IF63C      | 2016     | 1             |
| M31        | EM33H      | 2017     | 2             |
| N12A       | AN32C      | 2017     | 2             |
| N12B       | AN32C      | 2017     | 2             |
| N12B       | AN32C      | 2018     | 1             |
| N12SP      | AN32C      | 2017     | 2             |
| NULL       | NULL       | NULL     | NULL          |

```
9 • SELECT DISTINCT tur_codigo  
10 FROM turma_disciplina;
```

| tur_codigo |
|------------|
| C41        |
| C51        |
| E21        |
| ES21       |
| ES31       |
| M31        |
| N12A       |
| N12B       |
| N12SP      |

# Eliminar Linhas Repetidas (cont.)

## → Cenário

- Quando **apenas um atributo** é especificado para a cláusula DISTINCT, a busca extrairá valores únicos para esse atributo
- Quando **mais de um atributo** é especificado, a busca extrairá **combinações únicas** para os atributos especificados

```
9 • SELECT DISTINCT tur_codigo  
10 FROM turma_disciplina;
```

| tur_codigo |
|------------|
| C41        |
| C51        |
| E21        |
| ES21       |
| ES31       |
| M31        |
| N12A       |
| N12B       |
| N12SP      |

```
13 • SELECT DISTINCT tur_codigo, pro_siape  
14 FROM turma_disciplina;
```

| tur_codigo | pro_siape |
|------------|-----------|
| C41        | 6         |
| C51        | 12        |
| C51        | 13        |
| E21        | 4         |
| ES21       | 11        |
| ES31       | 5         |
| M31        | 1         |
| N12A       | 11        |
| N12B       | 7         |
| N12SP      | 11        |

# SELECT e os Operadores Aritméticos

- A cláusula **SELECT** pode conter expressões aritméticas envolvendo operadores aritméticos

| Operadores Aritméticos | Descrição         |
|------------------------|-------------------|
| +                      | Somar             |
| -                      | Subtrair          |
| *                      | Multiplicar       |
| /                      | Dividir           |
| ^                      | Elevar à potência |

- Operam sobre atributos de tuplas

# SELECT e os Operadores Aritméticos (cont.)

## → Cenário

- Selecione e mostre os nomes dos professores e seu respectivo salário incrementado em 10%

```
2 • SELECT
3     pro_pnome,
4     pro_unome,
5     pro_salario
6 FROM
7     professor;
```

| pro_pnome | pro_unome | pro_salario |
|-----------|-----------|-------------|
| Hanka     | Ruebbert  | 60000.00    |
| Tilo      | Gerhold   | 65000.00    |
| Ekkehart  | Schubbert | 55000.00    |
| Gerhard   | Huettig   | 85000.00    |
| Anaëla    | Lehmann   | 75000.00    |
| Lisa      | Reimann   | 95000.00    |
| Corinna   | Engelmann | 130000.00   |
| Manfred   | Schubbert | 79000.00    |
| Lena      | Reimann   | 145000.00   |
| Giovani   | Meinerz   | 84000.00    |
| Luiz      | Marengo   | 67000.00    |
| Mano      | Lima      | 81000.00    |
| Cenair    | Maicá     | 89000.00    |

```
9 • SELECT
10     pro_pnome,
11     pro_unome,
12     pro_salario * 1.1
13 FROM
14     professor;
```

| pro_pnome | pro_unome | pro_salario * 1.1 |
|-----------|-----------|-------------------|
| Hanka     | Ruebbert  | 66000.000         |
| Tilo      | Gerhold   | 71500.000         |
| Ekkehart  | Schubbert | 60500.000         |
| Gerhard   | Huettig   | 93500.000         |
| Anaëla    | Lehmann   | 82500.000         |
| Lisa      | Reimann   | 104500.000        |
| Corinna   | Engelmann | 143000.000        |
| Manfred   | Schubbert | 86900.000         |
| Lena      | Reimann   | 159500.000        |
| Giovani   | Meinerz   | 92400.000         |
| Luiz      | Marengo   | 73700.000         |
| Mano      | Lima      | 89100.000         |
| Cenair    | Maicá     | 97900.000         |

# Conectivos Lógicos e Operadores de Comparação

- Conectivos lógicos podem ser usados na cláusula **WHERE**
- Operadores de comparação podem ser usados para comparar *strings* e expressões aritméticas

| Comando ou Opção                    | Descrição                             |
|-------------------------------------|---------------------------------------|
| <b>Conectivos Lógicos</b>           | Utilizados em expressões condicionais |
| AND<br>OR<br>NOT                    |                                       |
| <b>Operadores de Comparação</b>     |                                       |
| =<br><<br>><br><=<br>>=<br><> ou != |                                       |

# A Cláusula WHERE

- Permite selecionar tuplas, em uma tabela, que satisfazem um predicado específico

```
SELECT A1 [, A2, ... , An]  
FROM T1 [, T2, ... , Tn]  
[WHERE P]
```

- **WHERE** – predicado envolvendo atributos das tabelas
  - **P** é o predicado.



