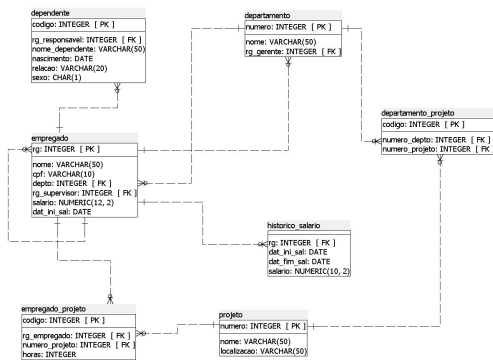


ORDENANDO RESULTADOS

Preparando base de dados de atividades

- Importe o banco de dados de exemplo completo com os relacionamentos aplicando o script denominado “empresa_crpfk.sql”

Modelo



Dados nas tabelas

EMPREGADO_PROJETO				HISTORICO_SALARIO			
CODIGO	RG_EMPREGADO	NUMERO_PROJETO	HORAS	RG	DAT_INI_SAL	DAT_FIM_SAL	SALARIO
1	2020	5	10	1010	01/01/2010	30/11/2010	2000.00
2	2020	10	25	1010	01/12/2010	30/04/2011	4000.00
3	3030	5	35	2020	01/05/2007	31/12/2007	2500.00
4	4040	20	50	2020	01/01/2008	30/12/2010	4000.00
5	5050	20	35	4040	01/08/2008	31/10/2009	1500.00
6	8080	5	70	4040	01/11/2009	30/09/2010	2500.00
				4040	01/10/2010	31/10/2012	3500.00
				7070	01/01/2008	31/07/2008	1000.00

DEPENDENTE				DEPARTAMENTO		
CODIGO	RG_RESPONSÁVEL	NOME_DEPENDENTE	NASCIMENTO	NUMERO	NOME	RG_GERENTE
1	1010	Jorge	1986-12-27	1	Contabilidade	1010
2	1010	Luiz	1979-11-18	2	Engenharia Civil	3030
3	2020	Fernanda Carla	1969-02-14	3	Engenharia Mecânica	2020
4	2020	Angelo	1995-02-10	4	Industrial	
5	3030	André	1990-01-01			
6	8080	Ana Maria	1980-06-30			
7	8080	Karla Cristina	1999-08-25			

EMPREGADO				DEPENDENTE		
RG	NOME	CPF	DEPTO	RG_SUPERVISOR	SALARIO	DAT_INI_SAL
1010	João Luiz	11111	1		6000.00	2011-05-01
2020	Fernanda	22222	1	1010	5500.00	2008-12-01
3030	Ricardo	33333	2	2020	2500.00	2009-08-01
4040	Jorge	44444	2	3030	3200.00	2010-10-01
5050	Renata	55555	2	3030	1300.00	2012-02-01
6060	Luiz Renato	66666	3	2020	3000.00	2012-05-01
7070	Luiz Fernando	77777	3	6060	2000.00	2008-08-01
8080	João Antonio	88888	1	2020	3950.00	2010-07-01

DEPENDENTE			EMPREGADO		
NUMERO	NOME	LOCALIZACAO	NUMERO	NOME	LOCALIZACAO
1	Contabilidade	São Paulo	1	João Luiz	São Paulo
2	Engenharia Civil	Rio Claro	2	Fernanda	Rio Claro
3	Engenharia Mecânica	Campanas	3	Ricardo	Campanas
4	Industrial	Vitória	4	Jorge	Vitória

DQL (Data Query Language)

- Select: ordenando resultados (múltiplas linhas)

```
SELECT [all | distinct] select_list
FROM <table_name>
ORDER BY <column_name> [Desc], <column_name2> [Desc]...
```

Execução de código:

```
select nome, numero, rg_gerente from departamento order by nome;
OU
select * from departamento order by nome;
OU
select * from departamento order by nome desc;
```

Exercícios Order by

1. Selecione código, nome e relação dos dependentes ordenados pelo nome do dependente.
2. Selecione todos os atributos da tabela projeto, e apresente os registros ordenados por localização.
3. Selecione todos os atributos da tabela projeto e apresente os registros ordenados primeiro por nome, e depois por localização.
4. selecione todos os atributos da tabela dependente e apresente os registros ordenados pelo rg do responsável de forma decrescente, e sub-ordenados pelo nome do dependente por ordem decrescente.
5. mostre todos registros da tabela empregados apresentando seus salários e código do departamento, ordenados do maior para o menor salário apenas onde ocorrem salários maiores que 2500.

