Práctica Bases de Datos Avanzadas
Optimización de Consultas

Breidy Núñez | 2017-5633
Felipe Ramírez | 2018-6104
Luis Franco | 2018-6342

Prof. Leandro Fondeur

30/5/2019

Luego de realizar un estudio del **capítulo 14** del libro de texto que cubre el Tema 3 - Optimización de Consultas, responda las siguientes preguntas:

1. ¿Qué es la optimización de consultas?

*Es el **proceso** de selección del plan de evaluación de las consultas más eficiente de entre las muchas estrategias generalmente disponibles para el procesamiento de una consulta dada, especialmente si la consulta es compleja.*

2. ¿Qué realizan los optimizadores para realizar sus estimaciones?

*Hacen uso de la **información estadística** sobre las relaciones, como los tamaños de las relaciones y las profundidades de los índices.*

3. La generación de planes de evaluación de consultas implica dos etapas. Descríbelas

Las dos etapas son:

   a. *La generación de expresiones con equivalencia lógica a la expresión dada.*
   b. *La anotación de las expresiones resultantes en formas diferentes destinadas a la evaluación de consultas alternas.*

4. ¿Qué son las reglas de equivalencia?

*Son **reglas de equivalencia** el conjunto de normas que son utilizadas para establecer una metodología de transición para crear consultas equivalentes lógicamente a una consulta madre.*

5. ¿Cuándo dos expresiones del álgebra relacional son equivalentes?

*Si en cada ejemplar legal de base de datos, **las dos expresiones generan el mismo conjunto de tupla**s.*

6. ¿Qué es un ejemplar legal de la base de datos?

*Un **ejemplar legal** satisface todas las restricciones de integridad especificadas en el esquema de la base de datos.*

Realice los siguientes ejercicios luego de estudiar el documento "**Les05-Group Functions.ppt**":

1. Busque el salario más alto, el más bajo, la suma y el salario medio de todos los empleados. Etiquete las columnas como Maximum, Minimum, Sum y Average, respectivamente. Redondee los resultados al siguiente número entero.

```
select round(max(salary),0) as "Maximum", round(min(salary),0)as "Minimum",
round(sum(salary),0) as"Sum",round(avg(salary),0) as "Average"
from employees;
```

Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 1 en 0.387 segundos

	Maximum	Minimum	Sum	Average
1	24000	2100	691416	6462

2. Modifique la consulta anterior para mostrar el salario mínimo, el máximo, la suma y el salario medio para cada tipo de trabajo.

Hoja de Trabajo | Generador de Consultas

```
SELECT  job_ID, ROUND(MAX(salary),0) AS "Maximum", ROUND(MIN(salary),0) AS "Minimum", ROUND(SUM(salary),0)
AS "Sum", ROUND(AVG(salary),0) AS "Average" from employees GROUP BY job_id;
```

Resultado de la Consulta

Todas las Filas Recuperadas: 19 en 0.003 segundos

	JOB_ID	Maximum	Minimum	Sum	Average
1	IT_PROG	9000	4200	28800	5760
2	AC_MGR	12008	12008	12008	12008
3	AC_ACCOUNT	8300	8300	8300	8300
4	ST_MAN	8200	5800	36400	7280
5	PU_MAN	11000	11000	11000	11000
6	AD_ASST	4400	4400	4400	4400
7	AD_VP	17000	17000	34000	17000
8	SH_CLERK	4200	2500	64300	3215
9	FI_ACCOUNT	9000	6900	39600	7920
10	FI_MGR	12008	12008	12008	12008

3. Escriba una consulta para mostrar el número de personas con el mismo puesto.

```
SELECT job_id AS "Puesto", count (employee_id ) AS "Cantidad" FROM employees group by job_id;
```

Salida de Script | Resultado de la Consulta | Resultado de la Consulta 1 | Resultado de la Consulta 2

