讲师: collen7788@126.com

Presentation

多表查询

本章目标

1. 使用等值和不等值连接。
2. 外连接查询
3. 使用自连接。

从多个表中获取数据

EMPLOYEES

EMPLOYEE_ID	LAST_NAME	DEPARTMENT_ID
100	King	90
101	Kochhar	90
...		
202	Fay	20
205	Higgins	110
206	Gietz	110

DEPARTMENTS

DEPARTMENT_ID	DEPARTMENT_NAME	LOCATION_ID
10	Administration	1700
20	Marketing	1800
50	Shipping	1500
60	IT	1400
80	Sales	2500
90	Executive	1700
110	Accounting	1700
190	Contracting	1700

EMPLOYEE_ID	DEPARTMENT_ID	DEPARTMENT_NAME
200	10	Administration
201	20	Marketing
202	20	Marketing
...		
102	90	Executive
205	110	Accounting
206	110	Accounting

笛卡尔集

EMPLOYEES (20行)

EMPLOYEE_ID	LAST_NAME	DEPARTMENT_ID
100	King	90
101	Kochhar	90
...		
202	Fay	20
205	Higgins	110
206	Gietz	110

20 rows selected.

DEPARTMENTS (8行)

DEPARTMENT_ID	DEPARTMENT_NAME	LOCATION_ID
10	Administration	1700
20	Marketing	1800
50	Shipping	1500
60	IT	1400
80	Sales	2500
90	Executive	1700
110	Accounting	1700
190	Contracting	1700

8 rows selected.

笛卡尔集:
20x8=160行

EMPLOYEE_ID	DEPARTMENT_ID	LOCATION_ID
100	90	1700
101	90	1700
102	90	1700
103	60	1700
104	60	1700
107	60	1700

笛卡尔集

- **笛卡尔集会在下面条件下产生:**
  - 省略连接条件
  - 连接条件无效
  - 所有表中的所有行互相连接
- **为了避免笛卡尔集， 可以在 WHERE 加入有效的连接条件。**
- **在实际运行环境下，应避免使用全笛卡尔全集。**

连接的类型

- **Oracle的连接**
  - 等值连接
  - 不等值连接
  - 外连接
  - 自连接
- **SQL99的连接**
  - Cross joins
  - Natural joins
  - Using clause
  - Full or two sided outer joins

Oracle 连接

- 使用连接在多个表中查询数据

```
SELECT    table1.column, table2.column
FROM      table1, table2
WHERE     table1.column1 = table2.column2;
```

- 在 **WHERE** 字句中写入连接条件。
- 在表中有相同列时，在列名之前加上表名前缀

等值连接

EMPLOYEES

EMPLOYEE_ID	DEPARTMENT_ID
200	10
201	20
202	20
124	50
141	50
142	50
143	50
144	50
103	60
104	60
107	60
149	80
174	80
176	80
...	

外键

DEPARTMENTS

DEPARTMENT_ID	DEPARTMENT_NAME
10	Administration
20	Marketing
20	Marketing
50	Shipping
50	Shipping
50	Shipping
50	Shipping
50	Shipping
60	IT
60	IT
60	IT
80	Sales
80	Sales
80	Sales
...	

主键

等值连接

- 查询员工信息：员工号、姓名、月薪和部门名称

```
select e.empno,e.ename,e.sal,d.dname
from emp e,dept d
where e.deptno=d.deptno;
```

EMPNO	ENAME	SAL	DNAME
7369	SMITH	800	RESEARCH
7499	ALLEN	1600	SALES
7521	WARD	1250	SALES
7566	JONES	2975	RESEARCH
7654	MARTIN	1250	SALES
7698	BLAKE	2850	SALES
7782	CLARK	2450	ACCOUNTING
7788	SCOTT	3000	RESEARCH
7839	KING	5000	ACCOUNTING
7844	TURNER	1500	SALES
7876	ADAMS	1100	RESEARCH
7900	JAMES	950	SALES
7902	FORD	3000	RESEARCH
7934	MILLER	1300	ACCOUNTING

多个连接条件与 AND 操作符

EMPLOYEES

LAST_NAME	DEPARTMENT_ID
Whalen	10
Hartstein	20
Fay	20
Mourgos	50
Rajs	50
Davies	50
Matos	50
Vargas	50
Hunold	60
Ernst	60

...

