

Banco de Dados 1

Consulta de Dados

Consultas no Banco de Dados

Selecionar um conjunto de registros em uma ou mais tabelas que atenda a uma determinada condição definida pelo comando.

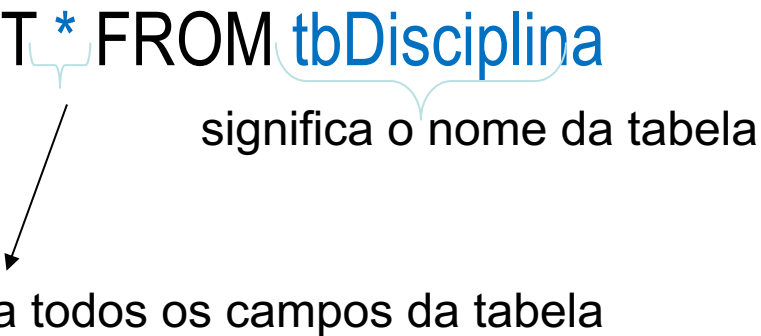
```
SELECT DISTINCT | ALL <nome_coluna>  
FROM <nome_tabela>  
WHERE condição  
GROUP BY <nome_coluna>  
HAVING <condição>  
ORDER BY <nome_coluna> ASC | DESC
```

Consultas no Banco de Dados

Sintaxe básica

As consultas no banco de dados são feitas utilizando a seguinte sintaxe:

`SELECT * FROM tbDisciplina`



significa todos os campos da tabela

significa o nome da tabela

Obs.: O * pode ser substituído pelo nome das colunas da tabela:

`SELECT DIS_CODIGO, DIS_NOME
FROM tbDisciplina`

```
SELECT DIS_CARGA_HORARIA FROM tbDisciplina
```

DIS_CARGA_HORARIA
80
40
80
80
40

A cláusula DISTINCT, utilizada após o SELECT, exibe apenas os dados distintos.

```
SELECT DISTINCT DIS_CARGA_HORARIA  
FROM tbDisciplina
```

DIS_CARGA_HORARIA
80
40

Consultas no Banco de Dados

Cláusula where

```
SELECT * FROM tbDisciplina  
WHERE DIS_CODIGO = 1
```

condição da consulta

Quando a consulta possui mais de uma condição, separa-se as condições por **OR** ou **AND**

```
SELECT * FROM tbDisciplina  
WHERE DIS_CODIGO = 1 OR  
      DIS_CODIGO = 2
```

Podemos utilizar os mesmos operadores utilizados no WHERE nos scripts de exclusão e alteração.

Consultas no Banco de Dados

Ordenação de consultas

A cláusula **ORDER BY** faz com que os resultados de uma consulta apareçam na ordem classificada.

```
SELECT * FROM tbDisciplina  
ORDER BY DIS_NOME
```

A ordem pode ser determinada de forma ascendente (ASC) ou descendente (DESC).

Por padrão, a ordem é ascendente (ASC), atribuído implicitamente.

