Lecture 4: Joins and other set operations

Wednesday, February 4, 2015

Agenda for today

- Questions about Quiz #1 or Homework #1?
- Continue Chapter 4 from Murach book

Types of joins

- Inner joins
- Outer joins (left, right or full)
- Self joins
- Cross joins

Inner join example

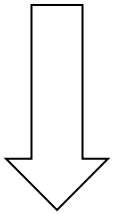
Vendors

id	name	city	state
1	Canon	Tokyo	NULL
3	Hitachi	Tokyo	NULL
4	IBM	Poughkeepsie	New York
10	USPS	DC	DC

Invoices

invoice_number	vendor_id	invoice_total
1234-56	1	25
78-9999	4	65
10-1234	10	15
99-9999	1000	0

```
SELECT name, invoice_number
FROM Vendors, Invoices
WHERE id = vendor_id
AND invoice_total < 50
```



name	invoice_total
?	?

Notice: vendors without invoices will be lost

Outer joins

- Left outer join:
  – Includes the left record(s) even when there is no match
- Right outer join:
  – Includes the right records(s) even when there is no match
- Full outer join:
  – Includes both left and right records even when there is no match

Left outer join example

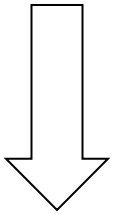
Vendors

id	name	city	state
1	Canon	Tokyo	NULL
3	Hitachi	Tokyo	NULL
4	IBM	Poughkeepsie	New York
10	USPS	DC	DC

Invoices

invoice_number	vendor_id	invoice_total
1234-56	1	25
78-9999	4	65
10-1234	10	15
99-9999	1000	0

```
SELECT name, invoice_number
FROM Vendors
LEFT JOIN Invoices
ON id = vendor_id
WHERE invoice_total < 50
```



name	invoice_total
?	?

Right outer join example

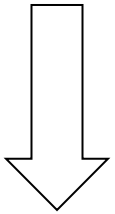
Vendors

id	name	city	state
1	Canon	Tokyo	NULL
3	Hitachi	Tokyo	NULL
4	IBM	Poughkeepsie	New York
10	USPS	DC	DC

Invoices

invoice_number	vendor_id	invoice_total
1234-56	1	25
78-9999	4	65
10-1234	10	15
99-9999	1000	0

```
SELECT name, invoice_number
FROM Vendors
RIGHT JOIN Invoices
ON id = vendor_id
WHERE invoice_total < 50
```



name	invoice_total
?	?

Full outer join example

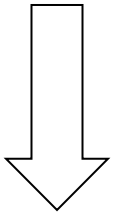
Vendors

id	name	city	state
1	Canon	Tokyo	NULL
3	Hitachi	Tokyo	NULL
4	IBM	Poughkeepsie	New York
10	USPS	DC	DC

Invoices

invoice_number	vendor_id	invoice_total
1234-56	1	25
78-9999	4	65
10-1234	10	15
99-9999	1000	0

```
SELECT name, invoice_number
FROM Vendors
FULL JOIN Invoices
ON id = vendor_id
WHERE invoice_total < 50
```



name	invoice_total
?	?

The Departments table

	DEPARTMENT_NUMBER	DEPARTMENT_NAME
1	1	Accounting
2	2	Payroll
3	3	Operations
4	4	Personnel
5	5	Maintenance

The Employees table

EMPLOYEE_ID	LAST_NAME	FIRST_NAME	DEPARTMENT_NUMBER
1	Smith	Cindy	2
2	Jones	Elmer	4
3	Simonian	Ralph	2
4	Hernandez	Olivia	1
5	Aaronsen	Robert	2
6	Watson	Denise	6
7	Hardy	Thomas	5
8	O'Leary	Rhea	4
9	Locario	Paulo	6

Another left outer join

```
SELECT department_name AS dept_name,
    d.department_number AS dept_no,
    last_name
FROM departments d
    LEFT JOIN employees e
    ON d.department_number = e.department_number
ORDER BY department_name
```

	DEPT_NAME	DEPT_NO	LAST_NAME
1	Accounting	1	Hernandez
2	Maintenance	5	Hardy
3	Operations	3	(null)
4	Payroll	2	Simonian
5	Payroll	2	Aaronsen
6	Payroll	2	Smith
7	Personnel	4	Jones
8	Personnel	4	O'Leary

Another right outer join

```
SELECT department_name AS dept_name,
    e.department_number AS dept_no,
    last_name
FROM departments d
    RIGHT JOIN employees e
    ON d.department_number = e.department_number
ORDER BY department_name
```

	DEPT_NAME	DEPT_NO	LAST_NAME
1	Accounting	1	Hernandez
2	Maintenance	5	Hardy
3	Payroll	2	Aaronsen
4	Payroll	2	Simonian
5	Payroll	2	Smith
6	Personnel	4	Jones
7	Personnel	4	O'Leary
8	(null)	6	Locario
9	(null)	6	Watson

A full outer join

```
SELECT department_name AS dept_name,
    d.department_number AS d_dept_no,
    e.department_number AS e_dept_no,
    last_name
FROM departments d
    FULL JOIN employees e
    ON d.department_number = e.department_number
ORDER BY department_name
```

	DEPT_NAME	D_DEPT_NO	E_DEPT_NO	LAST_NAME
1	Accounting	1	1	Hernandez
2	Maintenance	5	5	Hardy
3	Operations	3	(null)	(null)
4	Payroll	2	2	Simonian
5	Payroll	2	2	Smith
6	Payroll	2	2	Aaronsen
7	Personnel	4	4	Jones
8	Personnel	4	4	O'Leary
9	(null)	(null)	6	Locario
10	(null)	(null)	6	Watson