DEPARTMENTS

DEPARTMENT_ID	DEPARTMENT_NAME
10	Administration
20	Marketing
20	Marketing
50	Shipping
50	Shipping
50	Shipping
50	Shipping
50	Shipping
60	IT
60	IT

...

区分重复的列名

- 使用表名前缀在多个表中区分相同的列。
- 在不同表中具有相同列名的列可以用表的别名加以区分。

表的别名

- 使用别名可以简化查询。
- 使用表名前缀可以提高执行效率。

```
SELECT e.employee_id, e.last_name, e.department_id,
       d.department_id, d.location_id
FROM   employees e , departments d
WHERE  e.department_id = d.department_id;
```

- 如果使用了表的别名，则不能再使用表的真名。

连接多个表

EMPLOYEES

LAST_NAME	DEPARTMENT_ID
King	90
Kochhar	90
De Haan	90
Hunold	60
Ernst	60
Lorentz	60
Mourgos	50
Rajs	50
Davies	50
Matos	50
Vargas	50
Zlotkey	80
Abel	80
Taylor	80

...

20 rows selected.

DEPARTMENTS

DEPARTMENT_ID	LOCATION_ID
10	1700
20	1800
50	1500
60	1400
80	2500
90	1700
110	1700
190	1700

8 rows selected.

LOCATIONS

LOCATION_ID	CITY
1400	Southlake
1500	South San Francisco
1700	Seattle
1800	Toronto
2500	Oxford

- 连接 *n*个表,至少需要 n-1个连接条件。 例如：连接三个表，至少需要两个连接条件。

不等值连接

EMP

ENAME	SAL
SMITH	800
ALLEN	1600
WARD	1250
JONES	2975
MARTIN	1250
BLAKE	2850
CLARK	2450
SCOTT	3000
KING	5000
TURNER	1500
ADAMS	1100
JAMES	950
FORD	3000
MILLER	1300

SALGRADE

GRADE	LOSAL	HISAL
1	700	1200
2	1201	1400
3	1401	2000
4	2001	3000
5	3001	9999

EMP表中的列工资
应在SALGRADES表中的最高
工资与最低工资之间

不等值连接

- 查询员工信息：员工号、姓名、月薪和工资级别

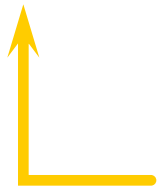
```
select e.empno,e.ename,e.sal,s.grade
from emp e,salgrade s
where e.sal between s.losal and s.hisal;
```

EMPNO	ENAME	SAL	GRADE
7369	SMITH	800	1
7900	JAMES	950	1
7876	ADAMS	1100	1
7521	WARD	1250	2
7654	MARTIN	1250	2
7934	MILLER	1300	2
7844	TURNER	1500	3
7499	ALLEN	1600	3
7782	CLARK	2450	4
7698	BLAKE	2850	4
7566	JONES	2975	4
7788	SCOTT	3000	4
7902	FORD	3000	4
7839	KING	5000	5

外连接

DEPT

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON



40号部门没有员工

EMP

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17-12月-80	800		20
7499	ALLEN	SALESMAN	7698	20-2月 -81	1600	300	30
7521	WARD	SALESMAN	7698	22-2月 -81	1250	500	30
7566	JONES	MANAGER	7839	02-4月 -81	2975		20
7654	MARTIN	SALESMAN	7698	28-9月 -81	1250	1400	30
7698	BLAKE	MANAGER	7839	01-5月 -81	2850		30
7782	CLARK	MANAGER	7839	09-6月 -81	2450		10
7788	SCOTT	ANALYST	7566	13-7月 -87	3000		20
7839	KING	PRESIDENT		17-11月-81	5000		10
7844	TURNER	SALESMAN	7698	08-9月 -81	1500	0	30
7876	ADAMS	CLERK	7788	13-7月 -87	1100		20
7900	JAMES	CLERK	7698	03-12月-81	950		30
7902	FORD	ANALYST	7566	03-12月-81	3000		20
7934	MILLER	CLERK	7782	23-1月 -82	1300		10

