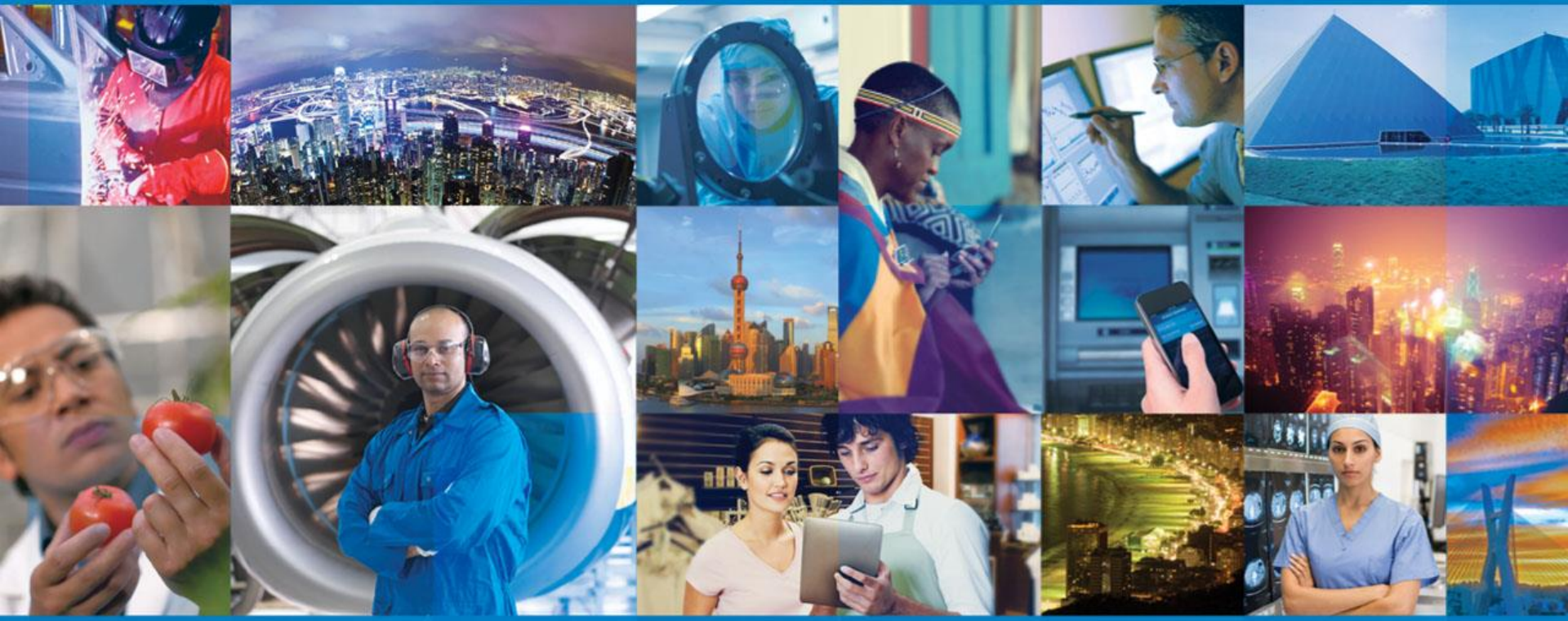
Relational Database Management System



Education, Training and Assessment

We enable you to leverage knowledge anytime, anywhere!

Infosys® | Building Tomorrow's Enterprise

Copyright Guideline

Confidential Information

- This Document is confidential to Infosys Limited. This document contains information and data that Infosys considers confidential and proprietary (“Confidential Information”).
- Confidential Information includes, but is not limited to, the following:
  - Corporate and Infrastructure information about Infosys
  - Infosys’ project management and quality processes
  - Project experiences provided included as illustrative case studies
- Any disclosure of Confidential Information to, or use of it by a third party, will be damaging to Infosys.
- Ownership of all Infosys Confidential Information, no matter in what media it resides, remains with Infosys.
- Confidential information in this document shall not be disclosed, duplicated or used – in whole or in part – for any purpose other than reading without specific written permission of an authorized representative of Infosys.
- This document also contains third party confidential and proprietary information. Such third party information has been included by Infosys after receiving due written permissions and authorizations from the party/ies. Such third party confidential and proprietary information shall not be disclosed, duplicated or used – in whole or in part – for any purpose other than reading without specific written permission of an authorized representative of Infosys.

