3 Database Queries

Peeking into Computer Science

© Jalal Kawash 2010

- Mandatory: Chapter 4 – Sections 4.6 & 4.7

Reading Assignment

- Can be found on:

http://pages.cpsc.ucalgary.ca/~kawash/peeking.html

- Includes all examples in the book
  - Numbered by exercise numbers

Example Access DB

Structured Query Language

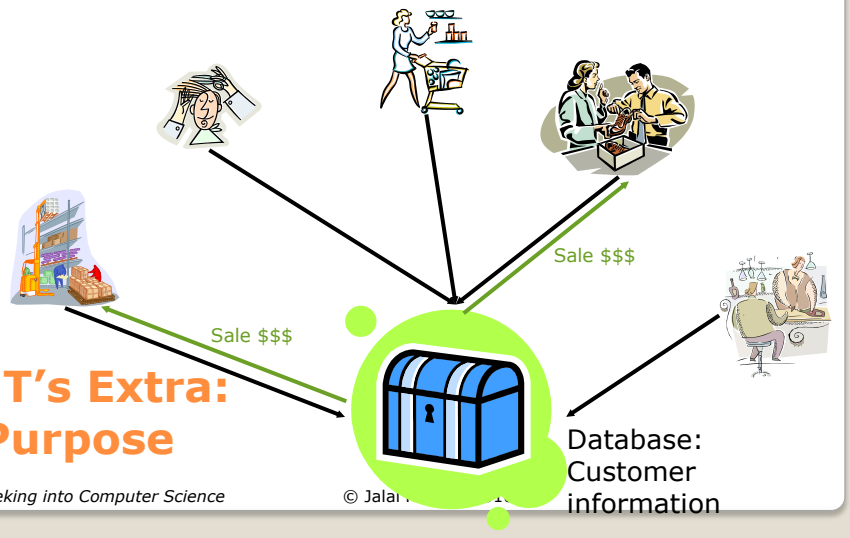
By the end of this section, you will be able to:

1. Name the two parts of SQL
2. Understand the function of each part
3. Formulate basic DDL & DML statements in SQL

Objectives

 © Jalal Kawash 2010

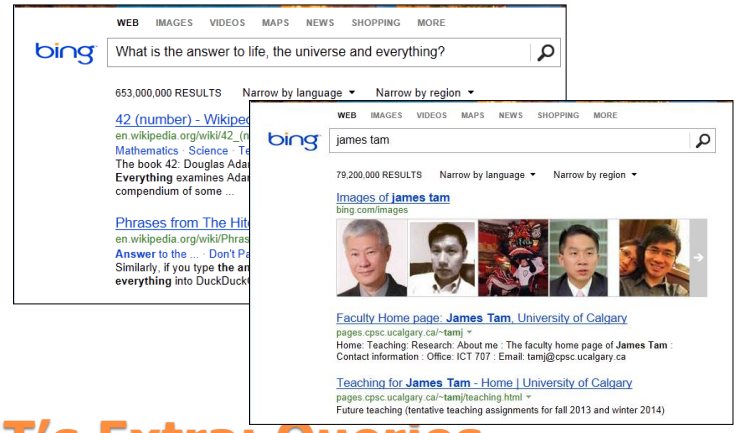
- To store & retrieve information



JT's Extra: Purpose

 © Jalal

- Queries are questions 'asked' of/to the database in order to retrieve information.



JT's Extra: Queries

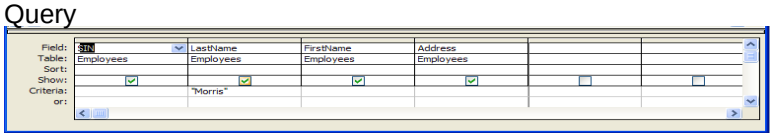
- Data retrieval occurs through the use of 'queries':
  - A query is a question asked of the data in the database.
  - Typically worded to show only the parts of the database for which the answer to the question is true.
  - **Example 1:** What is the SIN, name and pay rate of every employee in the Employees Table:

Employee Pay Rates : Select Query

SIN	LastName	FirstName	PayRate
123 115 323	Simcox	Cole	30
123 456 789	Smith	John	20
371 988 812	Carswell	Mary	30
413 754 621	Kennedy	Leon	30
444 638 047	Redfield	Claire	35

  - **Example 2:** What employees have the last name of Morris?

Query



JT's Extra: Retrieving Data Via Queries

- SQL JT's Extra: SQL = **S**tructured **Q**uery **L**anguage
- Programming language, specialized for databases
- Data Definition Language (DDL)
  - Defining the structure of the DB

JT's Extra: Creating the data (table)
- What fields?
- What will each field store?