外连接语法

- 使用外连接可以查询不满足连接条件的数据。
- 外连接的符号是（+）

```
SELECT table1.column, table2.column
FROM   table1, table2
WHERE  table1.column(+) = table2.column;
```

```
SELECT table1.column, table2.column
FROM   table1, table2
WHERE  table1.column = table2.column(+);
```

外连接

- 按部门统计员工人数：部门号，部门名称和人数

```
select d.deptno,d.dname,count(e.empno)
from emp e,dept d
where e.deptno(+)=d.deptno
group by d.deptno,d.dname
```

DEPTNO	DNAME	COUNT(E.EMPNO)
10	ACCOUNTING	3
40	OPERATIONS	0
20	RESEARCH	5
30	SALES	6

自连接

EMPLOYEES (WORKER)

EMPLOYEE_ID	LAST_NAME	MANAGER_ID
100	King	
101	Kochhar	100
102	De Haan	100
103	Hunold	102
104	Ernst	103
107	Lorentz	103
124	Mourgos	100

...

EMPLOYEES (MANAGER)

EMPLOYEE_ID	LAST_NAME
100	King
101	Kochhar
102	De Haan
103	Hunold
104	Ernst
107	Lorentz
124	Mourgos

...

WORKER 表中的MANAGER_ID 和 MANAGER 表中的 MANAGER_ID相等

自连接

- 查询员工信息：员工和员工的老板姓名

```
select e.ename 员工,b.ename 员工的老板
from emp e,emp b
where e.mgr = b.empno;
```

员工	员工的老板
FORD	JONES
SCOTT	JONES
JAMES	BLAKE
TURNER	BLAKE
MARTIN	BLAKE
WARD	BLAKE
ALLEN	BLAKE
MILLER	CLARK
ADAMS	SCOTT
CLARK	KING
BLAKE	KING
JONES	KING
SMITH	FORD

使用SQL: 1999 语法连接

- 使用连接从多个表中查询数据：

```
SELECT    table1.column, table2.column
FROM      table1
[CROSS JOIN table2] |
[NATURAL JOIN table2] |
[JOIN table2 USING (column_name)] |
[JOIN table2
  ON(table1.column_name = table2.column_name)] |
[LEFT|RIGHT|FULL OUTER JOIN table2
  ON (table1.column_name = table2.column_name)];
```

叉集

- 使用**CROSS JOIN** 子句使连接的表产生叉集。
- 叉集和笛卡尔集是相同的。

```
SELECT last_name, department_name
FROM   employees
CROSS JOIN departments ;
```

LAST_NAME	DEPARTMENT_NAME
King	Administration
Kochhar	Administration
De Haan	Administration
Hunold	Administration

...

160 rows selected.

自然连接

- **NATURAL JOIN** 子句，会以两个表中具有相同名字的列为条件创建等值连接。
- 在表中查询满足等值条件的数据。
- 如果只是列名相同而数据类型不同，则会产生错误。

自然连接

```
SELECT department_id, department_name,
       location_id, city
FROM   departments
NATURAL JOIN locations ;
```

DEPARTMENT_ID	DEPARTMENT_NAME	LOCATION_ID	CITY
60	IT	1400	Southlake
50	Shipping	1500	South San Francisco
10	Administration	1700	Seattle
90	Executive	1700	Seattle
110	Accounting	1700	Seattle
190	Contracting	1700	Seattle
20	Marketing	1800	Toronto
80	Sales	2500	Oxford

8 rows selected.

