



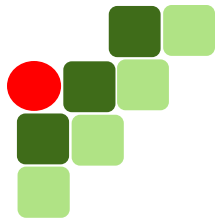
INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
SÃO PAULO  
Campus Araraquara

# Banco de Dados II

Cristiane Yaguinuma

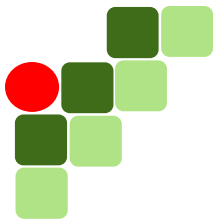
cristiane.yaguinuma@ifsp.edu.br

- Consultas aninhadas – subconsultas



# Roteiro da aula

- ▶ Consultas aninhadas
  - Subconsulta de única tupla
  - Subconsultas de múltiplas tuplas
    - Operadores IN, ANY, ALL, EXISTS
  
- ▶ Exercícios



# Consultas aninhadas

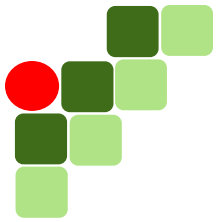
- ▶ Recuperar os sobrenomes dos empregados que ganham salário maior que o do empregado com sobrenome Gates

CONSULTA PRINCIPAL

Quais empregados possuem salário maior que o salário de Gates?

SUBCONSULTA

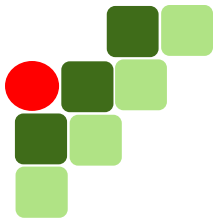
Qual o salário de Gates?



# Consultas aninhadas

```
SELECT <colunas>  
FROM <tabelas>  
WHERE <expr> <operador>  
      (SELECT <colunas>  
        FROM <tabelas>  
      ) ;
```

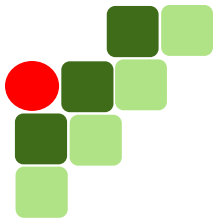
- ▶ A consulta aninhada executa antes da consulta principal
- ▶ O resultado da consulta aninhada é usado pela consulta principal



# Consultas aninhadas

- Recuperar os sobrenomes dos empregados que ganham salário maior que o do empregado com sobrenome Gates

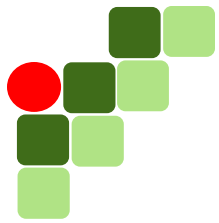
```
SELECT last_name, salary
FROM employees
WHERE salary >
      (SELECT salary
       FROM employees
       WHERE last_name = 'Gates') ;
```



# Consultas aninhadas

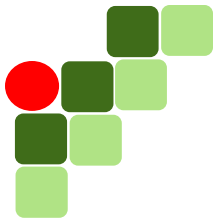
- ▶ Recuperar os sobrenomes dos empregados que ganham salário maior que o do empregado com sobrenome Gates
- Outra solução sem consulta aninhada

```
SELECT E.last_name, E.salary
FROM EMPLOYEES E, EMPLOYEES Gates
WHERE E.salary > Gates.salary
AND Gates.last_name = 'Gates'
ORDER BY E.salary DESC;
```



# Tipos de consultas aninhadas

- ▶ Subconsulta de única tupla  
(single-row subquery)
  - Retorna somente uma tupla como resultado
- ▶ Subconsulta de múltiplas tuplas  
(multiple-row subquery)
  - Retorna um conjunto de tuplas como resultado



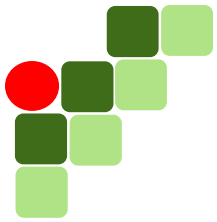
# Single-row subquery

```
SELECT last_name, salary
FROM employees
WHERE salary >
      (SELECT salary
       FROM employees
       WHERE last_name = 'Gates') ;
```

## ► Utilizar operadores

- =, <>
- <, <=
- >, >=

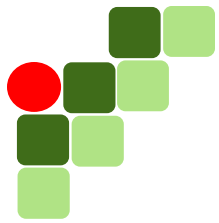




# Single-row subquery

- Recuperar o nome e a função do empregado que recebe o menor salário da empresa

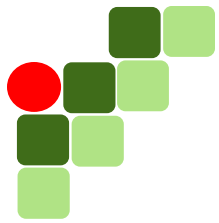
```
SELECT first_name, last_name, job_id, salary
FROM employees
WHERE salary =
      (SELECT MIN(salary)
       FROM employees);
```