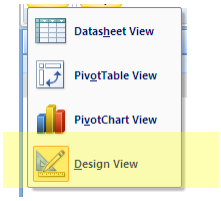
- Data Manipulation Language (DML)
  - Manipulating the contents of the DB

JT's Extra: Modifying the data (table)
- Insertions
- Deletions

Structured Query Language


```
CREATE TABLE EMPLOYEE
(
  SIN        CHAR(9),
  Fname      CHAR(15),
  Lname      CHAR(15),
  DOB        DATE,
  Gender     CHAR(6),
  Salary     NUMBER,
  Street     CHAR(30),
  City       CHAR(15),
  Pcode      CHAR(7),
  Dnumber    NUMBER
)
```

DDL



EMPLOYEE | Example6-1

Field Name	Data Type
SIN	Number
Fname	Text
Lname	Text
DOB	Date/Time
Gender	Text
Salary	Number
Number	Number
Street	Text
City	Text
Pcode	Text
Dnumber	Number

General | Lookup

Field Size	Long Integer
Format	
Decimal Places	Auto
Input Mask	000000000
Caption	
Default Value	
Validation Rule	
Validation Text	
Required	No
Indexed	Yes (No Duplicates)
Smart Tags	
Text Align	General

Access Table Creation

Recall: Data manipulation language modifies the tables

General format:

- <*Action*> into/from <*Table name*>

Example:

- Insert into Employees {some values}
- Delete from Employees {conditions for deletion }

JT's Extra: DML General Structure

```
INSERT INTO EMPLOYEE VALUES
('171717171', 'Debra', 'Beacon', '15-Aug-1961', 'Female', 70000, 15, 'Baron Hill', 'Calgary', 'T2X Y0Y', 1)
INSERT INTO EMPLOYEE VALUES
('181817178', 'Sam', 'Field', '17-Feb-1978', 'Male', 40000, 15, 'Kick Way', 'Calgary', 'Y2K K0K', 1)
INSERT INTO EMPLOYEE VALUES
('123456789', 'Rajeet', 'Folk', '30-Apr-1967', 'Male', 78000, 123, 'One Road', 'Toronto', 'H1H J9J', 2)
INSERT INTO EMPLOYEE VALUES
('987654321', 'Marie', 'Band', '12-Jan-1985', 'Female', 53500, 2828, 'Exit Close', 'Toronto', 'K8O O8K', 2)
INSERT INTO EMPLOYEE VALUES
('666333999', 'Saleh', 'Dice', '25-Mar-1970', 'Male', 90400, 66, 'Straight Way', 'Toronto', 'T4E T6B', 1)
```

EMPLOYEE

SIN	Fname	Lname	DOB	Gender	Salary	Number	Street	City	Pcode	Dnumber
171717171	Debra	Beacon	15-Aug-1961	Female	70000	15	Baron Hill	Calgary	T2X Y0Y	1
181817178	Sam	Field	17-Feb-1978	Male	40000	15	Kick Way	Calgary	Y2K K0K	1
12345679	Rajeet	Folk	30-Apr-1967	Male	78000	123	One Road	Toronto	H1H J9J	2
987654321	Marie	Band	12-Jan-1985	Female	53500	2828	Exit Close	Toronto	K8O O8K	2
666333999	Saleh	Dice	25-Mar-1970	Male	90400	66	Straight Way	Toronto	T4E T6B	3

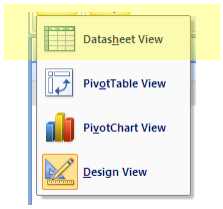
DML - Insertion


```
DELETE FROM EMPLOYEE
WHERE Gender = 'Male'
```

EMPLOYEE

SIN	Fname	Lname	DOB	Gender	Salary	Number	Street	City	Pcode	Dnumber
171717171	Debra	Beacon	15-Aug-1961	Female	70000	15	Baron Hill	Calgary	T2X Y0Y	1
~~181817178~~	~~Sam~~	~~Field~~	~~17-Feb-1978~~	~~Male~~	~~40000~~	~~15~~	~~Kick Way~~	~~Calgary~~	~~Y2K K0K~~	~~1~~
~~12345679~~	~~Rajeet~~	~~Folk~~	~~30-Apr-1967~~	~~Male~~	~~78000~~	~~123~~	~~One Road~~	~~Toronto~~	~~H1H J9J~~	~~2~~
987654321	Marie	Band	12-Jan-1985	Female	53500	2828	Exit Close	Toronto	K8O O8K	2
~~666333999~~	~~Saleh~~	~~Dice~~	~~25-Mar-1970~~	~~Male~~	~~90400~~	~~66~~	~~Straight Way~~	~~Toronto~~	~~T4E T6B~~	~~3~~

DML - Deletion



EMPLOYEE | Example6-1

SIN	Fname	Lname	DOB	Gender	Salary
123456789	Rajeet	Folk	30-Apr-67	Male	7
171717171	Debra	Beacon	15-Aug-61	Female	7
181817178	Sam	Field	17-Feb-78	Male	4
666333999	Saleh	Dice	25-Mar-70	Male	9
987654321	Marie	Band	12-Jan-85	Female	5

Access Insertion and Deletion

SQL Queries

Questioning the database

By the end of this section, you will be able to:

1. Know the basic parts of speech in SQL
2. Formulate SQL queries
3. Use set operations in SQL queries
4. Use complex logic in SQL queries

Objectives

EMPLOYEE

SIN	Fname	Lname	DOB	Gender	Salary	Number	Street	City	Pcode	Dnumber
171717171	Debra	Beacon	15-Aug-1961	Female	70000	15	Baron Hill	Calgary	T2X Y0Y	1
181817178	Sam	Field	17-Feb-1978	Male	40000	15	Kick Way	Calgary	Y2K K0K	1
123456789	Rajeet	Folk	30-Apr-1967	Male	78000	123	One Road	Toronto	H1H J9J	2
987654321	Marie	Band	12-Jan-1985	Female	53500	2828	Exit Close	Toronto	K8O O8K	2
666333999	Saleh	Dice	25-Mar-1970	Male	90400	66	Straight Way	Toronto	T4E T6B	3

DEPARTMENT

Dnumber	Dname	MGR_SIN	StartDate
1	IT	171717171	12-Feb-2008
2	Finance	123456789	1-Mar-2002
3	Marketing	666333999	1-Jan-2005