使用 USING 子句创建连接

- 在**NATURAL JOIN** 子句创建等值连接时，可以使用 USING 子句指定等值连接中需要用到的列。
- 使用 **USING** 可以在有多个列满足条件时进行选择。
- 不要给选中的列中加上表名前缀或别名。
- NATURAL JOIN 和 USING 子句经常同时使用。

USING 子句

```
SELECT e.employee_id, e.last_name, d.location_id
FROM   employees e JOIN departments d
USING (department_id) ;
```

EMPLOYEE_ID	LAST_NAME	LOCATION_ID
200	Whalen	1700
201	Hartstein	1800
202	Fay	1800
124	Mourgos	1500
141	Rajs	1500
142	Davies	1500
143	Matos	1500
144	Vargas	1500
103	Hunold	1400

...

19 rows selected.

使用ON 子句创建连接

- 自然连接中是以具有相同名字的列为连接条件的。
- 可以使用 ON 子句指定额外的连接条件。
- 这个连接条件是与其它条件分开的。
- ON 子句使语句具有更高的易读性。

ON 子句

```
SELECT e.employee_id, e.last_name, e.department_id,
       d.department_id, d.location_id
FROM   employees e JOIN departments d
ON     (e.department_id = d.department_id);
```

EMPLOYEE_ID	LAST_NAME	DEPARTMENT_ID	DEPARTMENT_ID	LOCATION_ID
200	Whalen	10	10	1700
201	Hartstein	20	20	1800
202	Fay	20	20	1800
124	Mourgos	50	50	1500
141	Rajs	50	50	1500
142	Davies	50	50	1500
143	Matos	50	50	1500

...

19 rows selected.

使用 ON 子句创建多表连接

```
SELECT employee_id, city, department_name
FROM   employees e
JOIN   departments d
ON     d.department_id = e.department_id
JOIN   locations l
ON     d.location_id = l.location_id;
```

EMPLOYEE_ID	CITY	DEPARTMENT_NAME
103	Southlake	IT
104	Southlake	IT
107	Southlake	IT
124	South San Francisco	Shipping
141	South San Francisco	Shipping
142	South San Francisco	Shipping
143	South San Francisco	Shipping
144	South San Francisco	Shipping

...

内连接和外连接(2)

- 在**SQL: 1999**中，内连接只返回满足连接条件的数据
- 两个表在连接过程中除了返回满足连接条件的行以外还返回左（或右）表中不满足条件的行，这种连接称为左（或右） 外联接。
- 两个表在连接过程中除了返回满足连接条件的行以外还返回两个表中不满足条件的行，这种连接称为满 外联接。

左外联接

```
SELECT e.last_name, e.department_id, d.department_name
FROM   employees e
LEFT OUTER JOIN departments d
ON    (e.department_id = d.department_id) ;
```

LAST_NAME	DEPARTMENT_ID	DEPARTMENT_NAME
Whalen	10	Administration
Fay	20	Marketing
Hartstein	20	Marketing
...		
De Haan	90	Executive
Kochhar	90	Executive
King	90	Executive
Gietz	110	Accounting
Higgins	110	Accounting
Grant		

20 rows selected.

右外联接

```
SELECT e.last_name, e.department_id, d.department_name
FROM   employees e
RIGHT OUTER JOIN departments d
ON     (e.department_id = d.department_id) ;
```

LAST_NAME	DEPARTMENT_ID	DEPARTMENT_NAME
King	90	Executive
Kochhar	90	Executive
...		
Whalen	10	Administration
Hartstein	20	Marketing
Fay	20	Marketing
Higgins	110	Accounting
Gietz	110	Accounting
		Contracting

20 rows selected.

满外联接

```
SELECT e.last_name, e.department_id, d.department_name
FROM   employees e
FULL OUTER JOIN departments d
ON    (e.department_id = d.department_id) ;
```

LAST_NAME	DEPARTMENT_ID	DEPARTMENT_NAME
Whalen	10	Administration
Fay	20	Marketing
...		
De Haan	90	Executive
Kochhar	90	Executive
King	90	Executive
Gietz	110	Accounting
Higgins	110	Accounting
Grant		
		Contracting

21 rows selected.

讲师: collen7788@126.com

Presentation

Thank you