# Single-row subquery

- Para cada departamento, recuperar a média de salário, sendo que esta deve ser maior que o salário médio do departamento de nome 'Sales'

```
SELECT department_id, AVG(salary)
FROM employees
GROUP BY department_id
HAVING AVG(salary) >
        (SELECT AVG(salary)
         FROM employees E JOIN departments D
          ON E.department_id = D.department_id
          WHERE department_name = 'Sales');
```

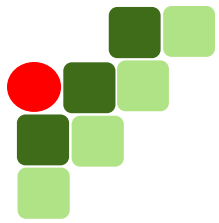


# Single-row subquery

- ▶ Recuperar os sobrenomes dos empregados que ganham os mesmos salários que os dos empregados com função IT\_PROG

```
SELECT LAST_NAME, SALARY
FROM EMPLOYEES
WHERE SALARY =
        (SELECT SALARY
         FROM EMPLOYEES
         WHERE JOB_ID = 'IT_PROG') ;
```

- ▶ **ERRO: a subconsulta retorna mais de um resultado!**



# Multiple-row subquery

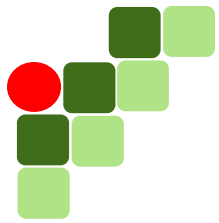
- ▶ Recuperar os sobrenomes dos empregados que ganham os mesmos salários que os dos empregados com função IT\_PROG

## CONSULTA PRINCIPAL

Quais empregados ganham os mesmos salários que os dos empregados com função IT\_PROG?

## SUBCONSULTA

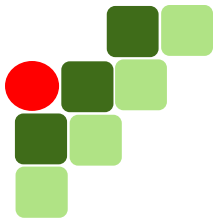
Quais os salários dos empregados com função IT\_PROG?



# Multiple-row subquery

- ▶ Subconsultas retornam **mais de uma tupla**
- ▶ Devem-se usar operadores de comparação de conjuntos de tuplas:

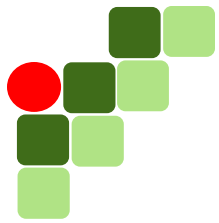
Operador	Significado
IN	Verifica se valor pertence ao conjunto de valores retornados pela subconsulta
ANY	Valor é comparado com cada valor retornado pela subconsulta Retorna TRUE se a comparação for TRUE para ALGUM dos valores retornados pela subconsulta
ALL	Valor é comparado com todos valores retornados pela subconsulta Retorna TRUE se a comparação for TRUE para TODOS os valores retornados pela subconsulta



# Operador IN

- Recuperar os sobrenomes dos empregados que ganham os mesmos salários que os dos empregados com função IT\_PROG

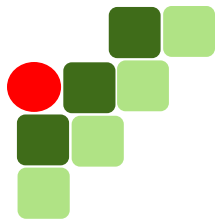
```
SELECT last_name, salary
FROM employees
WHERE job_id <> 'IT_PROG'
AND salary IN
    (SELECT salary
     FROM employees
     WHERE job_id = 'IT_PROG');
```



# Operador NOT IN

- Recuperar os sobrenomes dos empregados que ganham salários diferentes dos salários dos empregados com função IT\_PROG

```
SELECT last_name, salary
FROM employees
WHERE salary NOT IN
      (SELECT salary
       FROM employees
       WHERE job_id = 'IT_PROG');
```

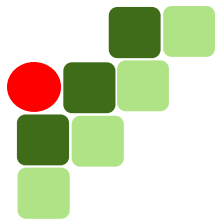


# Multiple-row subquery

- ▶ Recuperar os nomes de departamentos que não estejam localizados no país 'United States of America'

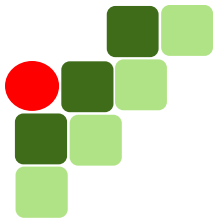
```
SELECT department_name
FROM departments
WHERE location_id NOT IN
    (SELECT L.location_id
     FROM locations L JOIN countries C
     USING (country_id)
     WHERE C.country_name =
           'United States of America');
```





# Operador ANY

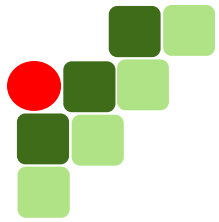
- ▶ Usado para comparar um valor  $v$  com um conjunto de valores retornados por uma subconsulta, usando operadores:
  - $=$ ,  $<>$
  - $<$ ,  $<=$
  - $>$ ,  $>=$
- ▶ Retorna TRUE se algum dos valores retornados pela subconsulta satisfaz a comparação