Todas las Filas Recuperadas: 19 en 0.003 segundos

	Puesto	Cantidad
1	AC_ACCOUNT	1
2	AC_MGR	1
3	AD_ASST	1
4	AD_PRES	1
5	AD_VP	2
6	FI_ACCOUNT	5
7	FI_MGR	1
8	HR_REP	1
9	IT_PROG	5
10	MK_MAN	1
11	MK_REP	1
12	PR_REP	1
13	PU_CLERK	5
14	PU_MAN	1
15	SA_MAN	5
16	SA_REP	30
17	SH_CLERK	20
18	ST_CLERK	20
19	ST_MAN	5

4. Determine el número de supervisores sin listarlos. Etiquete la columna como Number of Managers. Indicación: Utilice la columna MANAGER_ID para determinar el número de supervisores.

```
select  count(DISTINCT manager_id) as "Number of Managers"
from employees
```

Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 1 en 0.004 segundos

	Number of Managers
1	18

5. Cree un informe para mostrar el número de supervisor y el salario del empleado peor pagado de ese supervisor. Excluya a los empleados cuyo supervisor no se conozca. Excluya los grupos en los que el salario mínimo sea 6.000 dólares o menos. Clasifique la salida en orden descendente de salario.

Hoja de Trabajo | Generador de Consultas

```
SELECT DISTINCT manager_ID, MIN(salary) FROM employees GROUP BY manager_ID
HAVING manager_id IS NOT NULL AND MIN(salary) > 6000 ORDER BY MIN(salary) DESC;
```

Salida de Script | Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 8 en 0.122 segundos

	MANAGER_ID	MIN(SALARY)
1	102	9000
2	205	8300
3	145	7000
4	146	7000
5	108	6900
6	147	6200
7	149	6200
8	148	6100

Realice los siguientes ejercicios luego de estudiar el documento "**Les06-Join.ppt**":

1. Escriba una consulta para que el departamento de recursos humanos genere las direcciones de todos los departamentos. Utilice las tablas LOCATIONS y COUNTRIES. Muestre en la salida el identificador de ubicación, el domicilio de calle, la ciudad, el estado o la provincia y el país. Utilice una **unión natural** para generar los resultados.

```
select locations.location_id, locations.street_address , locations.city, countries.country_name FROM locations NATURAL JOIN countries;
```

Salida de Script × | Resultado de la Consulta 2 ×

SQL | Todas las Filas Recuperadas: 23 en 0.003 segundos

	LOCATION_ID	STREET_ADDRESS	CITY	COUNTRY_NAME
1	1000	1297 Via Cola di Rie	Roma	Italy
2	1100	93091 Calle della Testa	Venice	Italy
3	1200	2017 Shinjuku-ku	Tokyo	Japan
4	1300	9450 Kamiya-cho	Hiroshima	Japan
5	1400	2014 Jabberwocky Rd	Southlake	United States of America
6	1500	2011 Interiors Blvd	South San Francisco	United States of America
7	1600	2007 Zagora St	South Brunswick	United States of America
8	1700	2004 Charade Rd	Seattle	United States of America
9	1800	147 Spadina Ave	Toronto	Canada
10	1900	6092 Boxwood St	Whitehorse	Canada
11	2000	40-5-12 Laogianggen	Beijing	China
12	2100	1298 Vileparle (E)	Bombay	India
13	2200	12-98 Victoria Street	Sydney	Australia
14	2300	198 Clementi North	Singapore	Singapore
15	2400	8204 Arthur St	London	United Kingdom
16	2500	Magdalen Centre, The Oxford Science Park	Oxford	United Kingdom
17	2600	9702 Chester Road	Stretford	United Kingdom
18	2700	Schwanthalerstr. 7031	Munich	Germany
19	2800	Rua Frei Caneca 1360	Sao Paulo	Brazil
20	2900	20 Rue des Corps-Saints	Geneva	Switzerland
21	3000	Murtenstrasse 921	Bern	Switzerland
22	3100	Pieter Breughelstraat 837	Utrecht	Netherlands
23	3200	Mariano Escobedo 9991	Mexico City	Mexico

