

Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas

Sistemas Gerenciadores de Banco de Dados

Professor: André Ulisses da Silva
E-mail: andre.dasilva@prof.sc.senac.br



SQL - CONSULTA DE DADOS



Seleção dos dados	SELECT *, Campos função de agregação
Origem dos dados	FROM Tabela(s)
Filtro do resultado	WHERE operação lógica
Agrupar resultado	GROUP BY campos <small>Obrigatório sempre que tiver função de agregação (SUM, MAX, MIN, AVG E COUNT) Os campos fora da função devem estar aqui</small>
Agrupar filtro	HAVING função de agregação
Ordenar resultado	ORDER BY campos

SQL - MANIPULAÇÃO DE DADOS



WHERE

Operadores para a cláusula WHERE:

=	Igual
<>	Diferente
>	Maior que
<	Menor que
>=	Maior e igual à
<=	Menor e igual à
IN	Lista
LIKE	Que contenha
NOT	Negação
IS NULL	Somente valores nulos
BETWEEN	Entre valores

SQL - CONSULTA DE DADOS



Consultando dados – Group By e suas Funções

O cláusula **GROUP BY** serve para agrupar os resultado em uma consulta SQL,

COUNT	Retorna a quantidade de registros com valores não-NULL diferentes
AVG	Retorna o valor médio
MIN	Retorna o menor valor
MAX	Retorna o maior valor
SUM	Retorna a soma dos valores

SQL - CONSULTA DE DADOS



Consultas Joins

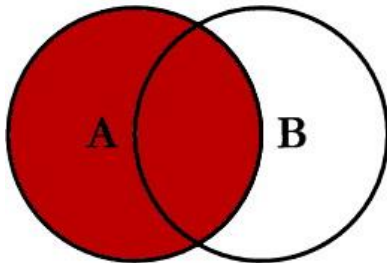
Nos Bancos de Dados relacionais a relação entre tabelas é o ponto fundamental da sua existência. Por isso muitas informações são acessadas através de junções entre tabelas.

Os relacionamentos são organizados através das chaves estrangeiras (FK).

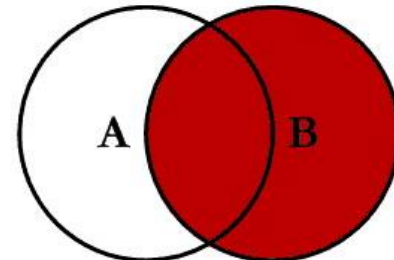
Atenção: Tem-se sempre a preferencia ao juntar duas tabelas à partir dos campos utilizados na foreign key existente entre as tabelas.

SQL - CONSULTA DE DADOS

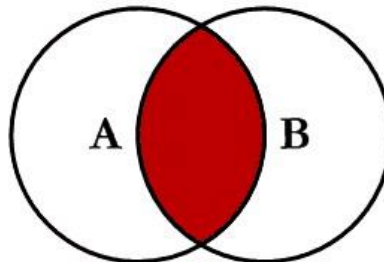
SQL JOINS



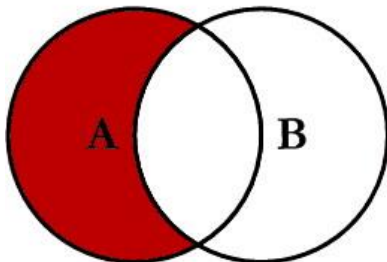
```
SELECT <select_list>  
FROM TableA A  
LEFT JOIN TableB B  
ON A.Key = B.Key
```



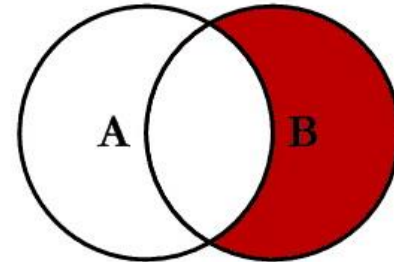
```
SELECT <select_list>  
FROM TableA A  
RIGHT JOIN TableB B  
ON A.Key = B.Key
```



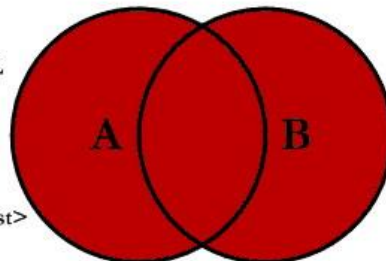
```
SELECT <select_list>  
FROM TableA A  
INNER JOIN TableB B  
ON A.Key = B.Key
```



