

# Bases de Dados

## Interrogações SQL – II (Subinterrogações)

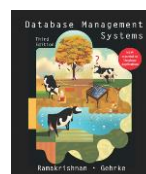
FCUL, Departamento de Informática

Ano Letivo 2015/2016

Ana Paula Afonso

## Sumário e Referências

- Sumário
  - Sub-Interrogações
    - Independentes (operadores IN e NOT IN)
    - Correlacionadas (operadores EXISTS e NOT EXISTS)
    - Múltiplas Sub-Interrogações
  - Comparações de conjuntos
    - Operadores ANY e ALL
  - Exemplos
    - Máximo, Intersecções e Divisões
- Referências
  - R. Ramakrishnan (**capítulo 5, secção 5.4**)



# Sub-Interrogações em Interrogações

- Uma interrogação pode conter outras interrogações
  - Um (ou mais) SELECT dentro de um SELECT
  - Também denominada *nested queries*
  - A interrogação que está embebida chama-se sub-interrogação

```
SELECT S.sname
FROM   Sailors S
WHERE  S.sid IN  ( SELECT R.sid
                  FROM Reserves R
                  WHERE R.bid = 103)
```

- Numa interrogação podem aparecer
  - Na cláusula WHERE (o mais frequente)

```
SELECT ...
FROM ...
WHERE ...OperadorConjunto (SELECT ... FROM ...)
OperadorConjunto: IN, NOT IN, EXISTS, NOT EXISTS, ANY, ALL
```

- Na Cláusula FROM (**não autorizados no âmbito desta disciplina**)
- Na cláusula HAVING (matéria futura)

# Tabelas de Exemplo

Sailors

<u>sid</u>	sname	rating	age
22	Dustin	7	45.0
29	Brutus	1	33.0
31	Lubber	8	55.5
32	Andy	8	25.5
58	Rusty	10	35.0
61	Horatio	7	35.0
71	Zorba	10	16.0
74	Horatio	9	35.0
85	Art	3	25.5
95	Bob	3	63.5

Boats

<u>bid</u>	bname	color
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

Reserves

<u>sid</u>	<u>bid</u>	<u>day</u>
22	101	10/10/96
58	103	11/12/96

## Interrogação com Operador IN

- Nomes dos marinheiros que reservaram o barco 103

```
SELECT S.sname
FROM   Sailors S
WHERE  S.sid IN ( SELECT R.sid
                  FROM Reserves R
                  WHERE R.bid = 103)
```

- A **sub-interrogação** devolve o conjunto dos identificadores dos marinheiros que reservaram o barco 103
- A interrogação seleciona apenas os marinheiros que pertencem ao conjunto da sub-interrogação

## Múltiplas Sub-Interrogações

- Nomes dos marinheiros que reservaram barcos vermelhos

```
SELECT S.sname
FROM   Sailors S
WHERE  S.sid IN ( SELECT R.sid
                  FROM Reserves R
                  WHERE R.bid IN (SELECT B.bid
                                  FROM Boats B
                                  WHERE B.color = 'red'))
```

- A **2ª sub-interrogação** devolve os identificadores dos barcos vermelhos
- A **1ª sub-interrogação** devolve os identificadores dos marinheiros que reservaram barcos vermelhos
- A interrogação devolve o nome dos marinheiros que reservaram barcos vermelhos

## Interrogação com Operador NOT IN

- Nomes dos marinheiros que **não** reservaram barcos vermelho

```
SELECT S.sname
FROM Sailors S
WHERE S.sid NOT IN( SELECT R.sid
                     FROM Reserves R
                     WHERE R.bid IN ( SELECT B.bid
                                     FROM Boats B
                                     WHERE B.color = 'red'))
```

- Outras alternativas
  - **NOT IN** no segundo WHERE em vez de no primeiro  
Nome dos marinheiros que reservaram barcos que não são vermelhos
  - **NOT IN** nos dois WHERE  
Nome dos marinheiros que não reservaram barcos que não são vermelhos  
Ou seja, que só reservaram barcos vermelhos ... juntamente com os que não fizeram qualquer reserva

© 2015 - Docentes SI - DI/FCUL

7

## Sub-Interrogações Correlacionadas

- Nomes dos marinheiros que reservaram o barco 103

```
SELECT S.sname
FROM Sailors S
WHERE EXISTS ( SELECT *
               FROM   Reserves R
               WHERE  R.bid = 103 AND
                     R.sid = S.sid )
```

- EXISTS verifica a existência de valores no resultado da sub-interrogação
- A condição na sub-interrogação tem em conta o marinheiro atual na interrogação principal
- A utilização de SELECT \* é recomendada nestas situações
- Outro caso
  - NOT EXISTS no WHERE: Marinheiros que não reservaram o barco 103