PROJECT

Pnumber	Pname	Location	Dnumber
1	Web Shopping	Calgary	1
2	Network Upgrade	Calgary	1
3	New Benefits	Toronto	2
4	Product XT345	Toronto	3

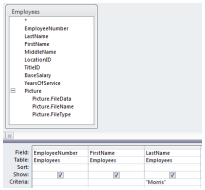
PROJ_EMP

SIN	Pnumber	Hours
171717171	1	15
171717171	2	20
171717171	4	5
181817178	1	30
181817178	2	10
123456789	3	40
666333999	4	40

Example Database

- Ways of submitting questions to the DB:

1. Query By Example (QBE)
   - JT: In Access it's graphical



2. Using SQL
   - Text

Formal SQL structure
- SELECT <columns of table(s)>
- FROM <table(s) >
- WHERE <Boolean expression(s)>
- ORDER BY[1]

Queries

Writing SQL queries can be complex

SQL queries are powerful (i.e., unusual searches can show useful information)

SQL is commonly used (standard)

JT's Extra: My $0.02 On Forming Queries Using SQL

Informal English description

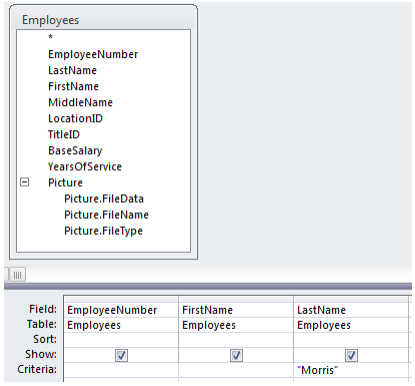
Given some condition(s) is/are met what rows and columns of what tables will appear.

Example:

If last name is "Morris" show the employee number, first name and last name from the employees table.

JT's Extra: Basic Format Of SQL Queries

- Specifying the query in the form of QBE (Query by example) in MS-Access



JT's Extra: Basic Format Of SQL Queries (2)

Formal SQL structure

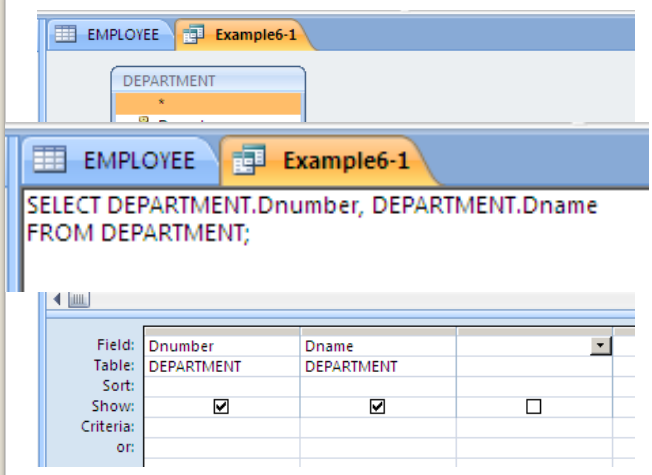
- SELECT <columns of table(s)>
- FROM <table(s) >
- WHERE <Boolean expression(s)>
- ORDER BY[1]

1 Optional section: used to format or rank query results

JT's Extra: Basic Format Of SQL Queries (3)

- **SELECT**
  - Employees.EmployeeNumber,
  - Employees.FirstName,
  - Employees.LastName
- **FROM** EMPLOYEES
- **WHERE**
  EMPLOYEES.LastName="Morris";

JT's Extra: Example SQL Query



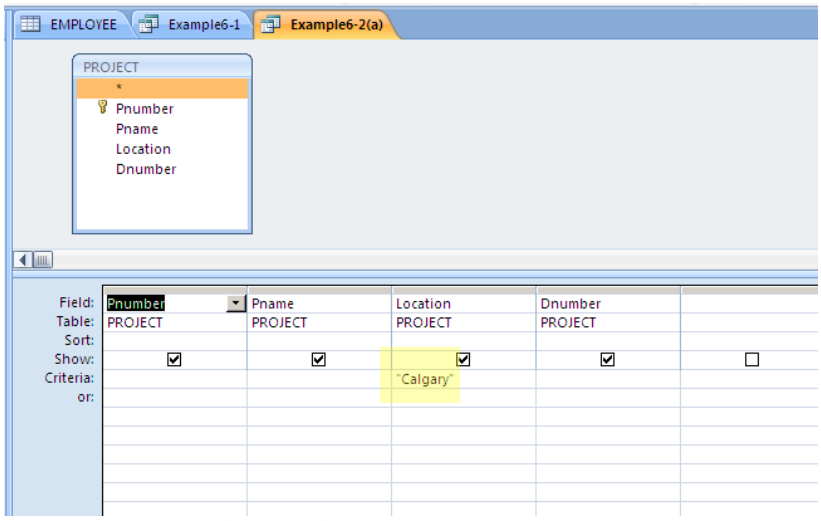
QBE – Projection


```
SELECT   Dnumber, Dname
FROM     DEPARTMENT
```

DEPARTMENT

Dnumber	Dname	MGR_SIN	StartDate
1	IT	17[illegible]1	12-F[illegible]-2[illegible]08
2	Finance	1234[illegible]789	1-Ma[illegible]002
3	Marketing	66[illegible]0	1-J[illegible]n-20[illegible]