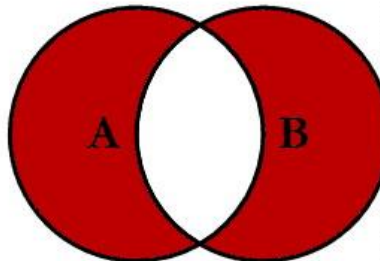
```
SELECT <select_list>  
FROM TableA A  
LEFT JOIN TableB B  
ON A.Key = B.Key  
WHERE B.Key IS NULL
```



```
SELECT <select_list>  
FROM TableA A  
RIGHT JOIN TableB B  
ON A.Key = B.Key  
WHERE A.Key IS NULL
```

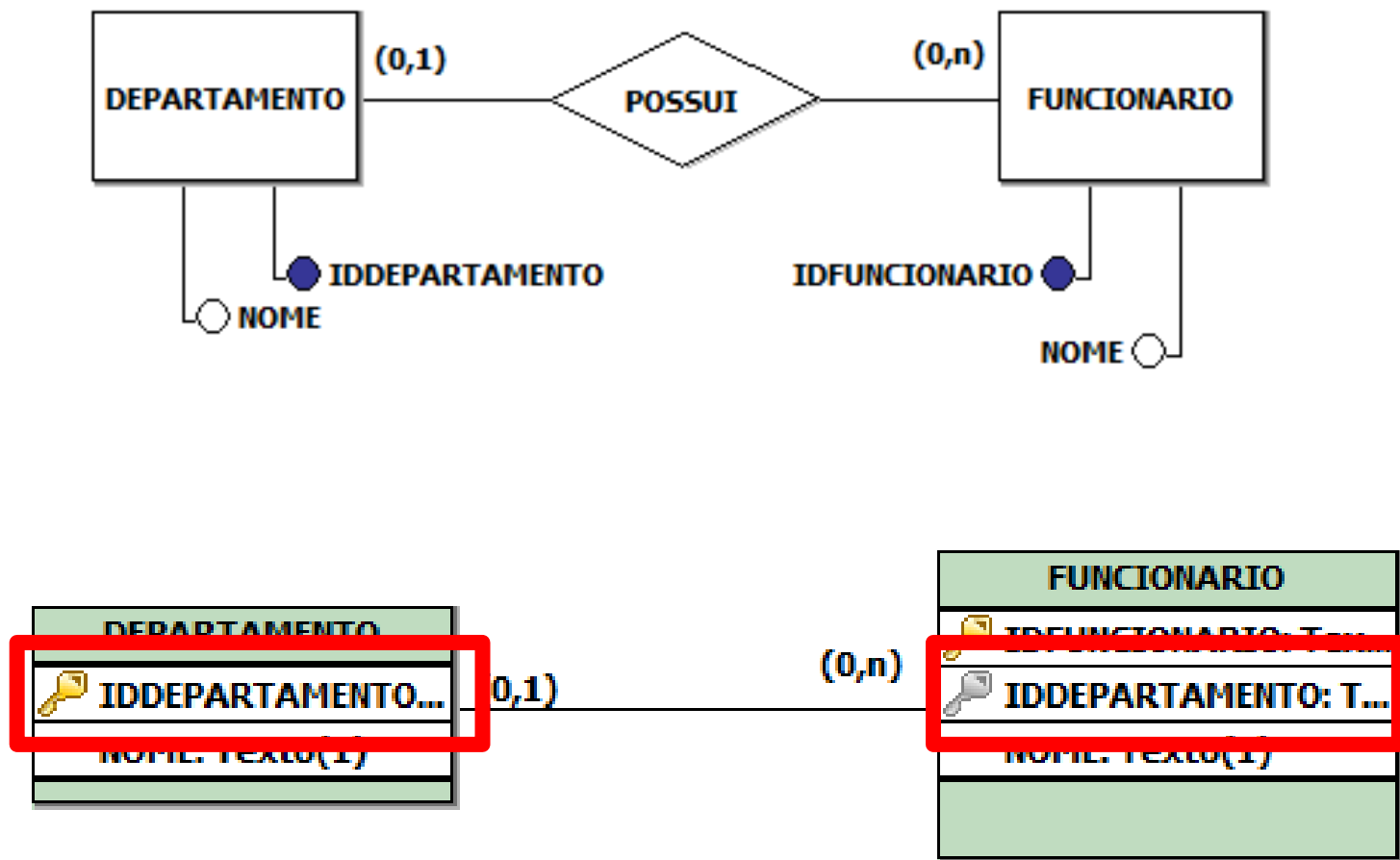


```
SELECT <select_list>  
FROM TableA A  
FULL OUTER JOIN TableB B  
ON A.Key = B.Key
```



```
SELECT <select_list>  
FROM TableA A  
FULL OUTER JOIN TableB B  
ON A.Key = B.Key  
WHERE A.Key IS NULL  
OR B.Key IS NULL
```