# Operador ANY

- Recuperar os sobrenomes dos empregados que ganham salários menores que o salário de algum empregado com função IT\_PROG

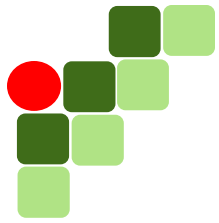
```
SELECT last_name, salary
FROM employees
WHERE job_id <> 'IT_PROG'
AND salary < ANY
    (SELECT salary
     FROM employees
     WHERE job_id = 'IT_PROG');
```



# Operador ANY

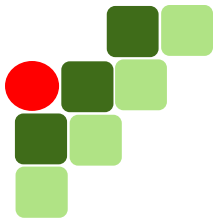
- ▶ Recuperar os sobrenomes dos empregados que ganham salários maiores que o salário de algum gerente de departamento

```
SELECT last_name, salary
FROM employees
WHERE salary > ANY
      (SELECT salary
        FROM employees E, departments D
        WHERE E.employee_id = D.manager_id);
```



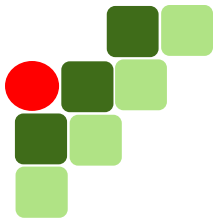
# Operador ANY

- ▶ **< ANY: valores menores que o valor máximo do conjunto de tuplas**
- ▶ **> ANY: valores maiores que o valor mínimo do conjunto de tuplas**
- ▶ **= ANY: valores iguais a algum valor do conjunto de tuplas**
  - É equivalente ao operador IN



# Operador ALL

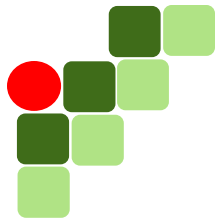
- ▶ Usado para comparar um valor  $v$  com um conjunto de valores retornados por uma subconsulta, usando operadores:
  - $=$ ,  $<>$
  - $<$ ,  $<=$
  - $>$ ,  $>=$
- ▶ Retorna TRUE se todos os valores retornados pela subconsulta satisfazem a comparação



# Operador ALL

- Recuperar os sobrenomes dos empregados que ganham salários menores que todos os salários dos empregados com função IT\_PROG

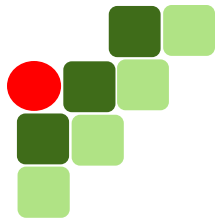
```
SELECT last_name, salary
FROM employees
WHERE job_id <> 'IT_PROG'
AND salary < ALL
      (SELECT salary
       FROM employees
       WHERE job_id = 'IT_PROG');
```



# Operador ALL

- Recuperar os sobrenomes dos empregados que ganham salários maiores que todos os salários dos empregados da cidade 'Southlake'

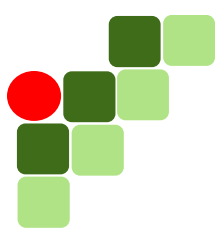
```
SELECT last_name, salary
FROM employees
WHERE salary > ALL
  (SELECT salary
   FROM employees E, departments D,
    locations L
   WHERE E.department_id = D.department_id
   AND D.location_id = L.location_id
   AND L.city = 'Southlake');
```



# Operador ALL

- ▶ **< ALL:** valores menores que o valor mínimo do conjunto de tuplas
- ▶ **> ALL:** valores maiores que o valor máximo do conjunto de tuplas
- ▶ **<> ALL:** valores diferentes de todos valores do conjunto de tuplas
  - É equivalente ao operador NOT IN



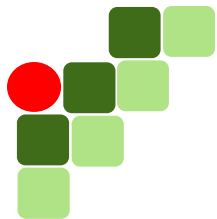


# Valores nulos em subconsultas

- ▶ Recuperar sobrenome dos empregados que não são gerentes de departamento

```
SELECT E.LAST_NAME  
FROM EMPLOYEES E  
WHERE E.EMPLOYEE_ID NOT IN  
      (SELECT D.MANAGER_ID  
       FROM DEPARTMENTS D);
```

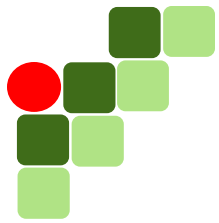
- ▶ Problema: a subconsulta retorna valores NULL
  - NOT IN é equivalente a <> ALL
  - Comparação com NULL retorna NULL



# Subconsultas correlacionadas

- Recuperar nome de empregados contratados antes do empregado com sobrenome Matos, que pertençam ao mesmo departamento que ele