# A Cláusula WHERE (cont.)

## → Cenário

- Selecione os professores cujo salário seja superior a 100 mil. Mostre nome e salário.

```
3  ●  SELECT
4      pro_pnome,
5      pro_unome,
6      pro_salario
7  FROM
8      professor
9  WHERE
10     pro_salario > 100000;
```

| pro_pnome | pro_unome  | pro_salario |
|-----------|------------|-------------|
| Corinna   | Endellmann | 130000.00   |
| Lena      | Reimann    | 145000.00   |

# A Cláusula WHERE (cont.)

## → Cenário

- Selecione todas as salas cuja quantidade de posições disponíveis seja inferior ou igual a 30. Mostre o código da sala e a quantidade de posições existentes na sala

```
3 • SELECT sal_codigo, sal_posicoes
4 FROM sala;
```

| sal_codigo | sal_posicoes |
|------------|--------------|
| A040       | 45           |
| A140       | 44           |
| A146       | 44           |
| I201       | 42           |
| I202       | 42           |
| I205       | 45           |
| K008       | 24           |
| K009       | 24           |
| P003       | 27           |
| P005       | 45           |
| P101       | 44           |
| P105       | 45           |
| P205       | 45           |
| NULL       | NULL         |

```
6 • SELECT
7     sal_codigo,
8     sal_posicoes
9 FROM
10     sala
11 WHERE
12     sal_posicoes <= 30;
```

| sal_codigo | sal_posicoes |
|------------|--------------|
| K008       | 24           |
| K009       | 24           |
| P003       | 27           |
| NULL       | NULL         |

Table: sala

Columns:

|                  |               |
|------------------|---------------|
| sal_codigo       | varchar(5) PK |
| sal_posicoes     | tinyint(4)    |
| sal_computadores | tinyint(4)    |
| tisa_codigo      | tinyint(4)    |
| pre_codigo       | tinyint(4)    |

# Consultas envolvendo múltiplas tabelas

- É a capacidade de recuperar dados de duas ou mais tabelas
- SQL permite listar as tabelas na cláusula **FROM** de uma sentença **SELECT**

```
SELECT A1 [, A2, ... , An]  
FROM T1 [, T2, ... , Tn]  
[WHERE P]
```

# Consultas envolvendo múltiplas tabelas (cont.)

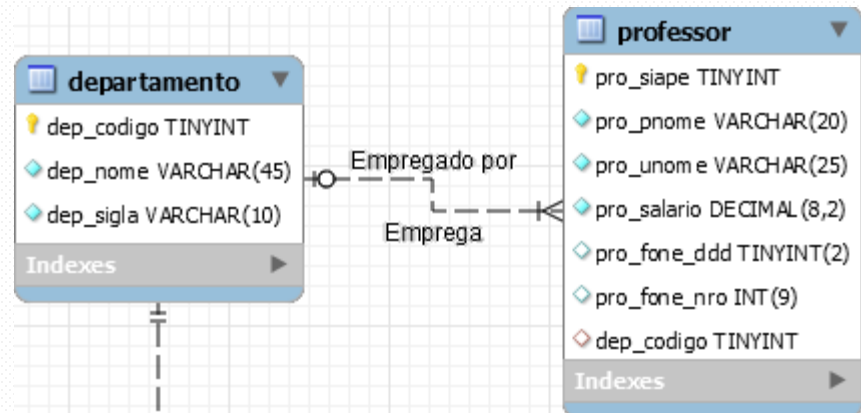
---

- Ao envolver duas ou mais tabelas, diz-se que está sendo realizada uma **junção de tabelas**
- Neste caso, serão selecionados apenas os pares de tuplas em que os valores de **atributos comuns** coincidam
- Para tanto, basta especificar os **atributos comuns** na cláusula **WHERE**, por meio de uma comparação de igualdade entre a **chave estrangeira** e a **chave primária** das tabelas relacionadas

# Consultas envolvendo múltiplas tabelas (cont.)

## → Cenário

- Selecione e mostre os nomes dos professores e respectivos departamentos (mostre a sigla) aos quais estão vinculados.