Consultas no Banco de Dados

Renomeando os termos

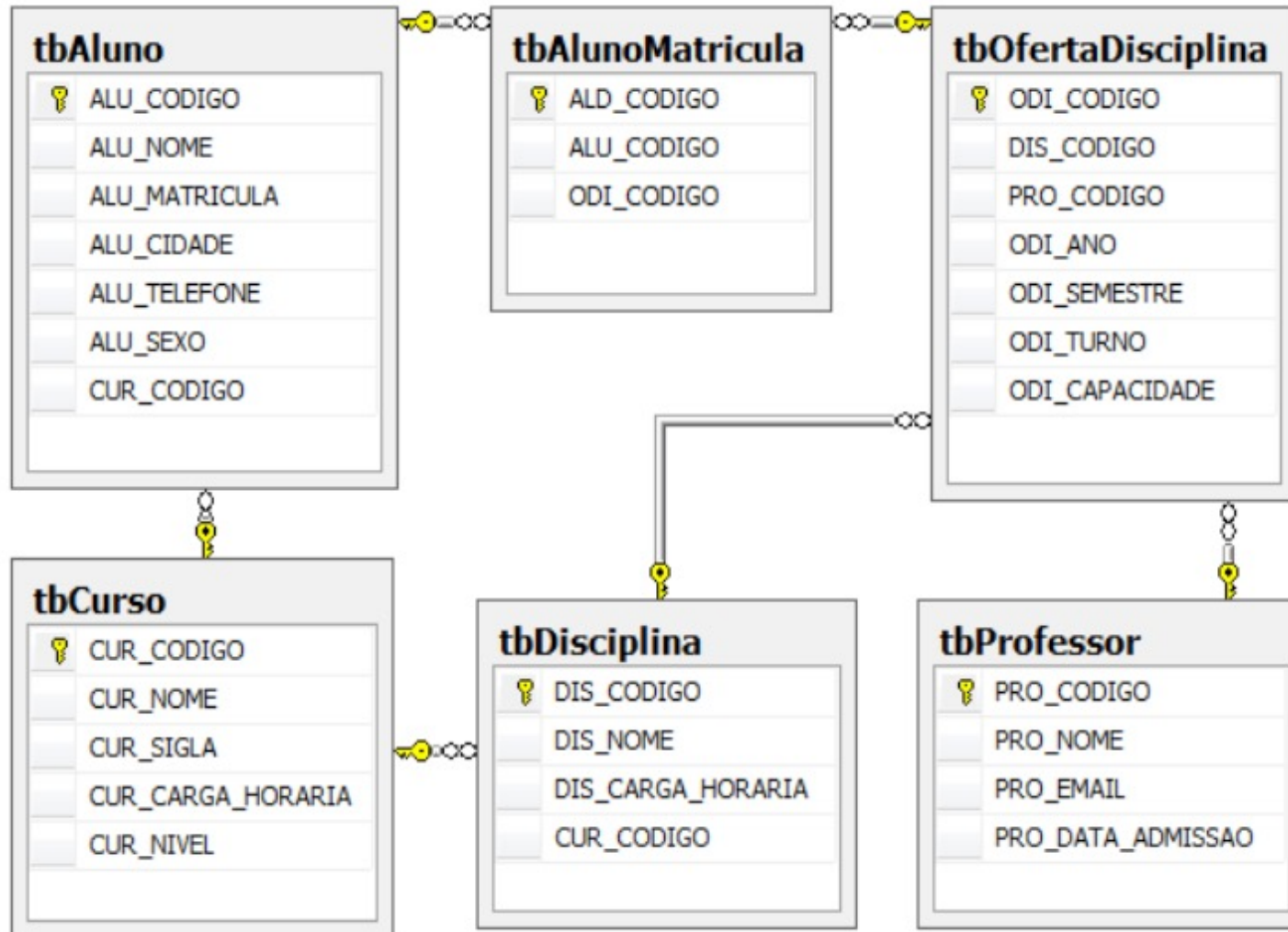
Podemos renomear tanto os termos da cláusula **SELECT** (colunas) quanto os termos da cláusula **FROM** (tabelas).

```
SELECT DIS_NOME as NOME FROM tbDisciplina
```

Obs.: Caso desejar renomear para um texto com mais de uma palavra, coloca-se o nome dentro de aspas simples.

```
SELECT DIS_NOME as 'Nome da disciplina' FROM tbDisciplina
```


Consultas no Banco de Dados



Consultas no Banco de Dados

- 1) Obter os dados de todos os alunos.
- 2) Obter o nome dos cursos de nível Técnico.
- 3) Obter o nome e o e-mail dos professores cujo código está entre 1 e 3.
- 4) Obter o nome e a carga horária das disciplinas Banco de Dados I, Banco de Dados II e Introdução à Programação. Renomeie as colunas para Nome e Carga horária, respectivamente, e ordene o resultado pelo Nome.
- 5) Obter o nome e o número de matrícula dos alunos cujo código está entre 5 e 10, o nome contém João e a cidade é Tianguá (considere variações na escrita do nome). Ordene pela matrícula em ordem decrescente.
- 6) Obter o nome e o e-mail dos professores admitidos no ano de 2018, cujo código é menor que 10 ou maior que 50.
- 7) Obter o nome e a cidade dos alunos de Tianguá (ou Tiangua), Viçosa (considere as diversas variações de escrita) e o telefone é diferente de nulo. Renomeie as colunas para Nome e Cidade do aluno, respectivamente.