SQL - Projection



QBE - Selection


```
SELECT   Pnumber, Pname, Location, Dnumber
FROM     PROJECT
WHERE    Location = 'Calgary'
```

PROJECT

Pnumber	Pname	Location	Dnumber
1	Web Shopping	Calgary	1
2	Network Upgrade	Calgary	1
~~3~~	~~New Benefits~~	~~Toronto~~	~~2~~
~~4~~	~~Product XT345~~	~~Toronto~~	~~3~~

SQL - Selection

- Poker is the wildcard in card game
- It can be substituted for any other card
- In the example it can take the place of: Ace, King, Queen, Jack or Ten to give you a "Royal Flush"

JT's Extra: Wildcards

- The 'wildcard' character can stand for any number of characters in the position that it's placed:
  - Example queries that follow will be from the Employees table:

Employees : Table

SIN	LastName	FirstName	Address	City	Province	PostalCode
123 115 323	Simcox	Cole	311 Ocean View Drive	Vancouver	British Columbia	T1N-4N9
123 456 789	Smith	John	123 Peanut Lane	Calgary	Alberta	T1N-3N4
371 988 812	Carswell	Mary	425 Remington Ave	Calgary	Alberta	T3N-7N4
413 754 621	Kennedy	Leon	808, 4900 Wildman A	Racoon City	Alberta	T2S-1M0
444 638 047	Redfield	Claire	653 Wildpark Place	Racoon City	Alberta	T2S-1M0
456 438 624	Lemoy	Leonard	55 Logic Way	Vulcan	Alberta	VS1-3N3
456 789 123	Cartman	Eric	456 Lynchview Road	Southpark	Alberta	S0S-9A9
456 789 124	Simpson	Homer	59 Evergreen Terrace	Springfield	Alberta	N1E-7X6
456 889 123	Flanders	Ned	60 Evergreen Terrace	Springfield	Alberta	N1E-7X6
620 451 097	Williams	Amanda	25 Rodeo Drive	Edmonton	Alberta	V6N-6N5
638 666 670	Cartland	Douglas	1109, 4944 Dalworth	Silent Hill	Alberta	S6N-9X9
666 666 666	Morris	Heather	7 Luckstone Dr	Silent Hill	Alberta	T3A-3H1
666 666 667	Mason	Harry	7 Luckstone Dr	Silent Hill	Alberta	T3A-3H1
666 666 668	Sunderland	James	7 Heartbroken Ave	Silent Hill	Alberta	T3A-2E6
666 666 669	Wolf	Claudia	66 Twisted View	Silent Hill	Alberta	T1N-3O4
670 380 456	Edgar	Maureen	300, Lockinvar Road	Calgary	Alberta	T4P-3N9

JT's Extra: Using The Wildcard In Queries

- Examples:
  - Which employees have a last name that begins with 'm'?

	LastName	FirstName
▶	Mason	Harry
	Morris	Heather

Field:	LastName	FirstName
Table:	Employees	Employees
Sort:		
Show:	☑	☑
Criteria:	Like "m*"	

  - Which employees have a last name ends with 's'?

	LastName	FirstName
▶	Flanders	Ned
	Morris	Heather
	Williams	Amanda

Field:	LastName	FirstName
Table:	Employees	Employees
Sort:		
Show:	☑	☑
Criteria:	Like "*s"	

JT's Extra: Using The Wildcard In Queries (Access)

 © Jalal Kawash 2009

- Which employees have the letter 'a' anywhere in their first name?

	LastName	FirstName
▶	Cartland	Douglas
	Edgar	Maureen
	Lemoy	Leonard
	Mason	Harry
	Morris	Heather
	Redfield	Claire
	Sunderland	James
	Williams	Amanda
	Wolf	Claudia
	Carswell	Mary

Field:	LastName	FirstName
Table:	Employees	Employees
Sort:		
Show:	☑	☑
Criteria:		Like "*a*"
or:		

JT's Extra: Using The Wildcard In Queries (Access: 2)

 © Jalal Kawash 2009

```
SELECT  *
FROM    PROJECT
WHERE   Location = 'Calgary'
```

PROJECT

Pnumber	Pname	Location	Dnumber
1	Web Shopping	Calgary	1
2	Network Upgrade	Calgary	1
3	New Benefits	Toronto	2
4	Product XT345	Toronto	3

Wild Cards

- JT's Extra: As mentioned each table can be viewed as a set of information.

EMPLOYEES (TABLE/SET)
* 456 789 123, Cartman Eric, Southpark
* 456 789 124, Simpson Homer, Springfield
* 666 666 666, Morris Heather, Silent Hill
* 666 666 667, Mason Harry, Silent Hill
* 670 380 456, Edgar Maureen, Calgary

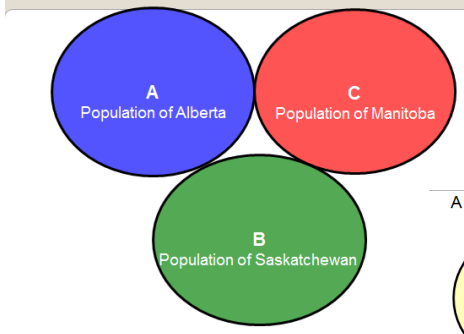
Departments (TABLE/SET)
* 1, Human Resources
* 2, Marketing
* 3, Finance
* 4, Management Information Systems