SQL - CONSULTA DE DADOS



SQL - CONSULTA DE DADOS



```
CREATE TABLE DEPARTAMENTO (  
    IDDEPARTAMENTO INT NOT NULL PRIMARY KEY auto_increment,  
    NOME VARCHAR(100));
```

```
CREATE TABLE FUNCIONARIO (  
    IDFUNCIONARIO INT NOT NULL PRIMARY KEY auto_increment,  
    IDDEPARTAMENTO INT NOT NULL,  
    NOME VARCHAR(100),  
    SEXO CHAR(1),  
    IDADE INT,  
    DTADMISSAO DATETIME NOT NULL,  
    DTDEMISSAO DATETIME,  
    SALARIO NUMERIC(9,2),  
    FOREIGN KEY (IDDEPARTAMENTO) REFERENCES DEPARTAMENTO (IDDEPARTAMENTO)  
);
```


SQL - CONSULTA DE DADOS



Consultas Joins

Exemplo:

Tabela **Departamento**

IDDEPARTAMENTO	NOME
1	Administração
2	Marketing
3	Financeiro
4	Cobrança
5	Logística
6	manutenção

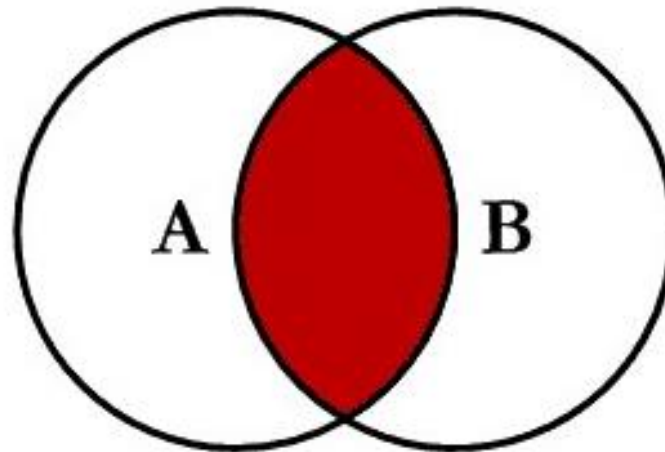
Tabela **Funcionário**

IDFUNCIONARIO	NOME	IDDEPARTAMENTO
1	João	1
2	Maria	2
3	Marcia	2
4	Ana	3
5	Lucas	3
6	Antonio	5
7	Karina	
8	Carlos	

condição de junção

Inner Join

As junções INNER JOIN se caracterizam por uma seleção que retorna apenas os dados que atendem a **condição de junção**



```
SELECT <select_list>  
FROM TableA A  
INNER JOIN TableB B  
ON A.Key = B.Key
```

SQL - CONSULTA DE DADOS



Inner Joins

Exemplo:

Tabela **Departamento**

IDDEPARTAMENTO	NOME
1	Administração
2	Marketing
3	Financeiro
4	Cobrança
5	Logística
6	manutenção

Tabela **Funcionário**

IDFUNCIONARIO	NOME	IDDEPARTAMENTO
1	João	1
2	Maria	2
3	Marcia	2
4	Ana	3
5	Lucas	3
6	Antonio	5
7	Karina	
8	Carlos	

condição de junção
DEPARTAMENTO INNER JOIN FUNCIONARIO

SQL - CONSULTA DE DADOS



Inner Joins

Exemplo:

Tabela Departamento

IDDEPARTAMENTO	NOME
1	Administração
2	Marketing
3	Financeiro
4	Cobrança
5	Logística
6	manutenção

