

Queries: Part 2 of 2

IS240 – DBMS

Lecture # 7 – 2010-02-22

M. E. Kabay, PhD, CISSP-ISSMP

Assoc. Prof. Information Assurance
School of Business & Management, Norwich University

<mailto:mekabay@gmail.com>

V: 802.479.7937

1

Copyright © 2010 Jerry Post & M. E. Kabay. All rights reserved.

Topics

- Multiple Tables (Intro & Distinct)
- Joining Tables
- SQL JOIN
- Syntax for Three Tables
- Multiple Tables (Many)
- Building a Query
- Joining Tables (Hints)
- Tables with Multiple Joins
- Table Alias
- Saved Query: Create View
- Updateable & Non-updateable Views

As usual, this material was created by Prof Jerry Post and reformatted by Prof M. E. Kabay for use in IS240 at Norwich University

2

Copyright © 2010 Jerry Post & M. E. Kabay. All rights reserved.

Multiple Tables (Intro & Distinct)

Sale	
SaleID	SELECT DISTINCT CustomerID
SaleDate	FROM Sale
EmployeeID	WHERE (SaleDate Between '01-Apr-2004'
CustomerID	And '31-May-2004')
SalesTax	ORDER BY CustomerID;

Field	CustomerID	SaleDate
Table	Sale	Sale
Sort	Ascending	
Criteria		Between '01-Apr-2004' And '31-May-2004'
Or		

CustomerID
6
8
14
19
22
24
28
36
37
38
39
42
50
57
58
63
74
80
90

List the CustomerID of everyone who bought something between 01-Apr-2004 and 31-May-2004.

3

Copyright © 2010 Jerry Post & M. E. Kabay. All rights reserved.

Problem: CustomerID not Ideal

- We would much rather see the name of the customer than the CustomerID only
- The Customer's last name is in the Customer table
- We will create a VIEW that *joins* the information from the Sale table with the information from the Customer table
- Think of the JOIN operation as creating a synthetic table that combines all the records from both tables
- This allows us to SELECT the right information, including LastName, from the combined records

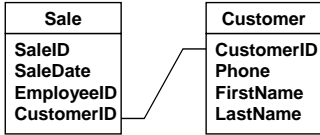
4

Copyright © 2010 Jerry Post & M. E. Kabay. All rights reserved.

Joining Tables



```
SELECT DISTINCT Sale.CustomerID, Customer.LastName
FROM Customer
INNER JOIN Sale ON Customer.CustomerID = Sale.CustomerID
WHERE (SaleDate Between '01-Apr-2004' And '31-May-2004')
ORDER BY Customer.LastName;
```

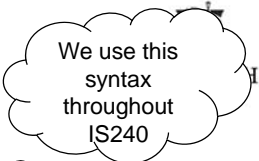


CustomerID	LastName
22	Adkins
57	Carter
38	Franklin
42	Froedje
63	Grimes
74	Hinton
36	Holland
6	Hopkins
50	Lee
58	McCain
...	

Field	CustomerID	LastName	SaleDate
Table	Sale	Customer	Sale
Sort		Ascending	
Criteria			Between '01-Apr-2004' And '31-May-2004'
Or			

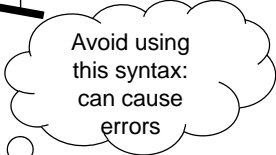
List LastNames of Customers who bought between 4/1/2004 and 5/31/2004.

SQL JOIN



```
FROM table1
INNER JOIN table2
ON table1.column = table2.column
SQL 92 syntax (Access and SQL Server)
```

```
FROM table1, table2
WHERE table1.column = table2.column
SQL 89 syntax (Oracle)
```



```
FROM table1, table2
JOIN table1.column = table2.column
Informal syntax
```

Syntax for Three Tables



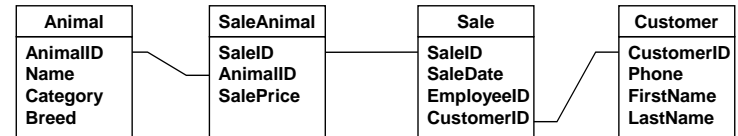
SQL '92 syntax to join three tables

```
FROM Table1
INNER JOIN
(Table2 INNER JOIN Table3
ON Table2.ColA = Table3.ColA)
ON Table1.ColB = Table2.ColB
```

Multiple Tables (Many)



```
SELECT DISTINCTROW Customer.LastName, Customer.Phone
FROM Customer INNER JOIN (Sale INNER JOIN (Animal INNER JOIN SaleAnimal
ON Animal.AnimalID = SaleAnimal.AnimalID) ON Sale.SaleID = SaleAnimal.SaleID)
ON Customer.CustomerID = Sale.CustomerID
WHERE ((Animal.Category='Cat') AND (Animal.Registered Is Not Null)
AND (Color Like '%White%') AND (SaleDate Between '01-Jun-2004'
And '31-Dec-2004'));
```



Field	LastName	Phone	Category	Registered	Color	SaleDate
Table	Customer	Customer	Animal	Animal	Animal	Sale
Sort	Ascending					
Criteria			'Cat'	Is Not Null	Like '%White%'	Between '01-Jun-2004' And '31-Dec-2004'
Or						

List the Last Name and Phone of anyone who bought a registered White cat between 6/1/2004 and 12/31/2004.