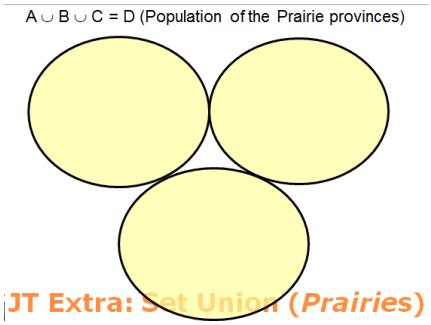
TimeBilled (TABLE/SET)
* 8, 456 789 123, 2, 10/1/2007, 80
* 9, 456 789 124, 2, 10/1,2007, 60
* 14, 666 666 666, 3, 10/1/2007, 50
* 15, 666 666 667, 3, 10/1/2007, 50
* 18, 670 380 456, 4, 10/1/2007, 40

- UNION
- INTERSECT
- MINUS

JT's Extra: Set Operations On Databases

 © Jalal Kawash 2010

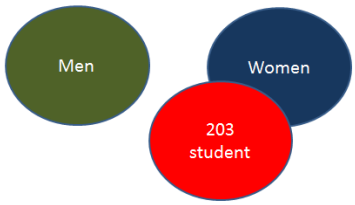


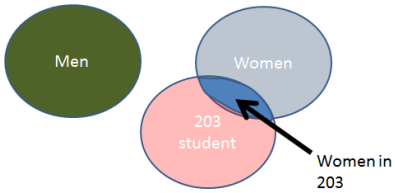


JT Extra: Set Union (*Prairies*)

JT's Extra: Union (Logical "OR")

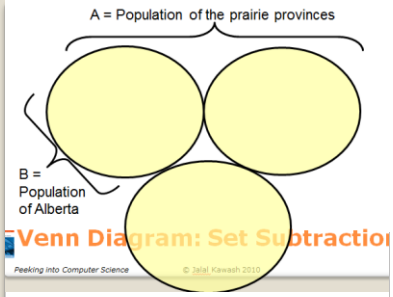
 © Jalal Kawash 2010

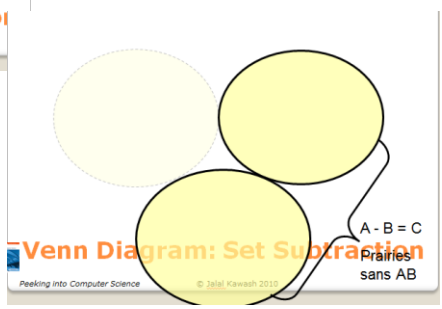




(203 student) AND (Women)

JT's Extra: Intersection (Logical "AND")





JT's Extra: Subtraction "Minus"

- **Format:**

  <SQL Query 1>

  SET OPERATION

  <SQL Query 2>

JT's Extra: Forming SQL Queries Using The Set Operations


```
SELECT   *
FROM     PROJECT
WHERE    Location = 'Calgary'
```

SQL – Set Operations

```
SELECT   *
FROM     PROJECT
WHERE    Location = 'Calgary'

SELECT   *
FROM     PROJECT
WHERE    Location = 'Toronto'
```