Tabela Funcionário

IDFUNCIONARIO	NOME	IDDEPARTAMENTO
1	João	1
2	Maria	2
3	Marcia	2
4	Ana	3
5	Lucas	3
6	Antonio	5
7	Karina	
8	Carlos	

condição de junção
DEPARTAMENTO **INNER JOIN** FUNCIONARIO

SQL - CONSULTA DE DADOS



Consultas Joins – Inner Join

SELECT

DEPARTAMENTO.IDDEPARTAMENTO
, DEPARTAMENTO.NOME
, FUNCIONARIO.IDFUNCIONARIO
, FUNCIONARIO.NOME

FROM

DEPARTAMENTO

INNER JOIN FUNCIONARIO **ON**

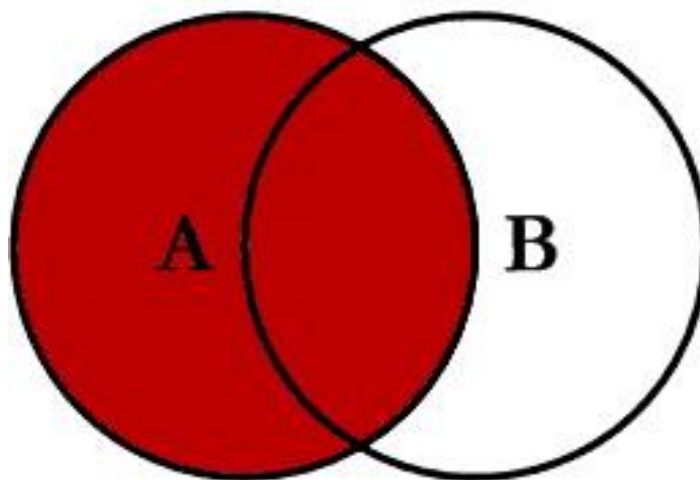
DEPARTAMENTO.IDDEPARTAMENTO = FUNCIONARIO.IDDEPARTAMENTO;

Iddepartamento	Nome	Idfuncionario	nome
1	Administração	1	João
2	Marketing	2	Maria
2	Marketing	3	Marcia
3	Financeiro	4	Ana
3	Financeiro	5	Lucas
5	Logística	6	Antonio

SQL - CONSULTA DE DADOS

Left Join

As junções externas LEFT JOIN se caracterizam por uma seleção que retorna todos os dados da tabela da ESQUERDA (left) e apenas os dados que atendem a **condição de junção** da tabela da DIREITA.



```
SELECT <select_list>  
FROM TableA A  
LEFT JOIN TableB B  
ON A.Key = B.Key
```

SQL - CONSULTA DE DADOS



Left Joins

Exemplo:

Tabela **Departamento**

IDDEPARTAMENTO	NOME
1	Administração
2	Marketing
3	Financeiro
4	Cobrança
5	Logística
6	manutenção

Tabela **Funcionário**

IDFUNCIONARIO	NOME	IDDEPARTAMENTO
1	João	1
2	Maria	2
3	Marcia	2
4	Ana	3
5	Lucas	3
6	Antonio	5
7	Karina	
8	Carlos	

condição de junção
DEPARTAMENTO LEFT JOIN FUNCIONARIO

SQL - CONSULTA DE DADOS



Left Joins

Exemplo:

Tabela Departamento

IDDEPARTAMENTO	NOME
1	Administração
2	Marketing
3	Financeiro
4	Cobrança
5	Logística
6	manutenção

Tabela Funcionário

IDFUNCIONARIO	NOME	IDDEPARTAMENTO
1	João	1
2	Maria	2
3	Marcia	2
4	Ana	3
5	Lucas	3
6	Antonio	5
7	Karina	
8	Carlos	

condição de junção
DEPARTAMENTO **LEFT JOIN** FUNCIONARIO

SQL - CONSULTA DE DADOS



Consultas Joins – Left Join

SELECT

DEPARTAMENTO.IDDEPARTAMENTO
, DEPARTAMENTO.NOME
, FUNCIONARIO.IDFUNCIONARIO
, FUNCIONARIO.NOME

FROM

DEPARTAMENTO

LEFT JOIN FUNCIONARIO **ON**

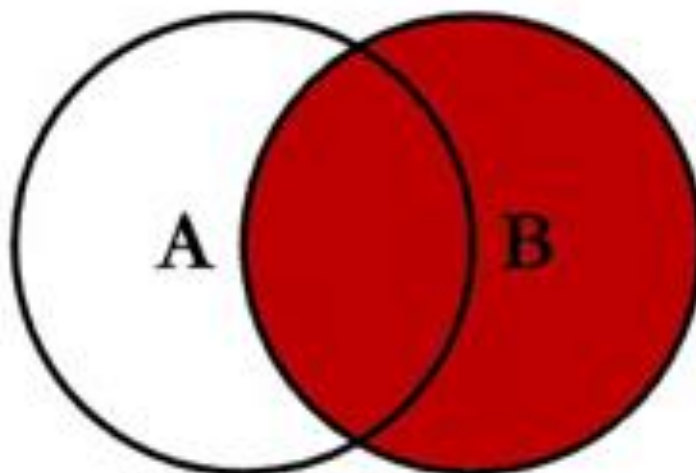
DEPARTAMENTO.IDDEPARTAMENTO = FUNCIONARIO.IDDEPARTAMENTO;