CRIANDO JUNÇÕES ENTRE TABELAS

Junção → Join

1. As chaves primárias e estrangeiras cumprem papel fundamental no relacionamento entre as tabelas.
2. Como veremos a seguir as consultas com 'join' possibilitam:
 1. Acessar mais de uma tabela em uma única consulta.
 2. Requerer dados dispersos em mais de uma tabela agrupando o resultado final em uma única instrução.

DQL (Data Query Language)

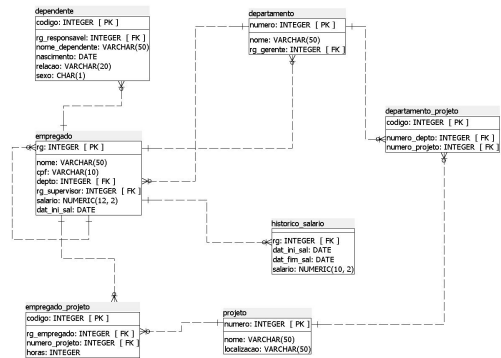
- Select: Junções (múltiplas linhas)

```
SELECT [all | distinct ] select_list
[FROM <table_name1>
[INNER] JOIN <table_name2>
On (<table_name1>.<column_1> = <table_name2>.<column_2>)
```

Execução de código:

```
Select nome, Rg, RG_EMPREGADO
from empregado_projeto
inner join
empregado on
(empregado_projeto.rg_empregado = empregado.rg)
```

Relacionamento entre as tabelas!



Select sem Join (Produto Cartesiano)

(select sem join)

CODIGO	RG_EMPREGADO	NUMERO_PROJETO	HORAS
1	2020	5	20
2	2020	10	25
3	3030	5	35
4	4040	20	50
5	5050	20	35
6	8080	5	70

Select nome, rg, rg_empregado
from empregado_projeto, empregado

RG	NOME	CPF	DEPTO	RG_SUPERVISOR	SALARIO	DAT_INI_SAL
1010	João Luiz	11111	1		6000.00	2011-05-01
2020	Fernanda	22222	1	1010	5500.00	2008-12-01
3030	Ricardo	33333	2	2020	2300.00	2009-08-01
4040	Jorge	44444	2	3030	3200.00	2010-10-01
5050	Renata	55555	2	3030	1300.00	2012-02-01
6060	Luiz Renato	66666	3	2020	3000.00	2012-05-01
7070	Luiz Fernando	77777	3	6060	2000.00	2008-08-01
8080	João Antonio	88888	1	2020	3950.00	2010-07-01

nome	rg	RG_EMPREGADO
João Luiz	1010	2020
João Luiz	1010	2020
João Luiz	1010	3030
João Luiz	1010	4040
João Luiz	1010	5050
João Luiz	1010	8080
Fernanda	2020	2020
Fernanda	2020	3030
Fernanda	2020	4040
Fernanda	2020	5050
Fernanda	2020	8080
Ricardo	3030	2020
Ricardo	3030	2020

Select com Join e sem restrições em "on" (Produto cartesiano)

(com cláusula join sem restrições em "on")

CODIGO	RG_EMPREGADO	NUMERO_PROJETO	HORAS
1	2020	5	10
2	2020	10	25
3	3030	5	35
4	4040	20	50
5	5050	20	35
6	8080	5	70

```
Select nome,rg,rg_empregado
from empregado_projeto
inner join empregado
on (true);
```