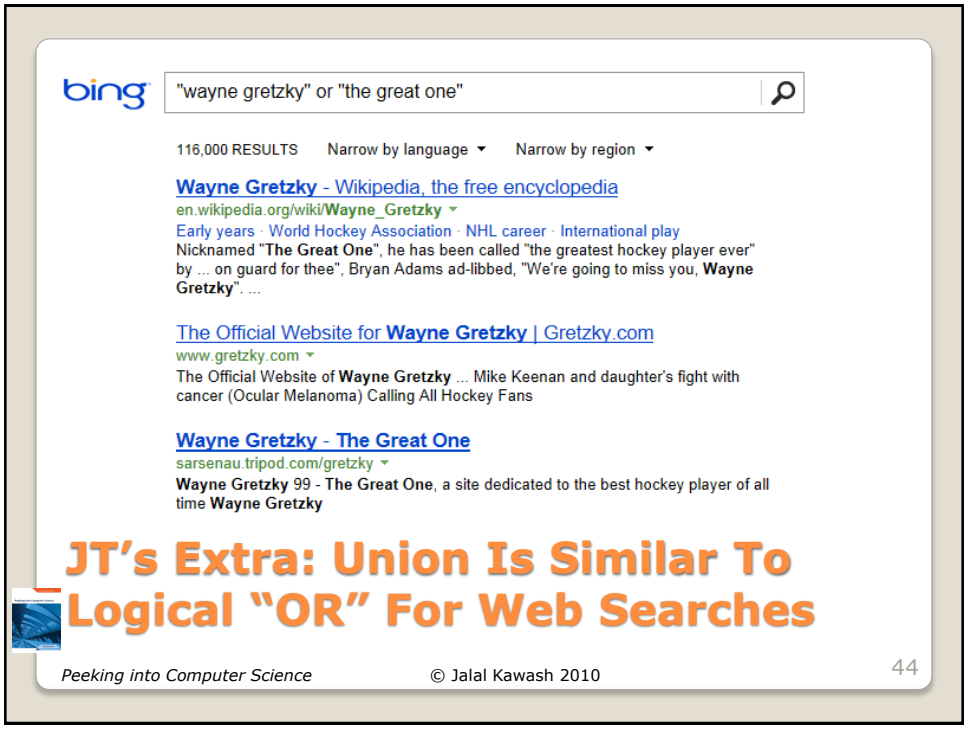
SQL – Set Operations

 © Jalal Kawash 2010

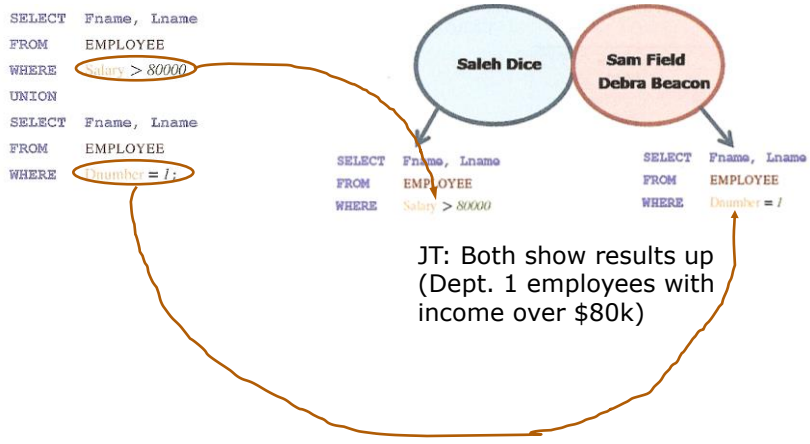
```
SELECT   *
FROM     PROJECT
WHERE    Location = 'Calgary'
UNION
SELECT   *
FROM     PROJECT
WHERE    Location = 'Toronto'
```

SQL – Set Operations

 © Jalal Kawash 2010



JT's Extra: Union Is Similar To Logical "OR" For Web Searches



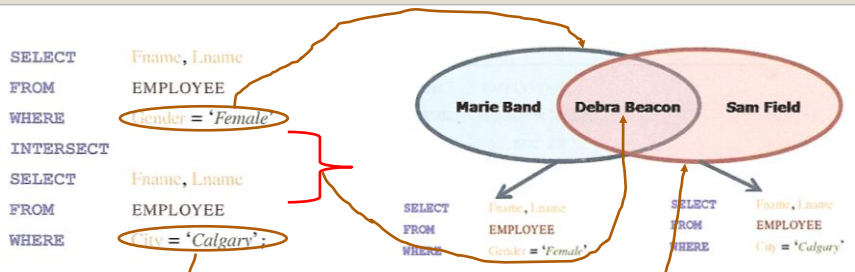
JT's Extra (From Textbook): Union

- Cannot be done in design view, only in SQL view

Example6-5(a) | Example6-5(b)

```
SELECT * FROM PROJECT
WHERE Location = "Calgary"
UNION SELECT * FROM PROJECT
WHERE Location = "Toronto";
```

Union in Access

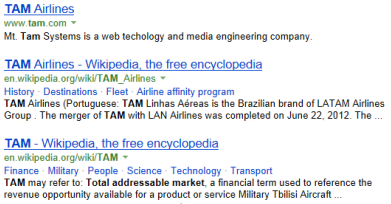


JT: show query results that meets both criteria (Female employees that live in Calgary)

JT's Extra (From Textbook): Intersect

Websites containing 'James'

Websites containing 'Tam'



Websites containing 'James' AND 'Tam'

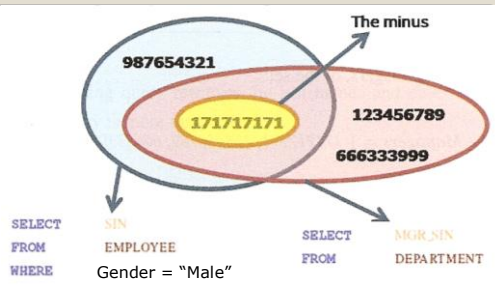
FYI: Search engine input: james tam (defaults to 'AND')

JT's Extra: Intersection Is Similar To Logical "AND" For Web Searches


```
SELECT      SIN
FROM        EMPLOYEE
WHERE       Gender = 'Female'
INTERSECT
SELECT      MGR_SIN
FROM        DEPARTMENT
```

SQL – Set Operations

```
SELECT   SIN
FROM     EMPLOYEE
WHERE    Gender = 'Male'
MINUS
SELECT   MGR_SIN
FROM     DEPARTMENT;
```



S1 = SIN of all male employees

MINUS

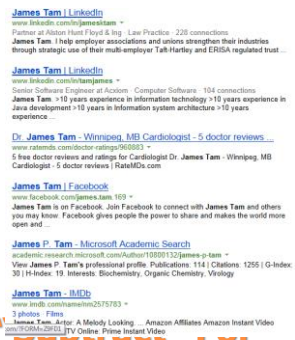
S2 = SIN of all employees that are managers

JT's Extra (From Textbook): Minus

Websites containing 'James' AND 'Tam'

Subtract all pages that contain 'ucalgary'

=



FYI: Search engine input:
james tam -ucalgary

JT's Extra: Minus Is Similar To "Subtract" For Web Searches

```
SELECT   SIN
FROM     EMPLOYEE
WHERE    Gender = 'Male'
MINUS
SELECT   MGR_SIN
FROM     DEPARTMENT
```

SQL – Set Operations

- No direct support
- Can use IN for intersect
- Can use NOT IN for Minus

INTERSECT and MINUS in Access

Example6-4(b)

```
SELECT SIN
FROM employee
WHERE Gender = 'Female'
AND SIN IN
(SELECT MGR_SIN
FROM DEPARTMENT);
```

INTERSECT in Access


```
SELECT SIN
FROM employee
WHERE Gender = 'Female'
AND SIN IN
(SELECT MGR_SIN
FROM DEPARTMENT);
```