Consultas no Banco de Dados

Funções agregadas

Funções agregadas são aquelas que tomam uma coleção (um conjunto ou multiconjunto) de valores como entrada e retornam um único valor.

AVG: média

MIN: menor valor

MAX: maior valor

SUM: soma

COUNT: quantidade de valores

Consultas no Banco de Dados

Funções agregadas

```
SELECT COUNT(*) as QTDE_DISCIPLINAS  
FROM tbDisciplina
```

Consultas no Banco de Dados

Agrupamento de Dados (Group By)

Agrupa diversos registros com base em uma ou mais colunas da tabela. A cláusula GROUP BY é costumeiramente associada ao uso de funções de agregação.

```
SELECT COUNT(*) as QTDE, CUR_CODIGO  
FROM tbDisciplina  
GROUP BY CUR_CODIGO
```

Um SELECT com a cláusula GROUP BY sem utilizar qualquer função de agregação é equivalente a um SELECT DISTINCT.

Consultas no Banco de Dados

tbDisciplina

DIS_CODIGO	DIS_NOME	PRO_CODIGO	CUR_CODIGO
1	BANCO DE DADOS 1	1	1
2	FÍSICA GERAL	NULL	2
3	POO	2	1
4	BANCO DE DADOS 2	1	1
5	FÍSICO-QUÍMICA	3	2
6	INTRODUÇÃO A BANCO DE DADOS	1	3
7	INTRODUÇÃO À PROGRAMAÇÃO	2	3
8	ESTRUTURA DE DADOS	2	1

tbProfessor

PRO_CODIGO	PRO_NOME
1	João da Silva
2	Maria de Fátima
3	Joaquim Alencar
4	Maria Aparecida

tbCurso

CUR_CODIGO	CUR_NOME
1	BCC
2	Lic. Física
3	Técnico em Informática

Consultas no Banco de Dados

tbDisciplina

DIS_CODIGO	DIS_NOME	PRO_CODIGO	CUR_CODIGO
1	BANCO DE DADOS 1	1	1
2	FÍSICA GERAL	NULL	2
3	POO	2	1
4	BANCO DE DADOS 2	1	1
5	FÍSICO-QUÍMICA	3	2
6	INTRODUÇÃO A BANCO DE DADOS	1	3
7	INTRODUÇÃO À PROGRAMAÇÃO	2	3
8	ESTRUTURA DE DADOS	2	1

tbProfessor

PRO_CODIGO	PRO_NOME
1	João da Silva
2	Maria de Fátima
3	Joaquim Alencar
4	Maria Aparecida

tbCurso

CUR_CODIGO	CUR_NOME
1	BCC
2	Lic. Física
3	Técnico em Informática

Consultas no Banco de Dados

DIS_CODIGO	DIS_NOME	PRO_CODIGO	CUR_CODIGO
1	BANCO DE DADOS 1	1	1
3	POO	2	1
4	BANCO DE DADOS 2	1	1
8	ESTRUTURA DE DADOS	2	1

DIS_CODIGO	DIS_NOME	PRO_CODIGO	CUR_CODIGO
2	FÍSICA GERAL	NULL	2
5	FÍSICO-QUÍMICA	3	2

DIS_CODIGO	DIS_NOME	PRO_CODIGO	CUR_CODIGO
6	INTRODUÇÃO A BANCO DE DADOS	1	3
7	INTRODUÇÃO À PROGRAMAÇÃO	2	3

Consultas no Banco de Dados

Agrupamento de Dados (Group By)

```
SELECT CUR_CODIGO, COUNT(*) as QTDE  
FROM tbDisciplina  
GROUP BY CUR_CODIGO
```

Considerando os dados apresentados anteriormente, a consulta acima exibe o seguinte resultado.