Iddepartamento	Nome	Idfuncionario	nome
1	Administração	1	João
2	Marketing	2	Maria
2	Marketing	3	Marcia
3	Financeiro	4	Ana
3	Financeiro	5	Lucas
4	Cobrança		
5	Logística	6	Antonio
6	Manutenção		

SQL - CONSULTA DE DADOS

Right Join

As junções externas RIGHT OUTER JOIN se caracterizam por uma seleção que retorna todos os dados da tabela da DIREITA (right) e apenas os dados que atendem a **condição de junção** da tabela da ESQUERDA.



```
SELECT <select_list>  
FROM TableA A  
RIGHT JOIN TableB B  
ON A.Key = B.Key
```

SQL - CONSULTA DE DADOS



Right Joins

Exemplo:

Tabela **Departamento**

IDDEPARTAMENTO	NOME
1	Administração
2	Marketing
3	Financeiro
4	Cobrança
5	Logística
6	manutenção

Tabela **Funcionário**

IDFUNCIONARIO	NOME	IDDEPARTAMENTO
1	João	1
2	Maria	2
3	Marcia	2
4	Ana	3
5	Lucas	3
6	Antonio	5
7	Karina	
8	Carlos	

condição de junção
DEPARTAMENTO RIGHT JOIN FUNCIONARIO

SQL - CONSULTA DE DADOS



Right Joins

Exemplo:

Tabela Departamento

IDDEPARTAMENTO	NOME
1	Administração
2	Marketing
3	Financeiro
4	Cobrança
5	Logística
6	manutenção

Tabela Funcionário

IDFUNCIONARIO	NOME	IDDEPARTAMENTO
1	João	1
2	Maria	2
3	Marcia	2
4	Ana	3
5	Lucas	3
6	Antonio	5
7	Karina	
8	Carlos	