RG	NOME	CPF	DEPTO	RG_SUPERVISOR	SALARIO	DAT_INI_SAL
1010	João Luiz	11111	1		6000.00	2011-05-01
2020	Fernanda	22222	1	1010	5500.00	2008-12-01
3030	Ricardo	33333	2	2020	2300.00	2009-08-01
4040	Jorge	44444	2	3030	3200.00	2010-10-01
5050	Renata	55555	2	3030	1300.00	2012-02-01
6060	Luiz Renato	66666	3	2020	3000.00	2012-05-01
7070	Luiz Fernando	77777	3	6060	2000.00	2008-08-01
8080	João Antonio	88888	1	2020	3950.00	2010-07-01

nome	rg	RG_EMPREGADO
João Luiz	1010	2020
João Luiz	1010	2020
João Luiz	1010	3030
João Luiz	1010	4040
João Luiz	1010	5050
João Luiz	1010	8080
Fernanda	2020	2020
Fernanda	2020	2020
Fernanda	2020	3030
Fernanda	2020	4040
Fernanda	2020	5050
Fernanda	2020	8080
Ricardo	3030	2020
Ricardo	3030	2020

Select com Join e com restrições em "on" (Intersecção entre tabelas com base num determinado atributo)

Join (com cláusula 'on' = resultado de correspondência)

CODIGO	RG_EMPREGADO	NUMERO_PROJETO	HORAS
1	2020	5	10
2	2020	10	25
3	3030	5	35
4	4040	20	50
5	5050	20	35
6	8080	5	70

```
Select nome,rg,rg_empregado
from empregado_projeto
inner join empregado
on (empregado_projeto.rg_empregado = empregado.rg)
```

RG	NOME	CPF	DEPTO	RG_SUPERVISOR	SALARIO	DAT_INI_SAL
1010	João Luiz	11111	1		6000.00	2011-05-01
2020	Fernanda	22222	1	1010	5500.00	2008-12-01
3030	Ricardo	33333	2	2020	2300.00	2009-08-01
4040	Jorge	44444	2	3030	3200.00	2010-10-01
5050	Renata	55555	2	3030	1300.00	2012-02-01
6060	Luiz Renato	66666	3	2020	3000.00	2012-05-01
7070	Luiz Fernando	77777	3	6060	2000.00	2008-08-01
8080	João Antonio	88888	1	2020	3950.00	2010-07-01

nome	rg	RG_EMPREGADO
Fernanda	2020	2020
Fernanda	2020	2020
Ricardo	3030	3030
Jorge	4040	4040
Renata	5050	5050
João Antonio	8080	8080

Exercícios - junção

1. Crie uma junção entre a tabela departamento_projeto e a tabela projeto mostrando o nome do projeto e campos de junção
2. Crie uma junção entre a tabela empregado_projeto, projeto e empregado, mostrando os nomes dos empregados que participam de algum projeto, e nomes destes projetos.
3. Crie uma junção , mostrando os nomes dos empregados que participam de algum projeto e a quantidade de horas trabalhada em cada projeto. Use o parâmetro 'as' para renomear as colunas para "funcionário", "projeto" e "horas trabalhadas".
4. Crie uma junção que retorne o código do dependente, nome do empregado, nome do dependente e relação dos dependentes ordenados pelo nome do empregado:
5. Crie uma junção que mostre todos nomes de empregados e seu histórico de salários (DAT_INI_SAL,DAT_FIM_SAL, salario).
6. Crie uma junção , mostrando os nomes dos empregados que participam de algum projeto e a quantidade de horas trabalhada em cada projeto. Use o parâmetro 'as' para renomear as colunas para "funcionário", "projeto" e "horas trabalhadas" exibindo por fim, apenas os resultados para projeto que possui nome igual a "Financeiro".
7. Crie uma junção que mostre todos empregados e seu histórico de salários e o departamento de que faz parte.

Cont. Exercícios - Junção

8. Crie uma junção que mostre todos empregados e seu histórico de salários (data inicial, data final e valor do salario) além do departamento de que o empregado faz parte, ordenado do maior para o menor salario com resultados apenas para os salários da tabela histórico salario maiores que 2500.
9. Crie uma junção que mostre todos empregados e seu histórico de salários (data inicial, data final e valor do salario), departamento de que faz parte, e projetos que participa, ordenado do maior para o menor salario com resultados apenas para os salários da tabela histórico salario maiores que 2000.
10. Crie uma junção que mostre todos empregados e seu histórico de salários (data inicial, data final e valor do salario), departamento de que faz parte, e projetos que participa, ordenado do maior para o menor salario com resultados apenas para os salários da tabela histórico salario maiores que 2000 e menores que 4000 com dependentes do sexo feminino.

Referencias

- CAP 06 - NAVATHE