CUR_CODIGO	QTDE
1	4
2	2
3	2

Consultas no Banco de Dados

- 1) Obter a quantidade de cursos.
- 2) Obter a média de carga horária dos cursos.
- 3) Obter a quantidade de alunos do sexo masculino da cidade de Tianguá (Tiangua ou Tianguá). Renomeie a coluna para Quantidade de alunos.
- 4) Obter a média da carga horária das disciplinas cujo curso é diferente de nulo.
- 5) Obter a quantidade de disciplinas ofertadas a tarde no semestre 2 do ano 2018.
- 6) Obter a maior, a menor e a média de carga horária por nível de curso.
- 7) Obter a quantidade de alunos por cidade.
- 8) Obter a quantidade de disciplinas e o somatório da carga horária das disciplinas por curso.
- 9) Obter a quantidade de disciplinas de Banco de Dados por curso.
- 10) Obter a quantidade de alunos do sexo feminino por cidade.

Consultas no Banco de Dados

Agrupamento de Dados (Group By)

A cláusula HAVING faz restrições ao nível dos grupos que são processados. Utilizada quando pretendemos mostrar os grupos que apresentam uma característica em particular.

Selecionar os cursos com quantidade inferior a 2 disciplinas.

```
SELECT CUR_CODIGO, COUNT(*) FROM tbDisciplina  
GROUP BY CUR_CODIGO  
HAVING COUNT(*) < 2
```

Consultas no Banco de Dados

- 1) Obter a quantidade de alunos por cidade. Exibir apenas as cidades que possuem mais de 10 alunos.
- 2) Obter o somatório da carga horária e a quantidade das disciplinas por curso. Exibir apenas os cursos que possuem mais de duas disciplinas.
- 3) Obter a média de carga horária dos cursos por nível. Exibir apenas os cursos de nível Superior ou Graduação; e apenas os cursos em que a menor carga horária é superior a 3000 horas.
- 4) Obter a quantidade de disciplinas ofertada por professor no semestre 2 do ano 2014. Exibir os professores que possuem mais de 1 disciplina.
- 5) Obter a média de carga horária e a quantidade de disciplinas de Programação por curso. Exibir apenas quando possuir mais de 1 disciplina.
- 6) Obter a quantidade de disciplinas por carga horária. Exibir apenas quando houver mais de uma disciplina por carga horária. Ordene o resultado por carga horária.
- 7) Obter a quantidade de alunos das cidades de Icapuí ou Aracati por sexo. Exibir apenas quando possuir mais de 5 alunos. Ordene o resultado pela quantidade.

Consultas no Banco de Dados

Consultas em mais de uma tabela

A maioria das consultas são realizadas em mais de uma tabela. Para realizar consultas em mais de uma tabela podemos utilizar:

- **LEFT JOIN***: Retorna todos os registros da tabela da esquerda.
- **RIGHT JOIN***: Retorna todos os registros da tabela da direita.
- **[INNER] JOIN***: Retorna todos os registros em comum entre as tabelas.
- **FULL [OUTER] JOIN***: Retorna todos os registros da tabela da esquerda e da tabela da direita.
- **CROSS JOIN**: Realiza o produto cartesiano entre duas relações. Se houver condição de comparação, deve ser incluída na cláusula WHERE.
- **,:** Realiza o produto cartesiano entre duas relações. Se houver condição de comparação, deve ser incluída na cláusula WHERE.

*É necessária uma condição de união (geralmente chaves estrangeiras).