Result
171717171

MGR_SIN
171717171
123456789
666333999

EMPLOYEE

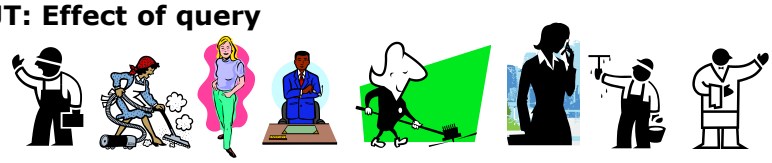
SIN	Fname	Lname	DOB	Gender	S
171717171	Debra	Beacon	15-Aug-1961	Female	7
181817178	Sam	Field	17-Feb-1978	Male	4
123456789	Rajeet	Polk	30-Apr-1967	Male	7
987654321	Marie	Band	12-Jan-1985	Female	5
666333999	Salen	Dice	23-Mar-1970	Male	9

How IN Works

Example6-4(c)

```
SELECT EMPLOYEE.Sin
FROM EMPLOYEE
WHERE (((EMPLOYEE.Sin) Not In
 (SELECT DEPARTMENT.MGR_SIN FROM DEPARTMENT)));
```

JT: Effect of query



MINUS in Access

- Equals =
- Not equal !=
- Greater >
- Greater or equal >=
- Less than <
- Less than or equal <=

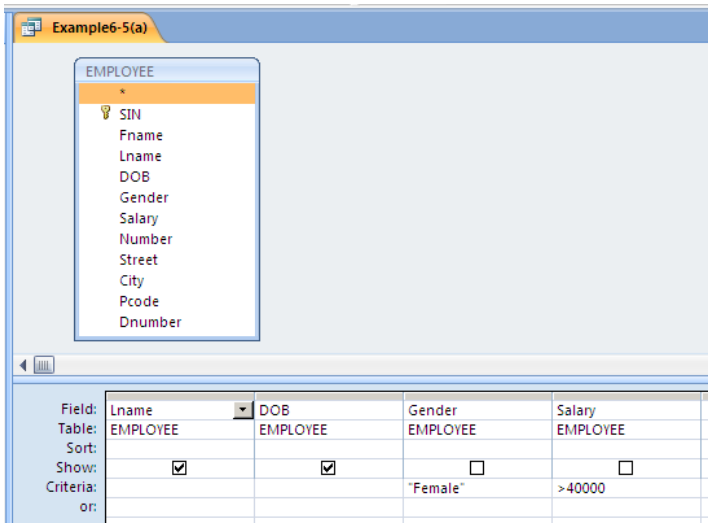
Comparison Operators

- AND
- OR
- NOT

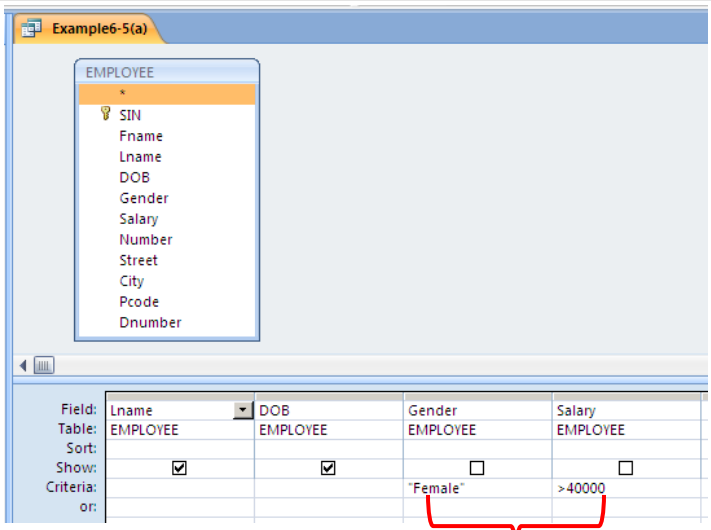
Boolean Operators


```
SELECT   Lname, DOB
FROM     EMPLOYEE
WHERE    Gender = 'Female'
         AND Salary > 40000
```

SQL – Example Query



QBE – Example Query



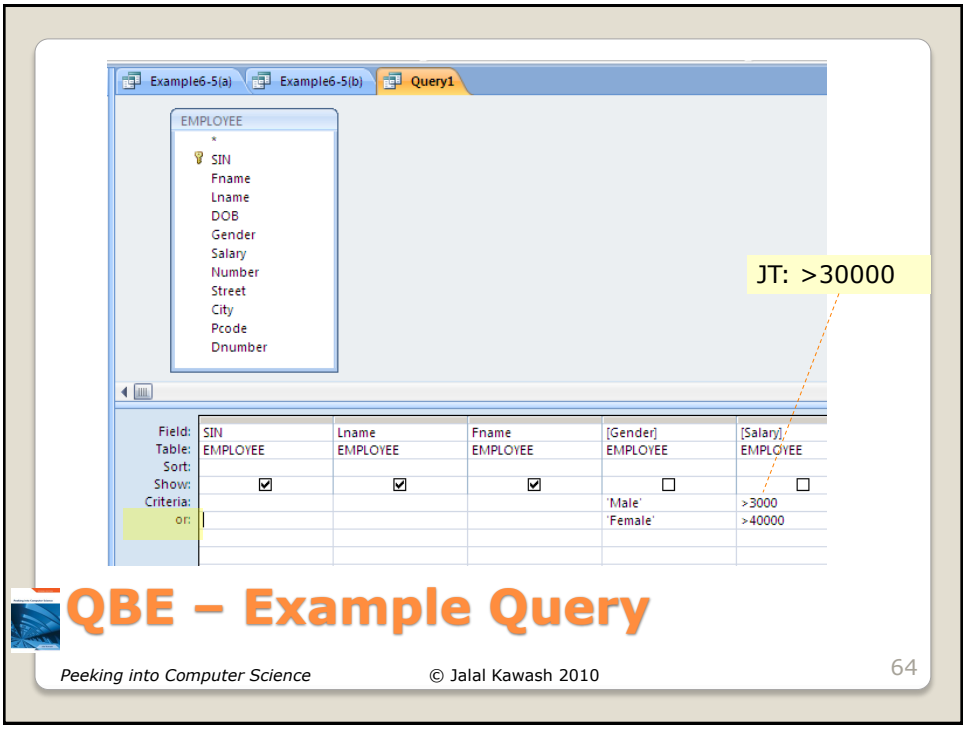
QBE – Example Query

```
SELECT   SIN, Lname, Fname
FROM     EMPLOYEE
WHERE    Salary >= 30000
         AND Salary <= 50000
```