## → Análise

- Quais são as tabelas envolvidas?
  - **departamento** e **professor**
- Quais atributos devem ter seu valor de campo mostrados?
  - **pro\_pnome**, **pro\_unome** e **dep\_sigla**
- Quais condições precisam ser satisfeitas?
  - Comparação de igualdade entre a chave estrangeira e a primária

# Consultas envolvendo múltiplas tabelas (cont.)

## → Cenário (cont.)

- Selecione e mostre os nomes dos professores e respectivos departamentos (mostre a sigla) aos quais estão vinculados.

```
2 • SELECT
3     pro_pnome, pro_unome, dep_codigo
4 FROM
5     professor;
```

| pro_pnome | pro_unome | dep_codigo |
|-----------|-----------|------------|
| Hanka     | Ruebbert  | 6          |
| Tilo      | Gerhold   | 2          |
| Ekkehart  | Schubbert | 3          |
| Gerhard   | Huettig   | 2          |
| Angela    | Lehmann   | 2          |
| Lisa      | Reimann   | 1          |
| Corinna   | Engelmann | 1          |
| Manfred   | Schubbert | 2          |
| Lena      | Reimann   | 3          |
| Giovani   | Meinerz   | 1          |
| Luiz      | Marengo   | NULL       |
| Mano      | Lima      | NULL       |
| Cenair    | Maicá     | NULL       |

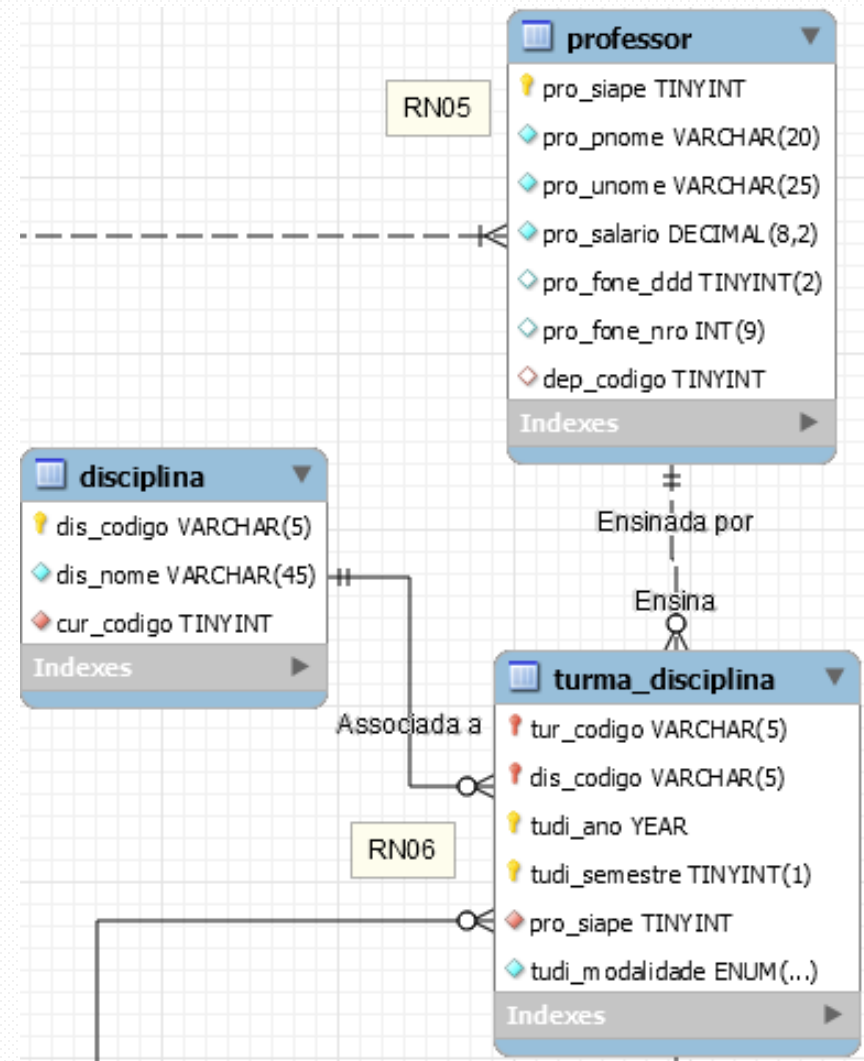
```
7 • SELECT
8     pro_pnome, pro_unome, dep_sigla
9 FROM
10    professor, departamento
11 WHERE
12    professor.dep_codigo = departamento.dep_codigo;
```