Course Information

- Course Code: CCFP4.1-RDBMS
- Course Name: Relational Database Management System
- Document Number: RDBMS-03
- Version Number: 4.1

Introduction to Joins

- Scenario

  Manager of EasyShop retail chain wants to know details of items and available quantity in their retail outlets

- Observation
  - Item details and stock details are present in different tables
  - Manager requires specific information that are present in those tables

- How to retrieve the required data?

By joining tables

Types of JOIN

- CROSS JOIN
- INNER JOIN
- Outer join
  - LEFT OUTER JOIN
  - RIGHT OUTER JOIN
  - FULL OUTER JOIN
- Self join

CROSS JOIN

Guided Activity: Execute the following statement and discuss the result:

```
SELECT * FROM item CROSS JOIN retailstock;
```

CROSS JOIN

```
SELECT * FROM item CROSS JOIN retailstock;
```

If table1 has m rows and table2 has n rows then the output will have m*n rows

item

itemcode	description
I1001	Britannia Marie
I1002	Taj Mahal Tea
I1003	Best Rice

retailstock

retailoutletid	itemcode	qtyavailable	retailunitprice
R1001	I1001	25	1600
R1002	I1003	50	6600

Output

itemcode	description	retailoutletid	itemcode	qtyavailable	retailunitprice
I1001	Britannia Marie	R1001	I1001	25	1600
I1001	Britannia Marie	R1002	I1003	50	6600
I1002	Taj Mahal Tea	R1001	I1001	25	1600
I1002	Taj Mahal Tea	R1002	I1003	50	6600
I1003	Best Rice	R1001	I1001	25	1600
I1003	Best Rice	R1002	I1003	50	6600

CROSS JOIN

Did the manager get the specific information with the help of cross join?

NO!

How to realize the requirement ?

USING INNER JOIN

INNER JOIN

- Scenario
  - The manager of EasyShop retail chain wants to know details of items and the quantity available in their retail outlets

```
SELECT i.itemcode, description, retailoutletid, qtyavailable FROM
item i INNER JOIN retailstock rs ON i.itemcode = rs.itemcode;
```

item

itemcode	description
I1001	Britannia Marie
I1002	Taj Mahal Tea
I1003	Best Rice

retailstock

retailoutletid	itemcode	qtyavailable	retailunitprice
R1001	I1001	25	1600
R1002	I1003	50	6600

Required output

itemcode	description	retailoutletid	qtyavailable
I1001	Britannia Marie	R1001	25
I1003	Best Rice	R1002	50

Working of INNER JOIN

```
SELECT i.itemcode, description, retailoutletid, qtyavailable FROM
item i INNER JOIN retailstock rs ON i.itemcode = rs.itemcode;
```

item

itemcode	description
I1001	Britannia Marie
I1002	Taj Mahal Tea
I1003	Best Rice

retailstock

retailoutletid	itemcode	qtyavailable	retailunitprice
R1001	I1001	25	1600
R1002	I1003	50	6600

Output

itemcode	description	retailoutletid	itemcode	qtyavailable	
I1001	Britannia Marie	R1001	I1001	25	✓
I1001	Britannia Marie	R1002	I1003	50	✗
I1002	Taj Mahal Tea	R1001	I1001	25	✗
I1002	Taj Mahal Tea	R1002	I1003	50	✗
I1003	Best Rice	R1001	I1001	25	✗
I1003	Best Rice	R1002	I1003	50	✓

Working of INNER JOIN

```
SELECT i.itemcode, description, retailoutletid, qtyavailable FROM
item i INNER JOIN retailstock rs ON i.itemcode = rs.itemcode;
```

Output

itemcode	description	retailoutletid	qtyavailable
I1001	Britannia Marie	R1001	25
I1003	Best Rice	R1002	50

INNER JOIN – Joining more than 2 tables (1 of 2)

- Scenario

  The Super Manager of EasyShop would like to know name of the suppliers and items with its quoted price for all the quotations with quotation status as closed

  – The following information are available:

    - Supplier details (supplierid, suppliername, etc.) are available in supplier table
    - Quotation details (supplierid, itemcode, quotedprice, quotationstatus, etc.) are available in quotation table
    - Item details (itemcode, description, etc.) are available in item table