Consultas no Banco de Dados

tbDisciplina

DIS_CODIGO	DIS_NOME	PRO_CODIGO	CUR_CODIGO
1	BANCO DE DADOS 1	1	1
2	FÍSICA GERAL	NULL	2
3	PROGRAMAÇÃO ORIENTADA A OBJETOS	1	1
4	BANCO DE DADOS 2	1	1
...			

tbProfessor

PRO_CODIGO	PRO_NOME
1	João da Silva
2	Maria de Fátima
3	Joaquim Alencar
4	Maria Aparecida

tbCurso

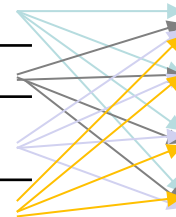
CUR_CODIGO	CUR_NOME
1	BCC
2	Lic. Física

Consultas no Banco de Dados

Obter o nome da disciplina e o nome do professor que ministra a respectiva disciplina?

```
SELECT DIS_NOME, PRO_NOME
FROM tbDisciplina AS D JOIN tbProfessor AS P
ON D.PRO_CODIGO = P.PRO_CODIGO
```

DIS_CODIGO	DIS_NOME	PRO_CODIGO	CUR_CODIGO
1	BANCO DE DADOS 1	1	1
2	FÍSICA GERAL	NULL	2
3	PROGRAMAÇÃO ORIENTADA A OBJETOS	1	1
4	BANCO DE DADOS 2	1	1
...			



PRO_CODIGO	PRO_NOME
1	João da Silva
2	Maria de Fátima
3	Joaquim Alencar
4	Maria Aparecida

Consultas no Banco de Dados

DIS_COD	DIS_NOME	PRO_COD	CUR_COD	PRO_COD	PRO_NOME
1	BANCO DE DADOS 1	1	1	1	João Silva
1	BANCO DE DADOS 1	1	1	2	Maria de Fátima
1	BANCO DE DADOS 1	1	1	3	Joaquim Alencar
1	BANCO DE DADOS 1	1	1	4	Maria Aparecida
2	FÍSICA GERAL	<i>NULL</i>	2	1	João Silva
2	FÍSICA GERAL	<i>NULL</i>	2	2	Maria de Fátima
2	FÍSICA GERAL	<i>NULL</i>	2	3	Joaquim Alencar
2	FÍSICA GERAL	<i>NULL</i>	2	4	Maria Aparecida
3	POO	1	1	1	João Silva
3	POO	1	1	2	Maria de Fátima
3	POO	1	1	3	Joaquim Alencar
3	POO	1	1	4	Maria Aparecida
4	BANCO DE DADOS 2	1	1	1	João Silva
4	BANCO DE DADOS 2	1	1	2	Maria de Fátima
4	BANCO DE DADOS 2	1	1	3	Joaquim Alencar
4	BANCO DE DADOS 2	1	1	4	Maria Aparecida

Consultas no Banco de Dados

Obter o nome da disciplina e o nome do professor que ministra a respectiva disciplina?

```
SELECT DIS_NOME, PRO_NOME
FROM tbDisciplina AS D JOIN tbProfessor AS P
ON D.PRO_CODIGO = P.PRO_CODIGO
```

DIS_CODIGO	DIS_NOME	PRO_CODIGO	CUR_CODIGO
1	BANCO DE DADOS 1	1	1
2	FÍSICA GERAL	NULL	2
3	PROGRAMAÇÃO ORIENTADA A OBJETOS	1	1
4	BANCO DE DADOS 2	1	1
...			

PRO_CODIGO	PRO_NOME
1	João da Silva
2	Maria de Fátima
3	Joaquim Alencar
4	Maria Aparecida

Consultas no Banco de Dados

DIS_COD	DIS_NOME	PRO_COD	CUR_COD	PRO_COD	PRO_NOME
1	BANCO DE DADOS 1	1	1	1	João Silva
1	BANCO DE DADOS 1	1	1	2	Maria de Fátima
1	BANCO DE DADOS 1	1	1	3	Joaquim Alencar
1	BANCO DE DADOS 1	1	1	4	Maria Aparecida
2	FÍSICA GERAL	<i>NULL</i>	2	1	João Silva
2	FÍSICA GERAL	<i>NULL</i>	2	2	Maria de Fátima
2	FÍSICA GERAL	<i>NULL</i>	2	3	Joaquim Alencar
2	FÍSICA GERAL	<i>NULL</i>	2	4	Maria Aparecida
3	POO	1	1	1	João Silva
3	POO	1	1	2	Maria de Fátima
3	POO	1	1	3	Joaquim Alencar
3	POO	1	1	4	Maria Aparecida
4	BANCO DE DADOS 2	1	1	1	João Silva
4	BANCO DE DADOS 2	1	1	2	Maria de Fátima
4	BANCO DE DADOS 2	1	1	3	Joaquim Alencar
4	BANCO DE DADOS 2	1	1	4	Maria Aparecida