Implicit syntax with a left outer join

```
SELECT department_name AS dept_name,
    dpt.department_number AS dept_no,
    last_name
FROM departments dpt, employees emp
WHERE dpt.department_number = emp.department_number (+)
ORDER BY department_name
```

	DEPT_NAME	DEPT_NO	LAST_NAME
1	Accounting	1	Hernandez
2	Maintenance	5	Hardy
3	Operations	3	(null)
4	Payroll	2	Simonian
5	Payroll	2	Aaronsen
6	Payroll	2	Smith
7	Personnel	4	Jones
8	Personnel	4	O'Leary

Implicit syntax with a right outer join

```
SELECT department_name AS dept_name,
    emp.department_number AS dept_no,
    last_name
FROM departments dpt, employees emp
WHERE dpt.department_number (+) = emp.department_number
ORDER BY department_name
```

	DEPT_NAME	DEPT_NO	LAST_NAME
1	Accounting	1	Hernandez
2	Maintenance	5	Hardy
3	Payroll	2	Aaronsen
4	Payroll	2	Simonian
5	Payroll	2	Smith
6	Personnel	4	Jones
7	Personnel	4	O'Leary
8	(null)	6	Locario
9	(null)	6	Watson

Cross join example

```
SELECT departments.department_number, department_name,
       employee_id, last_name
FROM departments
CROSS JOIN employees
ORDER BY departments.department_number
```

	DEPARTMENT_NUMBER	DEPARTMENT_NAME	EMPLOYEE_ID	LAST_NAME
1	1	Accounting	4	Hernandez
2	1	Accounting	3	Simonian
3	1	Accounting	9	Locario
4	1	Accounting	8	O'Leary
5	1	Accounting	7	Hardy
6	1	Accounting	6	Watson
7	1	Accounting	5	Aaronsen

Same cross join with the implicit syntax

```
SELECT departments.department_number, department_name,
    employee_id, last_name
FROM departments, employees
ORDER BY departments.department_number
```

	DEPARTMENT_NUMBER	DEPARTMENT_NAME	EMPLOYEE_ID	LAST_NAME
1	1	Accounting	4	Hernandez
2	1	Accounting	3	Simonian
3	1	Accounting	9	Locario
4	1	Accounting	8	O'Leary
5	1	Accounting	7	Hardy
6	1	Accounting	6	Watson
7	1	Accounting	5	Aaronsen

A self-join example

```
SELECT DISTINCT v1.vendor_name, v1.vendor_city,
    v1.vendor_state
FROM vendors v1 JOIN vendors v2
    ON (v1.vendor_city = v2.vendor_city) AND
       (v1.vendor_state = v2.vendor_state) AND
       (v1.vendor_id <> v2.vendor_id)
ORDER BY v1.vendor_state, v1.vendor_city
```

	VENDOR_NAME	VENDOR_CITY	VENDOR_STATE
1	AT&T	Phoenix	AZ
2	Computer Library	Phoenix	AZ
3	Wells Fargo Bank	Phoenix	AZ
4	Aztek Label	Anaheim	CA
5	Blue Shield of California	Anaheim	CA
6	ASC Signs	Fresno	CA
7	Abbey Office Furnishings	Fresno	CA
8	BFI Industries	Fresno	CA

Basic set operators

- Union (R ∪ S)
- Difference (R – S)
- Intersection (R ∩ S)

Example R and S Relations:

R
1
2
3
4
5

S
1
2
3
4

Question: what answer do we get when we apply these set operations to R and S?

A union example

```
SELECT 'Active' AS source, invoice_number, invoice_date,
invoice_total
FROM invoices
WHERE (invoice_total - payment_total - credit_total) > 0
UNION
SELECT 'Paid' AS source, invoice_number, invoice_date,
invoice_total
FROM invoices
WHERE (invoice_total - payment_total - credit_total) <= 0
ORDER BY invoice_total DESC
```

	SOURCE	INVOICE_NUMBER	INVOICE_DATE	INVOICE_TOTAL
1	Paid	0-2058	08-MAY-14	37966.19
2	Paid	P-0259	16-APR-14	26881.4
3	Paid	0-2060	08-MAY-14	23517.58
4	Active	40318	18-JUL-14	21842
5	Active	P-0608	11-APR-14	20551.18
6	Active	0-2436	07-MAY-14	10976.06

The Customers table

CUSTOMER_LAST_NAME	CUSTOMER_FIRST_NAME
Anders	Maria
Trujillo	Ana
Moreno	Antonio
Hardy	Thomas
Berglund	Christina
Moos	Hanna

The Employees table

EMPLOYEE_ID	LAST_NAME	FIRST_NAME	DEPARTMENT_NUMBER
1	Smith	Cindy	2
2	Jones	Elmer	4
3	Simonian	Ralph	2
4	Hernandez	Olivia	1
5	Aaronsen	Robert	2
6	Watson	Denise	6
7	Hardy	Thomas	5
8	O'Leary	Rhea	4
9	Locario	Paulo	6

A difference example

```
SELECT customer_first_name, customer_last_name
FROM customers
MINUS
SELECT first_name, last_name
FROM employees
ORDER BY customer_last_name
```

	CUSTOMER_FIRST_NAME	CUSTOMER_LAST_NAME
1	Maria	Anders
2	Christina	Berglund
3	Art	Braunschweiger
4	Donna	Chelan

An intersection example

```
SELECT customer_first_name, customer_last_name
FROM customers
INTERSECT
SELECT first_name, last_name
FROM employees
```

	CUSTOMER_FIRST_NAME	CUSTOMER_LAST_NAME
1	Thomas	Hardy

Next week

- Aggregate queries and subqueries
- Quiz #2