- Observation

  – Three tables need to be joined to get the desired output

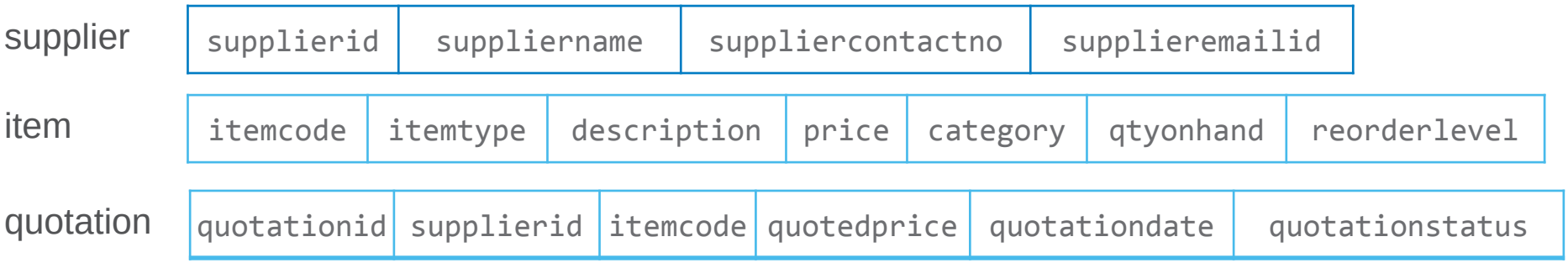
  – The common column supplierid is present in supplier and quotation table

  – The common column itemcode is present in item and quotation table

  – Quotation status ‘Closed’ can be checked from quotationstatus column

How to retrieve the required data?

INNER JOIN – Joining more than 2 tables (2 of 2)



At least N-1 conditions are required to join N tables

Solution

```
SELECT s.suppliername, i.description, q.quotedprice
FROM supplier s
INNER JOIN quotation q
ON s.supplierid = q.supplierid
INNER JOIN item i
ON i.itemcode = q.itemcode
WHERE UPPER(q.quotationstatus) ='CLOSED';
```

Guided activity

- The management of Easy Shop would like to know the id and name of customers whose total bill amount is more than 7000.

```
SELECT c.customerid, c.customername
FROM customer c
JOIN purchasebill p ON c.customerid=p.customerid
GROUP BY c.customerid, c.customername
HAVING SUM(billamount) >7000;
```

Guided Activity: CCFP4.1-RDBMSAssignments - Assignments on Join - 1a, 1b (Estimated Time: 75 mins.)

Scenario

List **all** the employees along with the locations of the outlets they are working. This should also include employees who are not allocated to any retail outlet

employee

empid	empname	worksin
1001	George	R1001
1002	Kevin	R1001
1003	Lisa	R1001
1004	Allen	
1005	Peter	R1002
1006	John	R1002

retailoutlet

retailoutletid	retailoutletlocation	retailoutletmanagerid
R1001	California	1002
R1002	New York	1006
R1003	Dallas	

Required output

empid	empname	retailoutletid	retailoutletlocation
1001	George	R1001	California
1002	Kevin	R1001	California
1003	Lisa	R1001	California
1005	Peter	R1002	New York
1006	John	R1002	New York
1004	Allen	NULL	NULL

Guided Activity: Try this with inner join
(Estimated Time: 10 mins.)

Scenario

List **all** the employees along with the locations of the outlets they are working. This should also include employees who are not allocated to any retail outlet

Did you get the required data with Inner Join?

NO

Let's try OUTER JOIN

Understanding LEFT OUTER JOIN

```
SELECT e.empid, e.empname, r.retailoutletid, r.retailoutletlocation FROM
employee e LEFT OUTER JOIN retailoutlet r ON e.worksin = r.retailoutletid;
```

employee

empid	empname	worksin
1001	George	R1001
1002	Kevin	R1001
1003	Lisa	R1001
1004	Allen	NULL
1005	Peter	R1002
1006	John	R1002

retailoutlet

retailoutletid	retailoutletlocation	retailoutletmanagerid
R1001	California	1002
R1002	New York	1006
R1003	Dallas	NULL

Required output:

empid	empname	retailoutletid	retailoutletlocation
1001	George	R1001	California
1002	Kevin	R1001	California
1003	Lisa	R1001	California
1005	Peter	R1002	New York
1006	John	R1002	New York
1004	Allen	NULL	NULL

Can we get the same output by changing the order of the table in the query ?