condição de junção
DEPARTAMENTO **RIGHT JOIN** FUNCIONARIO

SQL - CONSULTA DE DADOS



Consultas Joins – Right Join

SELECT

DEPARTAMENTO.IDDEPARTAMENTO
, DEPARTAMENTO.NOME
, FUNCIONARIO.IDFUNCIONARIO
, FUNCIONARIO.NOME

FROM

DEPARTAMENTO

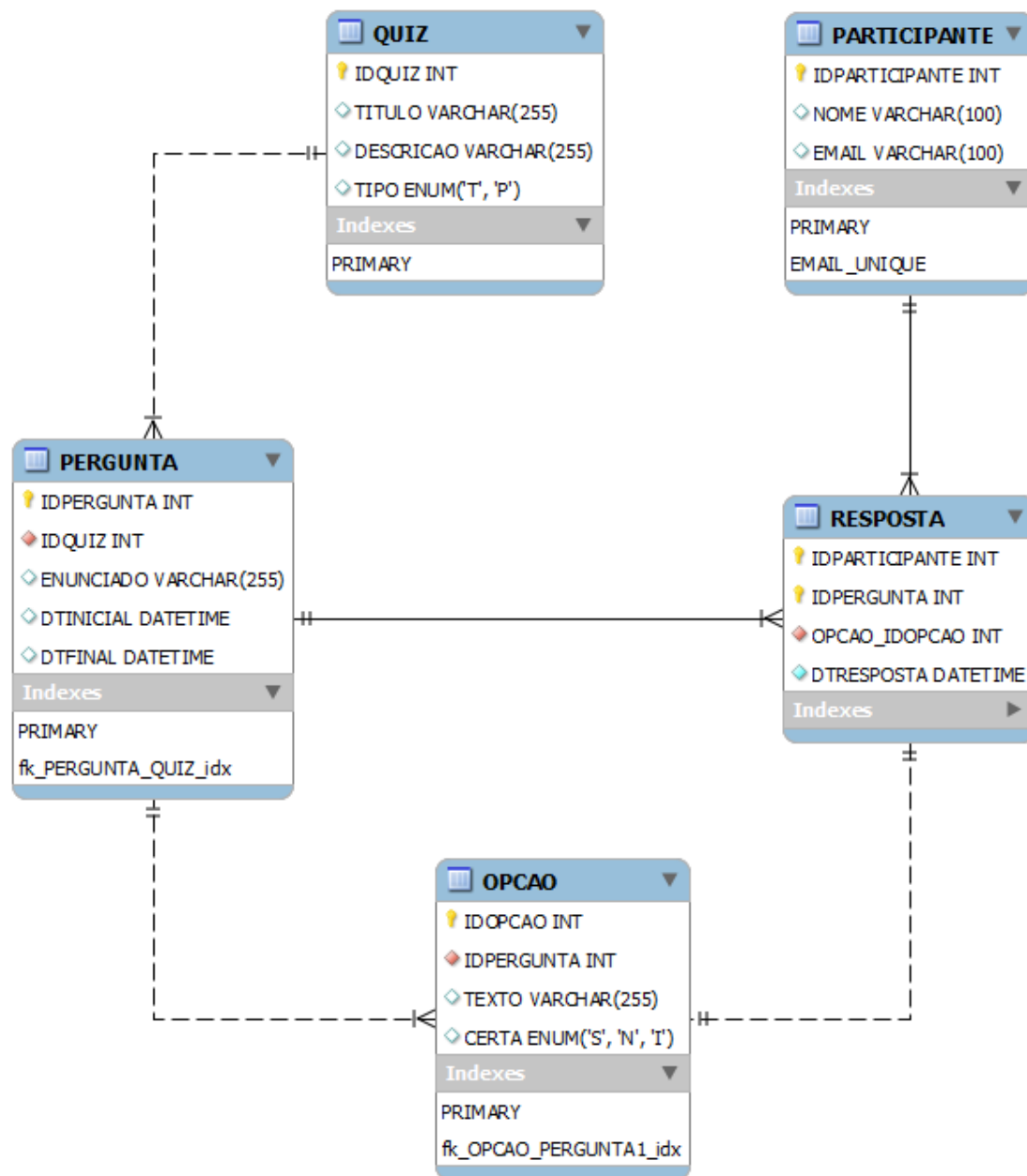
RIGHT JOIN FUNCIONARIO **ON**

DEPARTAMENTO.IDDEPARTAMENTO = FUNCIONARIO.IDDEPARTAMENTO;

Iddepartamento	Nome	Idfuncionario	nome
1	Administração	1	João
2	Marketing	2	Maria
2	Marketing	3	Marcia
3	Financeiro	4	Ana
3	Financeiro	5	Lucas
4	Logística	6	Antonio
		7	Karina
		8	Carlos

Exercícios de Revisão

DER



Exercício 1



Escreva os comandos SQL para listar o QUIZ com IDQUIZ = 2, as perguntas que pertencem esse QUIZ e as opções de resposta.

Resposta



```
SELECT
    QUIZ.IDQUIZ
    , QUIZ.TITULO
    , PERGUNTA.IDPERGUNTA
    , PERGUNTA,ENUCIADO
    , OPCAOS.IDOPCAO
    , OPCAOS.TEXTOS
    , OPCAOS.CERTAS
FROM
    QUIZ
    INNER JOIN PERGUNTA ON
    PERGUNTA.IDQUIZ
    INNER JOIN OPCAOS ON
    OPCAOS.IDPERGUNTA = PERGUNTA.IDPERGUNTA
WHERE
    QUIZ.IDQUIZ = 2;
```

Exercício 2



Escreva os comandos SQL para listar todas as participações, coloque o nome e o e-mail do participante, a resposta escolhida, o enunciado da pergunta e o título e a descrição do QUIZ

Resposta



SELECT

```
PARTICIPANTE.IDPARCIPANTE  
, PARTICIPANTE.NOME  
, PARTICIPANTE.EMAIL  
, RESPOSTA.OPCAO_IDOPCAO  
, QUIZ.TITULO  
, PERGUNTA.ENUNCIADO  
, OPCAOTEXTO
```

FROM

```
PARTICIPANTE  
INNER JOIN RESPOSTA ON  
PARTICIPANTE.IDPARTICIPANTE = REPOSTA.IDPARTICIPANTE  
INNER JOIN PERGUNTA ON  
RESPOSTA.IDPERGUNTA = PERGUNTA.IDPERGUNTA  
INNER JOIN QUIZ ON  
QUIZ.IDQUIZ = PERGUNTA.IDQUIZ  
INNER JOIN OPCAOTEXTO ON  
OPCAOTEXTO.IDOPCAO = RESPOSTA.OPCAO_IDOPCAO;
```

Exercício 3



Escreva os comandos SQL para listar somente as respostas certas, desde que o QUIZ seja do tipo “T”. Liste o código da pergunta, o enunciado, e todos os dados da opção.

