

## 5. SQL

### - Conjunto de Operações para Manipulação de Dados -

#### ❑ Predicados com operações sobre strings

##### ➡ Identificação de padrão

⇒ %

⇒ Casa com qualquer *substring*

⇒ \_

⇒ Casa com qualquer caractere

##### ➡ Operador

⇒ *like*

##### ➡ Exemplos: (Cont.)

## 5. SQL

### - Conjunto de Operações para Manipulação de Dados -

#### □ Predicados com operações sobre strings (cont.)

##### ➡ Exemplos:

⇒ nome like 'inf%'

↪ Retorna strings que iniciam pela substring inf

⇒ nome like '%si\_'

↪ Retorna strings que contenham 'si' como substring e terminem com um caracter qualquer após 'si'

⇒ Listar todos empregados com sobrenome 'Rodrigues'

↪ Select nome from Empregado where nome like '%Rodrigues%'

⇒ Listar todos empregados com último sobrenome 'Almeida'

↪ Select nome from Empregado where nome like '%Almeida'

⇒ Listar todos empregados com primeiro nome 'Fernando'

↪ Select nome from Empregado where nome like 'Fernando%'

## 5. SQL

- Conjunto de Operações para Manipulação de Dados -

### ❏ Consultas com o operador de união

#### ↳ UNION

⇒ União de duas relações (consultas)

⇒ Sem repetições

#### ↳ UNION ALL

⇒ União de duas relações

⇒ Com repetições

#### ↳ Exemplo

⇒ Considere as seguintes relações

⇒ Empregado(matr, nome, ender, dt\_nasc, cpf, salário, lotação)

⇒ Dependente(nome\_dep, data-nasc, matr\_resp)

⇒ Liste o nome e data de nascimento de todos os funcionários e dependentes existentes na empresa

```
select nome,dt_nasc from Empregado UNION  
select nome,data_nasc from Dependente
```

## 5. SQL

- Conjunto de Operações para Manipulação de Dados -

### ❑ Consultas com o operador de interseção

#### ↳ INTERSECT

⇒ Interseção entre duas relações (consultas)

⇒ Sem repetições

#### ↳ INTERSECT ALL

⇒ Interseção entre duas relações

⇒ Com repetições

### ❑ Consultas com o operador de diferença

#### ↳ EXCEPT

⇒ Diferença entre duas relações (consultas)

⇒ Sem repetições

#### ↳ EXCEPT ALL

⇒ Diferença entre duas relações (consultas)

⇒ Com repetições

## 5. SQL

### - Conjunto de Operações para Manipulação de Dados-

#### ❑ EXCEPT E INTERSECT no MySQL:

Não existem estas cláusulas no MySQL

Em substituição, podemos usar as cláusulas “NOT IN” e “IN” (respectivamente), para substituir tais cláusulas:

#### **Exemplo (EXCEPT):**

Retorna em uma única resposta os registros que existam somente na tabela Cliente e não estejam em Funcionário:

#### **Usando EXCEPT)**

```
SELECT cpf, nome  
FROM Cliente EXCEPT (SELECT cpf, nome FROM Funcionario)
```

#### **No MySQL:**

```
SELECT cpf, nome  
FROM Cliente  
WHERE (cpf, nome) NOT IN (SELECT cpf, nome FROM Funcionario)
```

## 5. SQL

### - Conjunto de Operações para Manipulação de Dados-

#### ❑ EXCEPT E INTERSECT no MySQL:

##### **Exemplo (INTERSECT):**

Retorna em uma única resposta os registros que existam tanto na tabela Cliente como na tabela Funcionário:

##### **Usando INTERSECT)**

```
SELECT cpf, nome  
FROM Cliente INTERSECT (SELECT cpf, nome FROM Funcionario)
```

##### **No MySQL:**

```
SELECT cpf, nome  
FROM Cliente  
WHERE (cpf, nome) IN (SELECT cpf, nome FROM Funcionario)
```

## 5. SQL

- Conjunto de Operações para Manipulação de Dados -

### ❑ Exercícios

- Listar matrícula dos empregados que não possuem dependentes

```
Select e.matr  
From empregado e  
EXCEPT  
Select d.matr-resp  
From dependente d
```

- Listar matrícula dos empregados que possuem dependentes

```
Select e.matr  
From empregado e  
INTERSECT  
Select d.matr-resp  
From dependente d
```

### ❑ Consultas com consultas ordenadas

- ORDER BY coluna-resultado [ASC | DESC]  
                                  {, coluna-resultado [ASC | DESC] ...}
- Listar empregados ordenados por salário na ordem decrescente e por nome na ordem crescente