© 2015 - Docentes SI - DI/FCUL

8

## Comparação de Conjuntos com ANY

- Marinheiros cujo *rating* é maior que o rating de **algum** dos marinheiros chamados *Horatio*

```
SELECT S.sid
FROM Sailors S
WHERE S.rating > ANY
      ( SELECT S2.rating
        FROM Sailors S2
        WHERE S2.sname = 'Horatio' )
```

- (1) Sub-interrogação devolve os ratings dos marinheiros Horatio
- Interrogação seleciona os marinheiros cujo rating seja superior a algum em (1)
- Se sub-interrogação devolve **conjunto vazio** então > ANY (empty) = false
- SOME é sinónimo de ANY

© 2015 - Docentes SI - DI/FCUL

9

## Comparação de Conjuntos com ALL

- Marinheiros cujo *rating* é maior que o rating de **todos** os marinheiros chamados *Horatio*

```
SELECT S.sid
FROM Sailors S
WHERE S.rating > ALL
      ( SELECT S2.rating
        FROM Sailors S2
        WHERE S2.sname = 'Horatio' )
```

- (1) Sub-interrogação devolve os ratings dos marinheiros Horatio
- Interrogação seleciona os marinheiros cujo rating seja superior a todos em (1)
- Se sub-interrogação devolve **conjunto vazio** então > ALL (empty) = true
- Se não há Horatios então devolve todos os marinheiros

© 2015 - Docentes SI - DI/FCUL

10

## Exemplo de Escolha do Valor Máximo

- Marinheiros com o maior *rating*?

```
SELECT S.sid
FROM Sailors S
WHERE S.rating >= ALL (
    SELECT S2.rating
    FROM Sailors S2)
```

- Observações

- WHERE ... IN .... equivalente a WHERE ... = ANY ...
- WHERE ... NOT IN .... equivalente a WHERE ... <> ANY ...

## Exemplo de Interseção de Conjuntos

- Quais os nomes dos marinheiros que reservaram (pelo menos) um barco verde e (um) vermelho?

```
SELECT S.sname
FROM Sailors S, Reserves R, Boats B
WHERE S.sid = R.sid AND R.bid = B.bid
    AND B.color = 'red'
    AND S.sid IN (
    SELECT S2.sid
    FROM Sailors S2, Boats B2, Reserves R2
    WHERE S2.sid = R2.sid
        AND R2.bid = B2.bid
        AND B2.color = 'green' )
```

- Observação: Equivalente a INTERSECT, e NOT IN equivalente a EXCEPT

## Exemplo de Divisão

- Nomes dos marinheiros que reservaram **todos** os barcos

```
SELECT S.sname
FROM Sailors S
WHERE NOT EXISTS (
    SELECT B.bid
    FROM Boats B
    WHERE NOT EXISTS (
        SELECT R. bid
        FROM Reserves R
        WHERE R.bid = B.bid
            AND R.sid = S.sid ))
```

Obs. All x: p(x)  $\Leftrightarrow$  Not Exists x: Not p(x)

© 2015 - Docentes SI - DI/FCUL

13

## Exemplo de Divisão

1. Nome dos marinheiros, para os quais
  - Qualquer que seja o barco
  - Existe uma reserva para esse barco e para esse marinheiro
2. Nome dos marinheiros:
 

$\forall \text{ barco: } (\exists \text{ reserva: } R.\text{bid} = B.\text{bid} \text{ AND } R.\text{sid} = S.\text{Sid})$

Nota:  $\forall x: p(x) \Leftrightarrow \sim \exists x: \sim p(x)$
3. Nomes dos marinheiros:
 

$\sim \exists \text{ barco: } \sim \exists \text{ reserva: } R.\text{bid} = B.\text{bid} \text{ AND } R.\text{sid} = S.\text{sid}$

```
SELECT S.sname
FROM Sailors S
WHERE NOT EXISTS (SELECT B.bid
    FROM Boats B
    WHERE NOT EXISTS (SELECT R. bid
        FROM Reserves R
        WHERE R.bid = B.bid AND R.sid = S.sid ))
```

© 2015 - Docentes SI - DI/FCUL

14

## Exemplo de Divisão com Restrição

---

- Nomes dos marinheiros que reservaram **todos** os barcos **vermelhos**

```
SELECT S.sname
FROM Sailors S
WHERE NOT EXISTS (
    SELECT B.bid
    FROM Boats B
    WHERE B.color = 'red'
        AND NOT EXISTS (
            SELECT R. bid
            FROM Reserves R
            WHERE R.bid = B.bid
                AND R.sid = S.sid ))
```