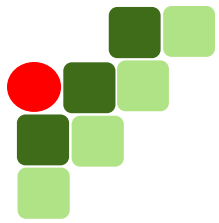
```
SELECT E.FIRST_NAME, E.LAST_NAME, E.HIRE_DATE,
       E.DEPARTMENT_ID
FROM EMPLOYEES E
WHERE E.HIRE_DATE <
      (SELECT M.HIRE_DATE
       FROM EMPLOYEES M
       WHERE M.LAST_NAME = 'Matos'
       AND M.DEPARTMENT_ID = E.DEPARTMENT_ID
      ) ;
```



# Subconsultas correlacionadas

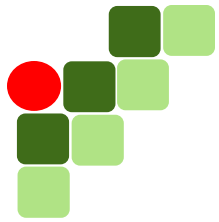
- Recuperar nome e salário dos empregados que ganham mais que seus gerentes

```
SELECT E.LAST_NAME, E.SALARY
FROM EMPLOYEES E
WHERE E.SALARY >
      (SELECT GERENTE.SALARY
       FROM EMPLOYEES GERENTE
       WHERE E.MANAGER_ID = GERENTE.EMPLOYEE_ID) ;
```



# Operador EXISTS

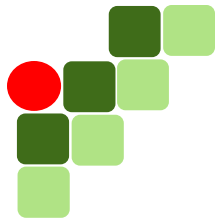
- ▶ Usado para verificar se uma subconsulta retorna algum resultado
  - Retorna TRUE se o resultado da subconsulta retorna pelo menos uma tupla
  - Retorna FALSE se o resultado da subconsulta não retorna tuplas (resultado vazio)
- ▶ Utiliza a técnica de subconsulta correlacionada



# Operador EXISTS

- Recuperar os nomes de departamento que possuem empregados

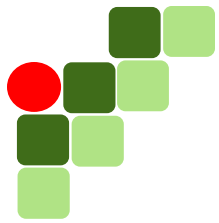
```
SELECT department_name  
FROM departments D  
WHERE EXISTS  
    (SELECT *  
     FROM employees E  
     WHERE D.department_id = E.department_id)  
ORDER BY department_name;
```



# Operador EXISTS

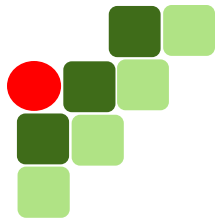
- Recuperar as cidades que tenham departamentos da empresa

```
SELECT city
FROM locations L
WHERE EXISTS
    (SELECT *
     FROM departments D
     WHERE L.location_id = D.location_id)
ORDER BY city;
```



# Operador NOT EXISTS

- ▶ Usado para verificar se o resultado de uma subconsulta é vazio (não retorna tuplas)
  - Retorna FALSE se o resultado da subconsulta retorna pelo menos uma tupla
  - Retorna TRUE se o resultado da subconsulta não retorna tuplas (resultado vazio)
- ▶ Também utiliza a técnica de subconsulta correlacionada

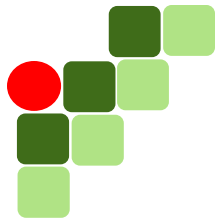


# Operador NOT EXISTS

- Recuperar os nomes de departamento que não possuem empregados cadastrados

```
SELECT department_name
FROM departments D
WHERE NOT EXISTS
      (SELECT *
        FROM employees E
         WHERE D.department_id = E.department_id)
ORDER BY department_name;
```

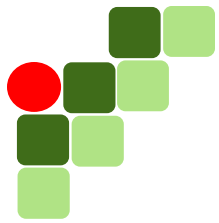




# Operador NOT EXISTS

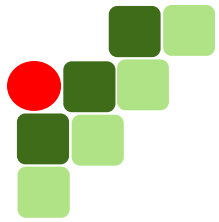
- Recuperar as regiões que não possuem departamentos

```
SELECT region_name
FROM regions R
WHERE NOT EXISTS
    (SELECT *
     FROM countries C, locations L,
          departments D
     WHERE R.region_id = C.region_id
     AND C.country_id = L.country_id
     AND L.location_id = D.location_id)
ORDER BY region_name;
```



# Dúvidas?





# Referências

- ▶ ELMASRI, Ramez E.; NAVATHE, Shamkant B. **Sistemas de Banco de Dados**. 6.ed. Pearson, 2011.
- ▶ Oracle Database SQL Language Reference
  - [SQL Queries and Subqueries](#)