Resposta



```
SELECT
    PERGUNTA.IDPERGUNTA
    , PERGUNTA.ENUNCIADO
    , OPCAOS.IDOPCAO
    , OPCAOS.TEXTOS
FROM
    QUIZ
    INNER JOIN PERGUNTA ON
        PERGUNTA.IDQUIZ
    INNER JOIN OPCAOS ON
        OPCAOS.IDPERGUNTA = PERGUNTA.IDPERGUNTA
WHERE
    QUIZ.TIPO = 'T'
    AND OPCAOS.CERTA = 'S';
```

Exercício 4

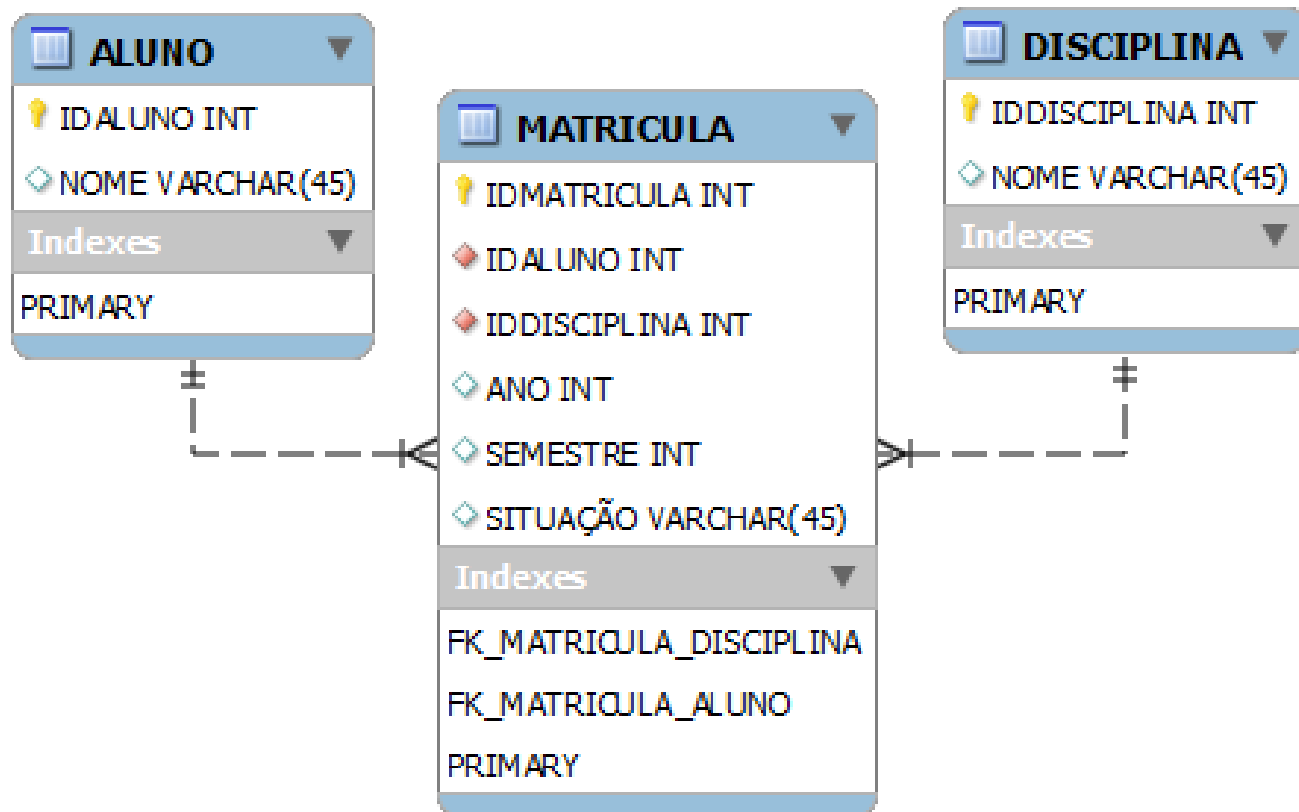
Para verificar qual pergunta teve mais participação foi utilizada uma consulta SQL que gerou o quadro abaixo, nesta tabela é listado o código da pergunta e a quantidade de participantes que respondeu essa questão.