Consultas no Banco de Dados

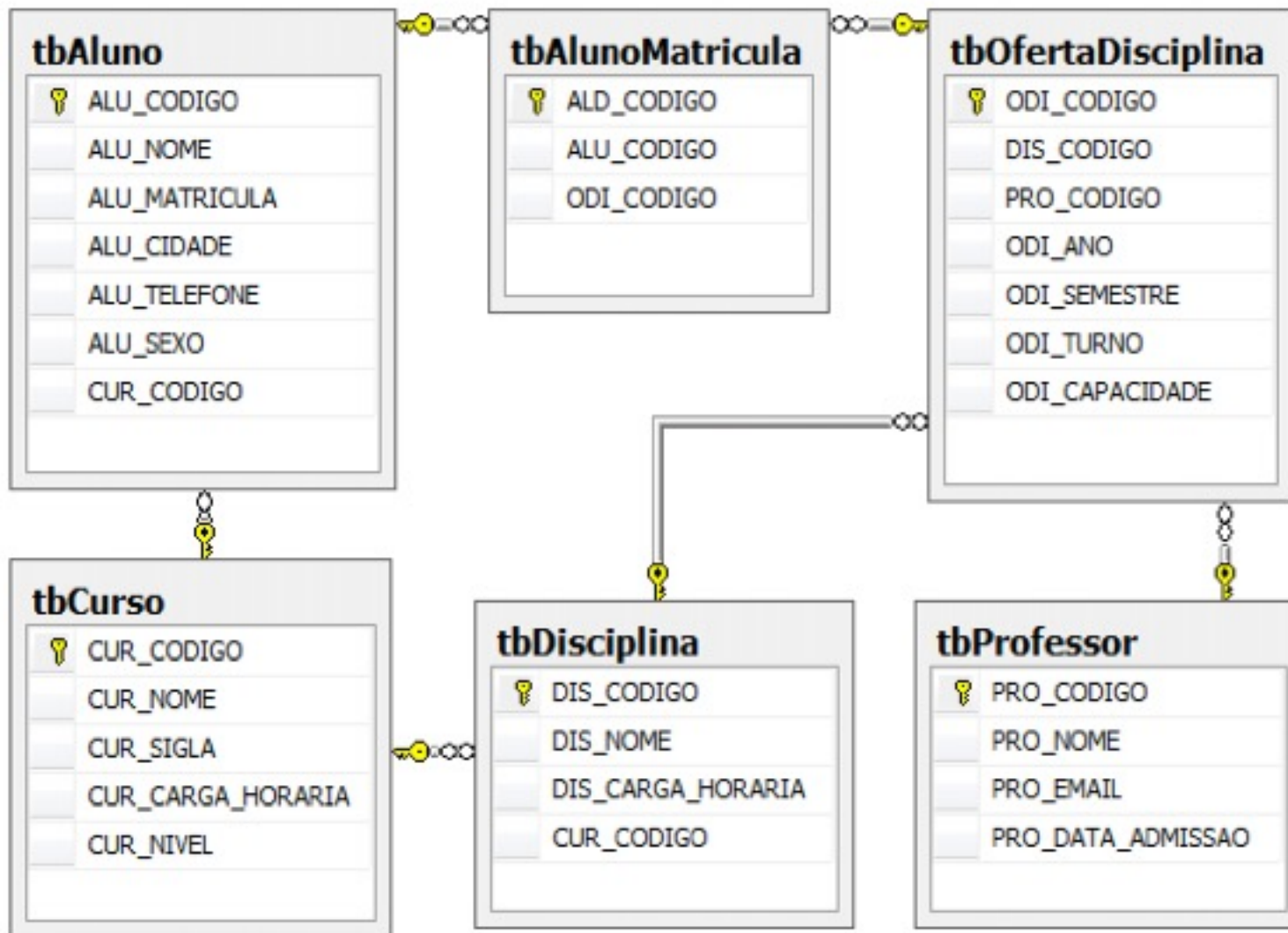
DIS_COD	DIS_NOME	PRO_COD	CUR_COD	PRO_COD	PRO_NOME
1	BANCO DE DADOS 1	1	1	1	João Silva
3	POO	1	1	1	João Silva
4	BANCO DE DADOS 2	1	1	1	João Silva