Guided Activity: CCFP4.1-RDBMSAssignments – Assignments on Join - 2a, 2b, 3 (Estimated Time: 50 mins.)

A scenario

The Manager of EasyShop would like to know the customer names along with their wife's name

customer

customerid	customername	spouse	gender
2001	John	2004	M
2002	Jason	2005	M
2003	Smith		M
2004	Susan	2001	F
2005	Nancy	2002	F

Required output

customerid	husband	wife
2001	John	Susan
2002	Jason	Nancy

Self join

```
SELECT h.customerid, h.customername AS husband, w.customername AS wife
FROM customer h INNER JOIN customer w ON h.spouse = w.customerid AND
h.gender = 'M' ;
```

Working of Self-Join

```
SELECT h.customerid, h.customername AS husband, w.customername
AS wife FROM customer h INNER JOIN customer w
ON h.spouse = w.customerid AND h.gender = 'M' ;
```

customerid	customername	spouse	gender
2001	John	2004	M
2002	Jason	2005	M
2003	Smith		M
2004	Susan	2001	F
2005	Nancy	2002	F

h

customerid	customername	spouse	gender
2001	John	2004	M
2002	Jason	2005	M
2003	Smith		M
2004	Susan	2001	F
2005	Nancy	2002	F

w

customerid	customername	spouse	gender
2001	John	2004	M
2002	Jason	2005	M
2003	Smith		M
2004	Susan	2001	F
2005	Nancy	2002	F

Self join

h

customerid	customername	spouse	gender
2001	John	2004	M
2002	Jason	2005	M
2003	Smith		M
2004	Susan	2001	F
2005	Nancy	2002	F

w

customerid	customername	spouse	gender
2001	John	2004	M
2002	Jason	2005	M
2003	Smith		M
2004	Susan	2001	F
2005	Nancy	2002	F

customerid	husband	wife
2001	John	Susan
2002	Jason	Nancy

```
SELECT h.customerid, h.customername AS husband, w.customername
AS wife FROM customer h INNER JOIN customer w
ON h.spouse = w.customerid AND h.gender = 'M' ;
```

Guided activity

- The Super Manager of Easy Shop wants to generate a list of all the items with their unit price and category having the same category as that of item ‘Xbox Gamepad’

```
SELECT i1.itemcode, i1.description, i1.price, i1.category
FROM item i1 JOIN item i2
ON i2.category = i1.category
AND i2.description = 'Xbox gamepad';
```

Guided Activity: CCFP4.1-RDBMSAssignments – Assignments on Joins - 4 (Estimated Time: 40 mins.)

Guided activity

- The Super Manager of Easy Shop wants to generate a list of all other items with their unit price and category having the same category as that of item ‘Xbox Gamepad’

```
SELECT i1.itemcode, i1.description, i1.price, i1.category
FROM item i1 JOIN item i2
ON i2.category = i1.category
AND i2.description = 'Xbox gamepad'
AND i1. description <> 'Xbox gamepad';
```

Summary

- Joins
  - CROSS JOIN
  - INNER JOIN
  - Outer join
    - LEFT OUTER JOIN
    - RIGHT OUTER JOIN
  - Self join

Self-Study

Refer to NPTEL course: http://nptel.ac.in/courses.php

Course : Course : NPTEL >> Computer Science and Engineering >> **Database Design**

Videos:

Structured Query Language

Structured Query Language II

Refer to:

https://class.stanford.edu/courses/Home/Databases/Engineering/about

References

- Abraham Silberschatz, Henry Korth and S. Sudarshan, Database System Concepts Jan 27, 2010
- C.J. Date, Database Design and Relational Theory: Normal Forms and All That Jazz (Theory in Practice) Apr 24, 2012
- Kevin Loney, George Koch “Oracle 9i, The Complete reference” Oracle Press
- http://en.wikipedia.org/wiki/Database_normalization
- http://en.wikipedia.org/wiki/Entity%E2%80%93relationship_model
- http://www.w3schools.com/sql/default.asp
- http://docs.oracle.com/cd/E11882_01/server.112/e41084/toc.htm
- http://online.stanford.edu/course/intro-to-databases-winter-2014
- http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-830-database-systems-fall-2010/lecture-notes/
- http://www.techopedia.com/definition/1245/structured-query-language-sql

Thank You

Infosys® | Building Tomorrow's Enterprise