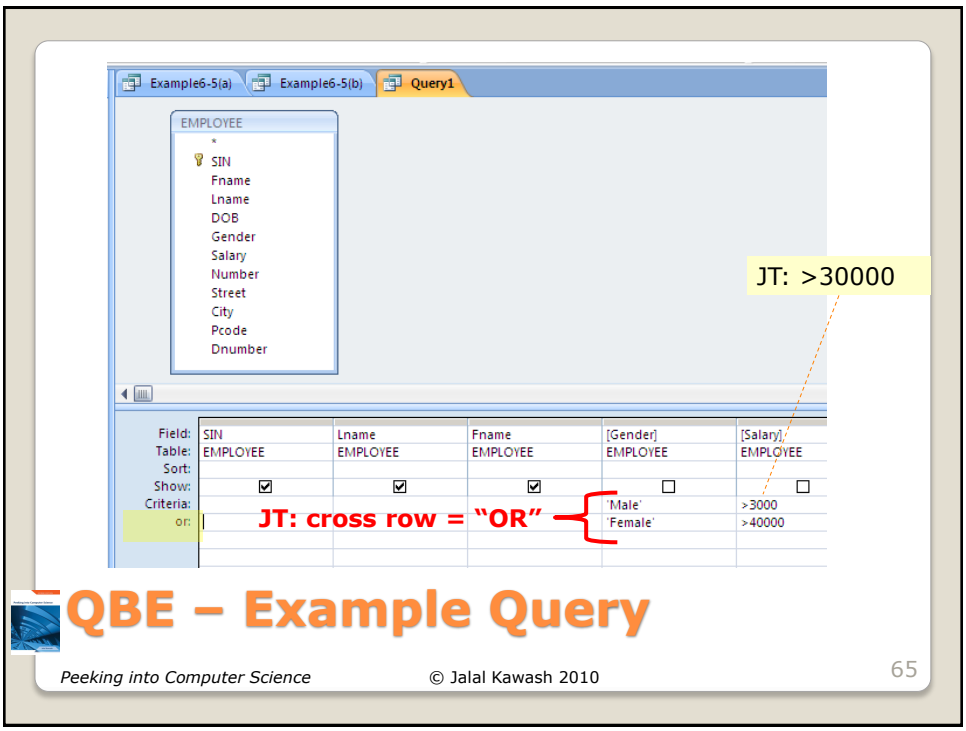
SQL – Example Query


```
SELECT   SIN, Lname, Fname
FROM     EMPLOYEE
WHERE    ( Gender = 'Male' AND Salary > 30000 )
         OR ( Gender = 'Female' AND Salary > 40000 )
```

SQL – Example Query



QBE – Example Query



QBE – Example Query

- Take care not to specify queries that can never be true! (Logic: contradiction)
- This will result in an "Empty Query", a query that yields no results.
  - Example: Which employees have a gross pay lower than $1,000 AND higher than $2,000 (inclusive for both) on one of their time cards?

Query

	StartPayPeriod	PayRate	HoursWorked	GrossPay: [PayRat
	TimeBilled	Employees	TimeBilled	
	☑	☑	☑	☑
				<=1000 And >=2000

Result of the (empty) query

Employees with pay les than $1K AND greater than $2K : Select Query

SIN	LastName	FirstName	StartPayPeriod	PayRate	HoursWorked	GrossPay

JT's Extra: Empty Queries (Contradiction)

 © Jalal Kawash 2009

Wav file from "The Simpson" © Fox

- In a similar fashion take care not to specify queries that are always true. (Tautology)

Query

Field:	YearsOfService	LastName	FirstName
Table:	Employees	Employees	Employees
Sort:			
Show:	☑	☑	☑
Criteria:	>=10		
or:	<=20		

SQL

```
SELECT Employees.YearsOfService,
Employees.LastName, Employees.FirstName
FROM Employees
WHERE (((Employees.YearsOfService)>=10))
OR (((Employees.YearsOfService)<=20));
```

Query result

YearsOfServ	LastName	FirstName
8	Tam	James
2	Morris	Heather
10	Mason	Harry
2	Cartman	Eric
1	Griffin	Stewie
15	Pike	Christopher
25	Lee	Bruce
12	Long	Fei
7	Akash	Akabar
20	Linnear	Nicolas

JT's Extra: Queries Resulting From A Tautology

 © Jalal Kawash 2009