2. El departamento de recursos humanos necesita un informe de todos los empleados. Escriba una consulta para mostrar el apellido, el número de departamento y el nombre de departamento de todos los empleados. Use la cláusula **USING**.

```
select last_name, department_id, department_name
from employees join departments
using (department_id)
```

Resultado de la Consulta ×

SQL | Todas las Filas Recuperadas: 106 en 0.008 segundos

	LAST_NAME	DEPARTMENT_ID	DEPARTMENT_NAME
1	Whalen	10	Administration
2	Fay	20	Marketing
3	Hartstein	20	Marketing
4	Tobias	30	Purchasing
5	Colmenares	30	Purchasing
6	Baida	30	Purchasing
7	Raphaely	30	Purchasing
8	Khoo	30	Purchasing
9	Himuro	30	Purchasing
10	Mavris	40	Human Resources
11	Feeney	50	Shipping
12	Fleaur	50	Shipping
13	Fripp	50	Shipping
14	Gates	50	Shipping

3. El departamento de recursos humanos necesita un informe de los empleados de Toronto. Muestre el apellido, el puesto, el número de departamento y el nombre de departamento de todos los empleados que trabajan en Toronto. Use la cláusula **JOIN+ON**.

Hoja de Trabajo | Generador de Consultas

```
SELECT e.last_name, e.job_id, e.department_id, d.department_name
FROM employees e
JOIN departments d ON e.department_id=d.department_id
JOIN locations l ON d.location_id=l.location_id
WHERE l.city = 'Toronto';
```

Salida de Script | Rastreo Automático | Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 2 en 0.002 segundos

	LAST_NAME	JOB_ID	DEPARTMENT_ID	DEPARTMENT_NAME
1	Hartstein	MK_MAN	20	Marketing
2	Fay	MK_REP	20	Marketing

4. Cree un informe que muestre el apellido del empleado y el número de empleado junto al apellido y número de supervisor del supervisor del empleado. Etiquete las columnas como Employee, Emp#, Manager y Mgr#, respectivamente. Use un **SELF-JOIN**.

```
SELECT e.last_name as "Employee", e.employee_id as "Emp#", m.last_name as "Manager", m.employee_id as "Mgr#"
FROM employees e
INNER JOIN employees m on m.employee_id = e.manager_id;
```

Salida de Script | Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 106 en 0.012 segundos

	Employee	Emp#	Manager	Mgr#
1	Kumar	173	Cambrault	148
2	Bates	172	Cambrault	148
3	Smith	171	Cambrault	148
4	Fox	170	Cambrault	148
5	Bloom	169	Cambrault	148
6	Ozer	168	Cambrault	148
7	Hunold	103	De Haan	102
8	Banda	167	Errazuriz	147
9	Ande	166	Errazuriz	147
10	Lee	165	Errazuriz	147
11	Marvins	164	Errazuriz	147
12	Greene	163	Errazuriz	147
13	Vishney	162	Errazuriz	147

5. Modifique el ejercicio anterior para que muestre todos los empleados, incluido King, que no tiene supervisor. Ordene los resultados por el número de empleado. Use un **OUTER-JOIN**.

```
select e.last_name as "Employee" , e.employee_id AS "Emp#" , m.last_name AS "Manager" , m.employee_ID AS "Mgr#"
from employees e left outer join employees m on m.employee_id= e.manager_id
order by e.employee_id;
```

sultado de la Consulta

SQL | Todas las Filas Recuperadas: 107 en 0.011 segundos

	Employee	Emp#	Manager	Mgr#
1	King	100	(null)	(null)
2	Kochhar	101	King	100
3	De Haan	102	King	100
4	Hunold	103	De Haan	102
5	Ernst	104	Hunold	103
6	Austin	105	Hunold	103
7	Pataballa	106	Hunold	103
8	Lorentz	107	Hunold	103
9	Greenberg	108	Kochhar	101

6. El departamento de recursos humanos necesita un informe sobre grados de trabajo y salarios. Para familiarizarse con la tabla JOB_GRADES, muestre en primer lugar su estructura. A continuación, cree una consulta que muestre el apellido, el puesto, el nombre de departamento, el salario y el grado de todos los empleados. Use un **NON-EQUIJOIN.**