Building a Query

- List the Last Name and Phone of anyone who bought a registered White cat between 6/1/04 and 12/31/04.
- Identify the tables involved.
 - ❑ Look at the columns you want to see.
 - ✓ LastName, Phone: Customer
 - ❑ Look at the columns used in the constraints.
 - ✓ Registered, Color, Category: Animal
 - ✓ Sale Date: Sale
 - ❑ Find connector tables.
 - ✓ To connect Animal to Sale: SaleAnimal
- Select the desired columns and test the query.
- Enter the constraints.
- Set Order By columns.
- Add Group By columns.
- Add summary computations to the SELECT statement.

9

Copyright © 2010 Jerry Post & M. E. Kabay. All rights reserved.

Joining Tables (Hints)

- Build Relationships First
 - ❑ Drag and drop
 - ❑ From one side to many side
- Avoid multiple ties between tables
- SQL
 - ❑ FROM Table1
 - ❑ INNER JOIN Table2
 - ❑ ON Table1.ColA = Table2.ColB
- Join columns are often keys, but they can be any columns--as long as the domains (types of data) match.
- Multiple Tables
 - ❑ FROM (Table1
 - ❑ INNER JOIN Table2
 - ❑ ON T1.ColA = T2.ColB)
 - ❑ INNER JOIN Table3
 - ❑ ON T3.ColC = T3.ColD
- Shorter Notation
 - ❑ FROM T1, T2, T3
 - ❑ JOIN T1.ColA = T2.ColB
 - ❑ T1.ColC = T3.ColD
- Shorter Notation is not correct syntax, but it is easier to write.

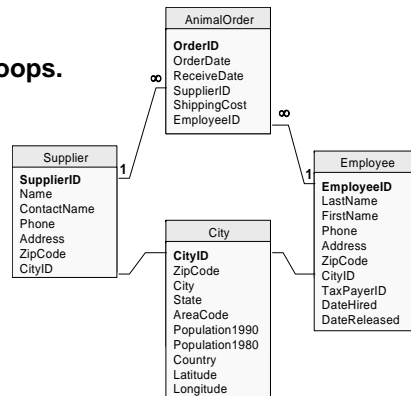
10

Copyright © 2010 Jerry Post & M. E. Kabay. All rights reserved.

Tables with Multiple Joins

- Potential problem with three or more tables.
- Access uses predefined relationships to automatically determine JOINS.
- JOINS might loop.
- Most queries will not work with loops.

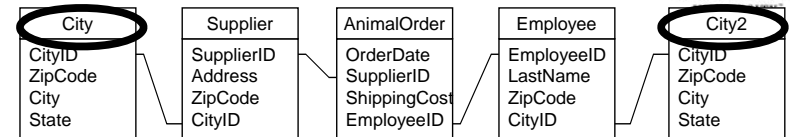
A query with these four tables with four JOINS would only return rows where the Employee had the same ZipCode as the Supplier. If you only need the Supplier city, just delete the JOIN between Employee and ZipCode. If you want both cities, add the ZipCode table again as a fifth table.



11

Copyright © 2010 Jerry Post & M. E. Kabay. All rights reserved.

Table Alias



```

SELECT Supplier.SID, Supplier.CityID, City.City, Employee.EID,
Employee.LastName, Employee.CityID, City2.City
FROM (City INNER JOIN Supplier ON City.CityID = Supplier.CityID) INNER
JOIN ((City AS City2 INNER JOIN Employee ON City2.CityID =
Employee.CityID) INNER JOIN AnimalOrder ON Employee.EmployeeID =
AnimalOrder.EmployeeID) ON Supplier.SupplierID = AnimalOrder.SupplierID;
  
```

SID	Supplier.CityID	City.City	EID	LastName	Employee.CityID	City2.City
4	7972	Middleboro	5	James	7083	Lincoln
2	10896	Springfield	1	Reeves	9201	Springfield
4	7972	Middleboro	3	Reasoner	8313	Philadelphia
9	10740	Columbia	8	Carpenter	10592	Springfield
5	10893	Smyrna	3	Reasoner	8313	Springfield

12

Copyright © 2010 Jerry Post & M. E. Kabay. All rights reserved.

Saved Query: Create View



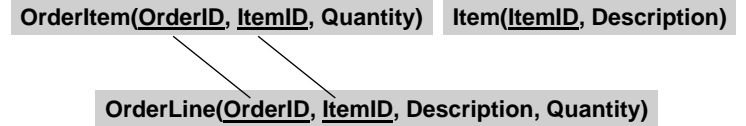
- Save a query
 - Faster: only enter once
 - Faster: only analyze once
- Any SELECT statement
- Can use the View within other SQL queries.

Examples:

```
CREATE VIEW Kittens AS
SELECT *
FROM Animal
WHERE (Category = 'Cat') AND
(Today - DateBorn < 180);
```

```
SELECT Avg(ListPrice)
FROM Kittens
WHERE (Color LIKE '%Black%');
```

Updateable Views



- To be updateable, a view must focus on one primary table. (OrderItem)
 - Goal is to change data in *only one table*. (OrderItem)
 - Data can be *displayed* from other tables. (Item)
 - Never include or attempt to change primary keys from more than one table.* (Item.ItemID)

Non-Updateable View



`OrderItem(OrderID, ItemID, Quantity)` `Item(ItemID, Description)`

121	57	3	57	Cat food
121	82	2	58	Dog food
122	57	1	59	Bird food

`OrderLine(OrderID, Item.ItemID, Description, Quantity)`

121	57	Cat food	3
121	82	Bird feeder	2
122	57	Cat food	1

32

If you attempt to change the Item.ItemID in the OrderLineView:
 You will simply change the primary key value in the Item table.
 It will not add a new row to the OrderItem table.

Homework



- By Sunday 28 Feb 2010
- Chapter 4 main text page 176:
 - Sally's Pet Store DB from your Student CD
 - Questions 8,10,12,14,16,18, 20