| pro_pnome | pro_unome | dep_sigla |
|-----------|-----------|-----------|
| Hanka     | Ruebbert  | DAMEC     |
| Tilo      | Gerhold   | DAELE     |
| Ekkehart  | Schubbert | DAMAT     |
| Gerhard   | Huettig   | DAELE     |
| Angela    | Lehmann   | DAELE     |
| Lisa      | Reimann   | DACOM     |
| Corinna   | Engelmann | DACOM     |
| Manfred   | Schubbert | DAELE     |
| Lena      | Reimann   | DAMAT     |
| Giovani   | Meinerz   | DACOM     |

# Consultas envolvendo múltiplas tabelas (cont.)

## ➤ Cenário

- Para todos os professores que ministraram aula em relação a alguma turma/disciplina, mostre seus nomes, os códigos das turmas/disciplinas ensinadas, o respectivo nome da disciplina, ano e semestre em que a turma/disciplina foi ofertada.



# Consultas envolvendo múltiplas tabelas (cont.)

## → Cenário (cont.)

- Para todos os professores que ministraram aula em relação a alguma turma/disciplina, mostre seus nomes, os códigos das turmas/disciplinas ensinadas, o respectivo nome da disciplina, ano e semestre em que a turma/disciplina foi ofertada.

```
2 • SELECT
3     pro_pnome, pro_unome, tur_codigo, dis_nome, tudi_ano, tudi_semestre
4 FROM
5     professor, turma_disciplina, disciplina
6 WHERE
7     turma_disciplina.pro_siape = professor.pro_siape
8     AND turma_disciplina.dis_codigo = disciplina.dis_codigo;
```

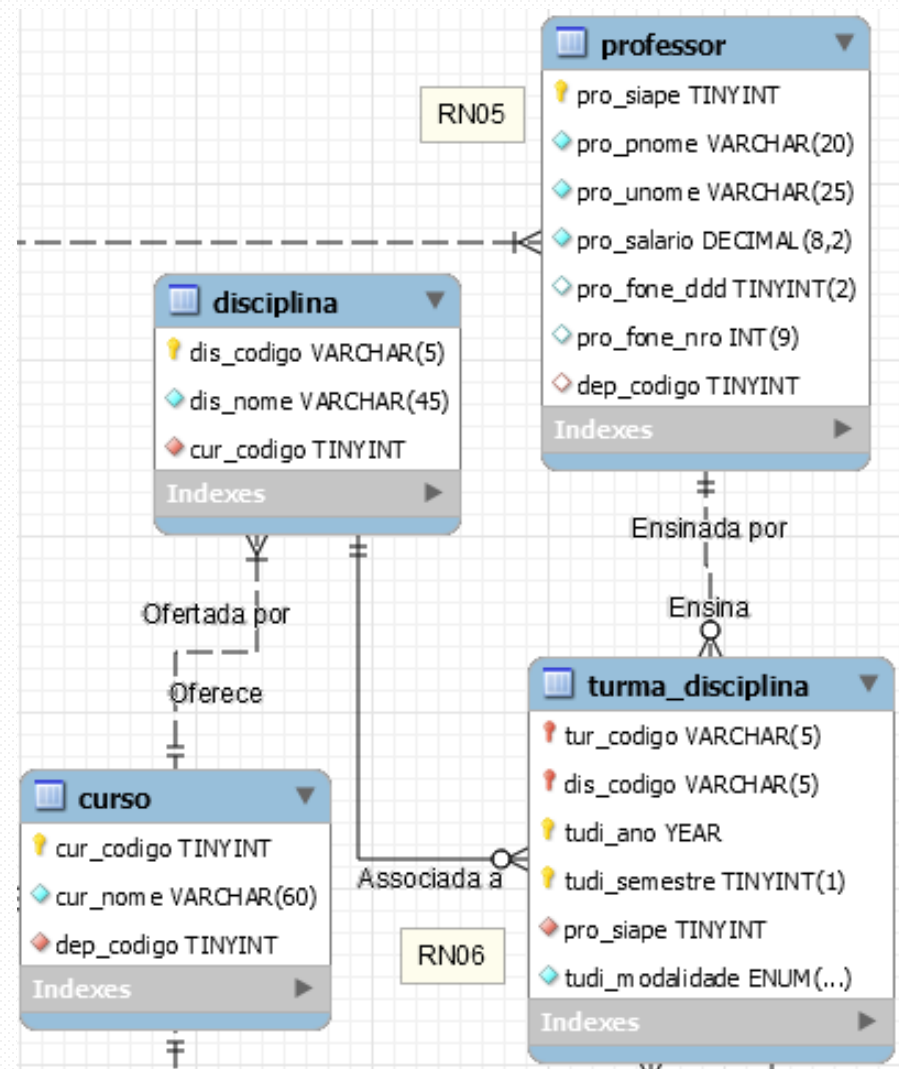
| pro_pnome | pro_unome  | tur_codigo | dis_nome                  | tudi_ano | tudi_semestre |
|-----------|------------|------------|---------------------------|----------|---------------|
| Hanka     | Ruebbert   | M31        | Física Experimental       | 2017     | 2             |
| Gerhard   | Huettio    | E21        | Probabilidade Estatística | 2017     | 2             |
| Angela    | Lehmann    | ES31       | Requisitos de Software    | 2016     | 1             |
| Lisa      | Reimann    | C41        | Banco De Dados 1          | 2017     | 2             |
| Corinna   | Engellmann | N12B       | Banco De Dados 1          | 2017     | 2             |
| Corinna   | Engellmann | N12B       | Banco De Dados 1          | 2018     | 1             |
| Luiz      | Marengo    | ES21       | Banco De Dados 1          | 2017     | 1             |
| Luiz      | Marengo    | ES21       | Banco De Dados 1          | 2017     | 2             |
| Luiz      | Marengo    | N12A       | Banco De Dados 1          | 2017     | 2             |
| Luiz      | Marengo    | N12SP      | Banco De Dados 1          | 2017     | 2             |
| Mano      | Lima       | C51        | Banco de Dados 2          | 2017     | 1             |
| Cenair    | Maicá      | C51        | Banco de Dados 2          | 2017     | 2             |