Hoja de Trabajo | Generador de Consultas

```
DESCRIBE job_grades;
```

Salida de Script | Rastreo Automático | Resultado de la Consulta

Tarea terminada en 0.028 segundos

```
Nombre       ¿Nulo?   Tipo
----------- -------- -----------
GRADE_LEVEL           CHAR(1)
LOWEST_SAL   NOT NULL NUMBER(8,2)
HIGHEST_SAL  NOT NULL NUMBER(8,2)
```

Hoja de Trabajo | Generador de Consultas

```
SELECT e.last_name, e.job_id, d.department_name, e.salary, j.grade_level
FROM employees e
JOIN job_grades j ON e.salary BETWEEN j.lowest_sal AND j.highest_sal
LEFT OUTER JOIN departments d ON e.department_id = d.department_id;
```

Salida de Script | Rastreo Automático | Resultado de la Consulta

SQL | Se han recuperado 50 filas en 0.005 segundos

	LAST_NAME	JOB_ID	DEPARTMENT_NAME	SALARY	GRADE_LEVEL
1	Olson	ST_CLERK	Shipping	2100	A
2	Markle	ST_CLERK	Shipping	2200	A
3	Philtanker	ST_CLERK	Shipping	2200	A
4	Landry	ST_CLERK	Shipping	2400	A
5	Gee	ST_CLERK	Shipping	2400	A
6	Colmenares	PU_CLERK	Purchasing	2500	A
7	Marlow	ST_CLERK	Shipping	2500	A
8	Patel	ST_CLERK	Shipping	2500	A
9	Vargas	ST_CLERK	Shipping	2500	A
10	Sullivan	SH_CLERK	Shipping	2500	A

7. El departamento de recursos humanos quiere determinar los nombres de todos los empleados contratados después de Davies. Cree una consulta para mostrar el nombre y la fecha de contratación de cualquier empleado contratado después que Davies. Use un **SELF-JOIN**.

```
Select e.first_name as "Nombre", e.hire_date FROM employees e
INNER JOIN employees d on e.hire_date > d.hire_date WHERE d.last_name = 'Davies';
```

Salida de Script | Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 81 en 0.007 segundos

	Nombre	HIRE_DATE
1	Neena	21-SEP-05
2	Alexander	03-ENE-06
3	Bruce	21-MAY-07
4	David	25-JUN-05
5	Valli	05-FEB-06
6	Diana	07-FEB-07
7	John	28-SEP-05
8	Ismael	30-SEP-05
9	Jose Manuel	07-MAR-06
10	Luis	07-DIC-07
11	Shelli	24-DIC-05

Realice los siguientes ejercicios luego de estudiar el documento "**Les07-Subqueries.ppt**":

1. El departamento de recursos humanos necesita una consulta que pida al usuario el apellido de un empleado. La consulta muestra entonces el apellido y la fecha de contratación de cualquier empleado del mismo departamento que aquel cuyo apellido se suministre (excepto ese empleado). Por ejemplo, si el usuario introduce Zlotkey, busca todos los empleados que trabajen con Zlotkey (excepto Zlotkey).

Hoja de Trabajo | Generador de Consultas

```
DEFINE apellido = '&apellido';
SELECT last_name, hire_date, department_id FROM employees WHERE department_id = ANY
                                            (SELECT department_id FROM employees WHERE last_name = INITCAP('&apellido'))
                                            AND employee_id <> ALL
                                            (SELECT employee_id FROM employees WHERE last_name = INITCAP('&apellido'));
UNDEFINE apellido;
```

Salida de Script | Resultado de la Consulta

Todas las Filas Recuperadas: 35 en 0.007 segundos

	LAST_NAME	HIRE_DATE	DEPARTMENT_ID
1	Vishney	11-NOV-05	80
2	Ande	24-MAR-08	80
3	Fox	24-ENE-06	80
4	Bernstein	24-MAR-05	80
5	Hutton	19-MAR-05	80
6	Cambrault	09-DIC-06	80
7	Kochhar	21-SEP-05	90
8	Partners	05-ENE-05	80
9	Olsen	30-MAR-06	80
10	Doran	15-DIC-05	80

2. Cree un informe que muestre el número de empleado y el apellido de todos los empleados que ganan más que el sueldo medio. Clasifique los resultados en orden ascendente de salario.