SELECT DIS_NOME, PRO_NOME

FROM tbDisciplina AS D JOIN tbProfessor AS P
ON D.PRO_CODIGO = P.PRO_CODIGO

DIS_NOME	PRO_NOME
BANCO DE DADOS 1	João Silva
POO	João Silva
BANCO DE DADOS 2	João Silva

Consultas no Banco de Dados



Consultas no Banco de Dados

- 1) Obter o nome das disciplinas e o nome dos cursos as quais são vinculadas.
- 2) Obter os dados dos alunos dos cursos de nível Técnico.
- 3) Obter o nome e a matrícula dos alunos do curso de Bacharelado em Ciência da Computação.
- 4) Obter o nome a carga horária das disciplinas ofertadas no semestre 1 do ano 2018. Ordene o resultado pela carga horária.
- 5) Obter o nome dos professores (de forma distinta) que ministram disciplina no semestre 1 do ano 2018.
- 6) Obter a quantidade de disciplinas do curso de Ciência da Computação.
- 7) Obter o nome da disciplinas ofertadas no turno da noite do semestre 1 do ano 2018, com o nome do respectivo professor.
- 8) Obter o nome dos alunos que cursam a disciplina de Banco de Dados I no semestre 1 do ano 2018.
- 9) Obter o nome e a carga horária das disciplinas ofertadas no semestre 1 do ano 2018 para o curso Bacharelado em Ciência da Computação.
- 10) Obter a quantidade de alunos matriculados por disciplina no semestre 1 do ano 2018. Exibir apenas as disciplinas que possuem mais de 5 alunos. Ordene o resultado pela quantidade de alunos.

Consultas em SQL

SELECT DISTINCT | ALL <nome_coluna>
FROM <nome_tabela>
WHERE condição
GROUP BY <nome_coluna>
HAVING <condição>
ORDER BY <nome_coluna> ASC | DESC

- Operadores Relacionais e Lógicos
- Funções Agregadas
- Junção de Tabelas

Consultas em SQL

Considere as seguintes tabelas:

tbFuncionario

FUN_CODIGO	FUN_NOME	FUN_SALARIO	FUN_CARGO	DEP_CODIGO
1	Antônio Dias	1250,00	Limpeza	1
2	Augusto Nascimento	3500,00	Gerente	1
3	Paulo Viegas	2450,00	Coordenador	2
4	José Antônio	2350,00	Auxiliar	2
5	Maria Esmeralda	2350,00	Coordenador	1

tbDepartamento

DEP_CODIGO	DEP_NOME
1	Almoxarifado
2	Administração

Consultas em SQL

1. Obter o nome dos funcionários da empresa ordenados pelo nome.
2. Obter o nome dos funcionários cujo cargo é Gerente, o nome inicia com A e o salário é superior a 1000.
3. Obter o maior, o menor e a média de salário dos funcionários cujo cargo é Coordenador.
4. Obter o nome de todos os funcionários com o nome do seu respectivo departamento.
5. Obter o somatório dos salários dos funcionários agrupados por departamento. Exiba também o nome do departamento.
6. Obter a quantidade de funcionários da empresa em cada departamento que possui mais de 1 funcionário.
7. Obter a quantidade de funcionários do Cargo Coordenador que existe em cada departamento (exiba também o nome do departamento), ordenado pela quantidade.