# Consultas envolvendo múltiplas tabelas (cont.)

## ➤ Cenário

- Para todos os professores que ministraram aula em relação a alguma turma/disciplina, mostre seus nomes, os códigos das turmas/disciplinas ensinadas, o respectivo nome da disciplina e o curso do qual a disciplina é originária.



# Consultas envolvendo múltiplas tabelas (cont.)

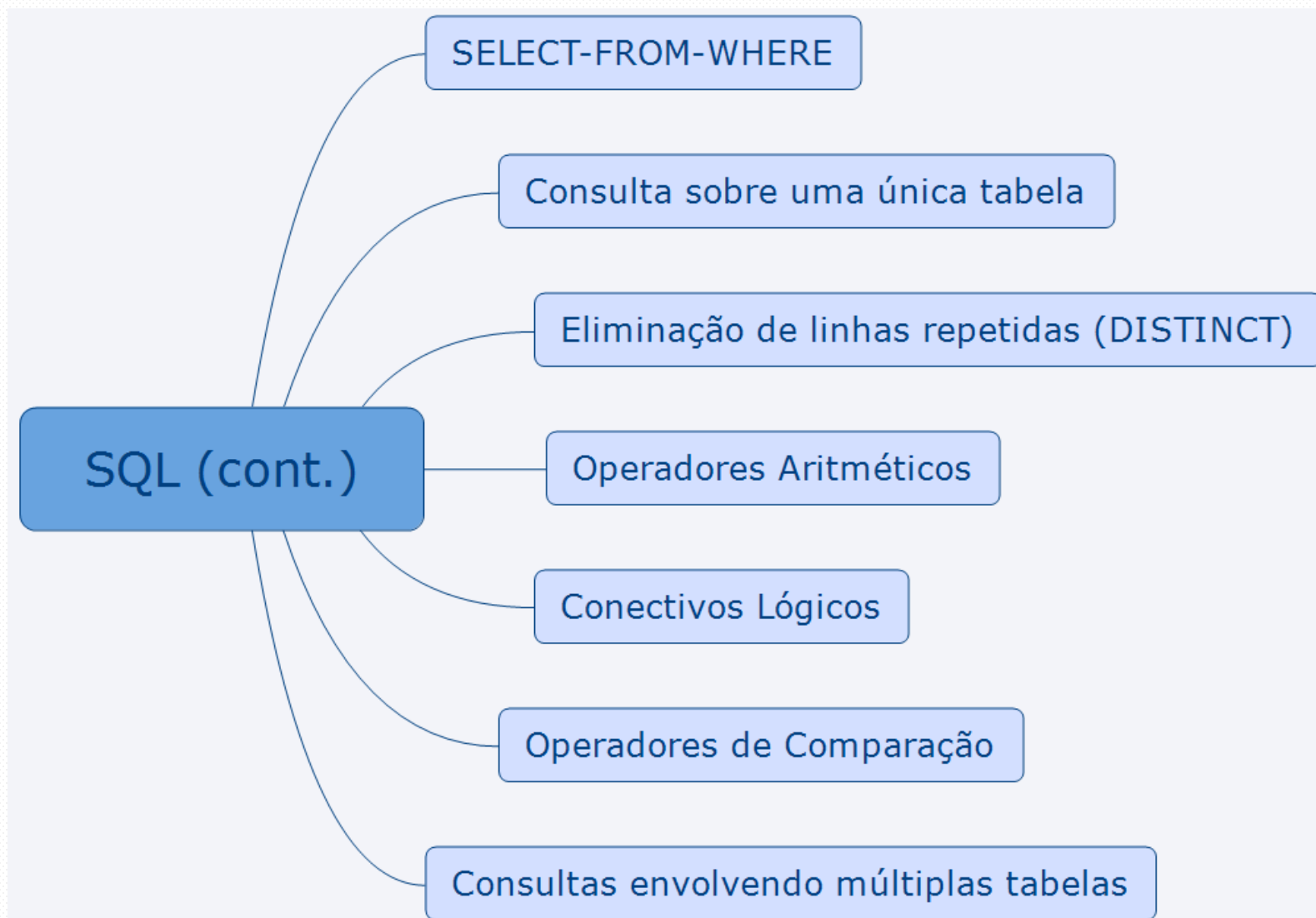
## → Cenário

- Para todos os professores que ministraram aula em relação a alguma turma/disciplina, mostre seus nomes, os códigos das turmas/disciplinas ensinadas, o respectivo nome da disciplina e o curso do qual a disciplina é originária.

```
4 • SELECT
5     pro_pnome, pro_unome, tur_codigo, dis_nome, cur_nome
6 FROM
7     professor, turma_disciplina, disciplina, curso
8 WHERE
9     turma_disciplina.pro_siape = professor.pro_siape
10    AND turma_disciplina.dis_codigo = disciplina.dis_codigo
11    AND disciplina.cur_codigo = curso.cur_codigo;
```

| pro_pnome | pro_unome  | tur_codigo | dis_nome                  | cur_nome  |
|-----------|------------|------------|---------------------------|---|
| Lisa      | Reimann    | C41        | Banco De Dados 1          | Engenharia de Computação                            |
| Mano      | Lima       | C51        | Banco de Dados 2          | Engenharia de Computação                            |