Hoja de Trabajo | Generador de Consultas

```
SELECT employee_id, last_name FROM employees WHERE salary >
                                            (SELECT AVG(salary) FROM employees)
ORDER BY salary ASC;
```

Salida de Script | Resultado de la Consulta | Resultado de la Consulta 1

Se han recuperado 50 filas en 0.005 segundos

	EMPLOYEE_ID	LAST_NAME
1	203	Mavris
2	123	Vollman
3	165	Lee
4	113	Popp
5	155	Tuvault
6	161	Sewall
7	178	Grant
8	164	Marvins
9	172	Bates
10	171	Smith

3. Escriba una consulta que muestre el número de empleado y el apellido de todos los empleados que trabajen en un departamento en que haya algún empleado cuyo apellido contenga una "**a**".

```
select employee_id, last_name AS "Apellido" FROM employees
where department_id = ANY
(select department_id FROM employees where last_name LIKE ('%a%'));
```

Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 104 en 0.033 segundos

	EMPLOYEE_ID	Apellido
1	204	Baer
2	119	Colmenares
3	118	Himuro
4	117	Tobias
5	116	Baida
6	115	Khoo
7	114	Raphaely
8	179	Johnson
9	177	Livingston
10	176	Taylor
11	175	Hutton
12	174	Abel
13	173	Kumar
14	172	Bates
15	171	Smith

4. El departamento de recursos humanos necesita un informe que muestre el apellido, el número de departamento y el identificador de puesto de todos los empleados cuyos identificadores de ubicación de departamento sean 1700.

```
select last_name, department_id,job_id  from employees
where department_id = any (select department_id from departments
                           where location_id = 1700)
```

Resultado de la Consulta | Resultado de la Consulta 1 | Resultado de la Consulta 2 | Resultado de la Consulta 3

SQL | Todas las Filas Recuperadas: 18 en 0.003 segundos

	LAST_NAME	DEPARTMENT_ID	JOB_ID
1	King	90	AD_PRES
2	Kochhar	90	AD_VP
3	De Haan	90	AD_VP
4	Greenberg	100	FI_MGR
5	Faviet	100	FI_ACCOUNT
6	Chen	100	FI_ACCOUNT
7	Sciarra	100	FI_ACCOUNT
8	Urman	100	FI_ACCOUNT
9	Popp	100	FI_ACCOUNT
10	Raphaely	30	PU_MAN
11	Khoo	30	PU_CLERK
12	Baida	30	PU_CLERK
13	Tobias	30	PU_CLERK
14	Himuro	30	PU_CLERK
15	Colmenares	30	PU_CLERK
16	Whalen	10	AD_ASST
17	Higgins	110	AC_MGR
18	Gietz	110	AC_ACCOUNT

5. Cree un informe para recursos humanos que muestre el apellido y el salario de todos los empleados que informen a King.

Hoja de Trabajo | Generador de Consultas

```
SELECT last_name, salary FROM employees WHERE manager_id = ANY
                                (SELECT employee_id FROM employees WHERE last_name = 'King');
```

Salida de Script | Resultado de la Consulta

Todas las Filas Recuperadas: 14 en 0.004 segundos

	LAST_NAME	SALARY	MANAGER_ID
1	Kochhar	17000	100
2	De Haan	17000	100
3	Raphaely	11000	100
4	Weiss	8000	100
5	Fripp	8200	100
6	Kaufling	7900	100
7	Vollman	6500	100
8	Mourgos	5800	100
9	Russell	14000	100
10	Partners	13500	100
11	Errazuriz	12000	100
12	Cambrault	11000	100
13	Zlotkey	10500	100
14	Hartstein	13000	100