IDPERGUNTA	COUNT(IDPARTICIPANTE)
1	36
2	73
3	22
4	128

Resposta



```
SELECT
    IDPERGUNTA
    , COUNT (IDPARTICIPANTE)
FROM
    RESPOSTA
GROUP BY
    IDPERGUNTA;
```



Dados de Exemplo



ALUNO	
IDALUNO	NOME
1	ARIANO VILAR SUASSUNA
2	CECÍLIA BENEVIDES DE CARVALHO MEIRELES
3	MANUEL CARNEIRO DE SOUZA BANDEIRA FILHO

DISCIPLINA	
IDDISCIPLINA	NOME
1	PRODUÇÃO DA LEITURA E DA ESCRITA
2	INTRODUÇÃO AOS ESTUDOS LITERÁRIOS
3	ANTROPOLOGIA TEOLÓGICA

MATRICULA					
IDMATRICULA	IDALUNO	IDDISCIPLINA	ANO	SEMESTRE	SITUAÇÃO
1	1	1	2015	1	CONCLUÍDO
2	2	1	2015	1	CONCLUÍDO
3	1	2	2015	1	CONCLUÍDO
4	1	3	2015	2	MATRICULADO
5	2	1	2015	2	MATRICULADO
6	3	1	2015	1	CONCLUÍDO
7	3	2	2015	2	MATRICULADO
8	3	3	2015	1	CANCELADO
9	3	3	2015	2	MATRICULADO

Exercício 6



Crie uma consulta SQL que liste o código identificador da matrícula, o código identificador da disciplina, o nome da disciplina, o código identificador do aluno, o nome do aluno, o ano da matrícula, o semestre da matrícula e a situação da matrícula. Liste apenas as matrículas com ano igual a 2015 e semestre igual a 1.

Resposta



SELECT

```
MATRICULA.IDMATRICULA  
, DISCIPLINA.IDDISCIPLINA  
, DISCIPLINA.NOME  
, ALUNO.IDALUNO  
, ALUNO.NOME  
, MATRICULA.ANO  
, MATRICULA.SEMESTRE  
, MATRICULA.SITUACAO
```

FROM

```
MATRICULA  
INNER JOIN DISCIPLINA ON  
DISCIPLINA.IDDISCIPLINA = MATRICULA.IDDISCIPLINA  
INNER JOIN ALUNO ON  
ALUNO.IDALUNO = MATRICULA.IDDALUNO
```

WHERE

```
MATRICULA.ANO = 2015  
AND MATRICULA.SEMESTRE = 1;
```

Exercício 7



Crie uma consulta SQL que liste o código identificador do aluno, o nome do aluno, o código identificador da disciplina, o nome da disciplina e a situação da matrícula. Liste apenas as matrículas com ano igual a 2015 e semestre igual a 2 e a disciplina “ANTROPOLOGIA TEOLÓGICA”, ordenado pelo nome do aluno.

Resposta



SELECT

```
MATRICULA.IDMATRICULA  
, DISCIPLINA.IDDISCIPLINA  
, DISCIPLINA.NOME  
, ALUNO.IDALUNO  
, ALUNO.NOME  
, MATRICULA.ANO  
, MATRICULA.SEMESTRE  
, MATRICULA.SITUACAO
```

FROM

```
MATRICULA  
INNER JOIN DISCIPLINA ON  
DISCIPLINA.IDDISCIPLINA = MATRICULA.IDDISCIPLINA  
INNER JOIN ALUNO ON  
ALUNO.IDALUNO = MATRICULA.IDDALUNO
```

WHERE

```
MATRICULA.ANO = 2015  
AND MATRICULA.SEMESTRE = 2  
AND DISCIPLINA.NOME LIKE '%ANTROPOLOGIA TEOLÓGICA%'
```

ORDER BY ALUNO.NOME;

Exercício 8

Crie uma consulta para listar a quantidade de matrícula, quantidade de disciplinas, quantidade de alunos, agrupado por ano, semestre e situação, conforme imagem abaixo:

CONSULTA					
ANO	SEMESTRE	SITUACAO	COUNT(IDMATRICULA)	COUNT(IDALUNO)	COUNT(DISCIPLINA)
2015	1	CONCLUÍDO	4	3	2
2015	1	CANCELADO	1	1	1
2015	2	MATRICULADO	4	3	3

Resposta



SELECT

```
MATRICULA.ANO  
, MATRICULA.SEMESTRE  
, MATRICULA.SITUACAO  
, COUNT (MATRICULA.IDMATRICULA)  
, COUNT (MATRICULA.IDALUNO)  
, COUNT (MATRICULA.IDDISCIPLINA)
```

FROM

```
MATRICULA
```

GROUP BY

```
MATRICULA.ANO  
, MATRICULA.SEMESTRE  
, MATRICULA.SITUACAO
```