| Cenair    | Maicá      | C51        | Banco de Dados 2          | Engenharia de Computação                            |
| Gerhard   | Huettio    | E21        | Probabilidade Estatística | Engenharia Elétrica                                 |
| Hanka     | Ruebbert   | M31        | Física Experimental       | Engenharia Mecânica                                 |
| Anoela    | Lehmann    | ES31       | Requisitos de Software    | Engenharia de Software                              |
| Luiz      | Marengo    | ES21       | Banco De Dados 1          | Engenharia de Software                              |
| Luiz      | Marengo    | ES21       | Banco De Dados 1          | Engenharia de Software                              |
| Corinna   | Engellmann | N12B       | Banco De Dados 1          | Tecnologia em Análise e Desenvolvimento de Sistemas |
| Corinna   | Engellmann | N12B       | Banco De Dados 1          | Tecnologia em Análise e Desenvolvimento de Sistemas |
| Luiz      | Marengo    | N12A       | Banco De Dados 1          | Tecnologia em Análise e Desenvolvimento de Sistemas |
| Luiz      | Marengo    | N12SP      | Banco De Dados 1          | Tecnologia em Análise e Desenvolvimento de Sistemas |

# Resumo da Aula





# **DISCIPLINA: Banco de Dados 1**

Prof. **GIOVANI** Volnei Meinerz

Aula 17 – SQL (cont.)