Chapter 3: Specifics on The Relational Model

CSCI 475

SQL Basics

Help

- http://www.w3schools.com/sql/

Data types

- http://www.w3schools.com/sql/sql_datatypes.asp
- varchar(n)
- char(n)
- int
- double
- date yyyy-mm-dd

Table name: EMPLOYEE

Employee_ID	Employee_FName	Employee_LName	Employee_HireDate	Employee_Title
02345	Johnny	Jones	2/14/1995	DBA
03373	Franklin	Johnson	3/15/2002	Purchasing Agent
04893	Patricia	Richards	6/11/2004	DBA
06234	Jasmine	Patel	8/10/2005	Programmer
08273	Marco	Bienz	7/28/2006	Analyst
09002	Ben	Joiner	5/20/2010	Clerk
09283	Juan	Chavez	7/4/2010	Clerk
09382	Jessica	Johnson	8/2/2010	Database Programmer
10282	Amanda	Richardson	4/11/2011	Clerk
13383	Raymond	Matthews	3/12/2012	Programmer
13567	Robert	Almond	9/30/2012	Analyst
13932	Megan	Lee	9/29/2013	Programmer
14311	Lee	Duong	9/1/2014	Programmer

Table name: CERTIFIED

Employee_ID	Skill_ID	Certified_Date
02345	100	2/14/2000
02345	110	8/9/2001
02345	180	2/14/2003
03373	120	6/20/2009
04893	180	6/11/2004
04893	220	9/20/2010
06234	110	8/10/2005
06234	200	8/10/2005
06234	210	1/29/2010
08273	110	3/8/2007
08273	190	8/19/2010
09002	110	5/16/2011
09002	120	5/16/2011
09382	140	8/2/2010
09382	210	8/2/2010
09382	220	5/1/2011
13383	170	3/12/2012
13567	130	9/30/2012
13567	140	5/23/2013
14311	110	9/1/2014

Table name: SKILL

Skill_ID	Skill_Name	Skill_Description
100	Basic Database Management	Create and manage database user accounts.
110	Basic Web Design	Create and maintain HTML and CSS documents.
120	Advanced Spreadsheets	Use of advanced functions, user-defined functions, and macroing.
130	Basic Process Modeling	Create core business process models using standard libraries.
140	Basic Database Design	Create simple data models.
150	Master Database Programming	Create integrated trigger and procedure packages for a distributed environment.
160	Basic Spreadsheets	Create single tab worksheets with basic formulas
170	Basic C# Programming	Create single-tier data aware modules.
180	Advanced Database Management	Manage Database Server Clusters.
190	Advance Process Modeling	Evaluate and Redesign cross-functional internal and external business processes.
200	Advanced C# Programming	Create multi-tier applications using multi-threading
210	Basic Database Manipulation	Create simple data retrieval and manipulation statements in SQL.
220	Advanced Database Manipulation	Use of advanced data manipulation methods for multi-table inserts, set operations, and correlated subqueries.

Creating Tables in SQL

```
CREATE TABLE EMPLOYEE (
EMPLOYEE_ID     VARCHAR(6) PRIMARY KEY,
EMPLOYEE_FNAME  VARCHAR(20),
EMPLOYEE_LNAME  VARCHAR(20),
EMPLOYEE_HIREDATE DATE,
EMPLOYEE_TITLE  VARCHAR(30)
);

CREATE TABLE SKILL (
SKILL_ID   INT AUTO_INCREMENT PRIMARY KEY,
SKILL_NAME VARCHAR(75),
SKILL_DESCRIPTION VARCHAR(200)
);
```

Creating a Table with 2 Primary Keys

```
CREATE TABLE CERTIFIED (
EMPLOYEE_ID   VARCHAR(6) REFERENCES EMPLOYEE,
SKILL_ID      INTEGER REFERENCES SKILL,
CERTIFIED_DATE DATE,
CONSTRAINT PRIMARY KEY (EMPLOYEE_ID, SKILL_ID)
);
```

Populating a Database

- Insert all field values
- **INSERT INTO EMPLOYEE VALUES ('02345', 'Johnny', 'Jones', '02/14/1995', 'DBA');**

- Customize which field values to insert
- **INSERT INTO SKILL (SKILL_NAME, SKILL_DESCRIPTION) VALUES ('Basic Database Managment', 'Create and manage database user accounts.');**

Relational Algebra

Manipulating data in a relational model

SELECT

JOIN

INTERSECT (tables must have same fields)

UNION (tables must have same fields)

DIFFERENCE*

PRODUCT* (Cartesian product – rows of one table times row of second table)

Relational Algebra - Completeness & Closure

Completeness – degree to which Relational Algebra is supported

- Minimally Relational if SELECT & JOIN supported

Closure – Relational Algebra produces new tables

- SELECT is a closed operation

Relational Algebra - Examples

Tickets

TICKETNUM	OFFICER_ID	DLNUM	LOCATION	SPEED
1	234	800123	S Lamar	45
2	456	800456	W Jackson	55
3	234	800789	S Lamar	42
4	234	800111	S Lamar	52

Officers

OFFICE_ID	Name
456	Roy Smith
234	John Applegate

Relational Algebra - Select Examples

SELECT * FROM Tickets WHERE speed > 45;

SELECT Location, OfficerID from Tickets;

SELECT Location, OfficerID, FROM Tickets WHERE speed > 45;

Relational Algebra - Union Example

Union requires identical table fields (attributes/columns)

Tickets_January

TICKETNUM	OFFICER_ID	DLNUM	LOCATION	SPEED
1	234	800123	S Lamar	45
2	456	800456	W Jackson	55
3	234	800789	S Lamar	42
4	234	800111	S Lamar	52

Tickets_February

TICKETNUM	OFFICER_ID	DLNUM	LOCATION	SPEED
1	798	800223	S Lamar	60
2	456	800455	W Jackson	57

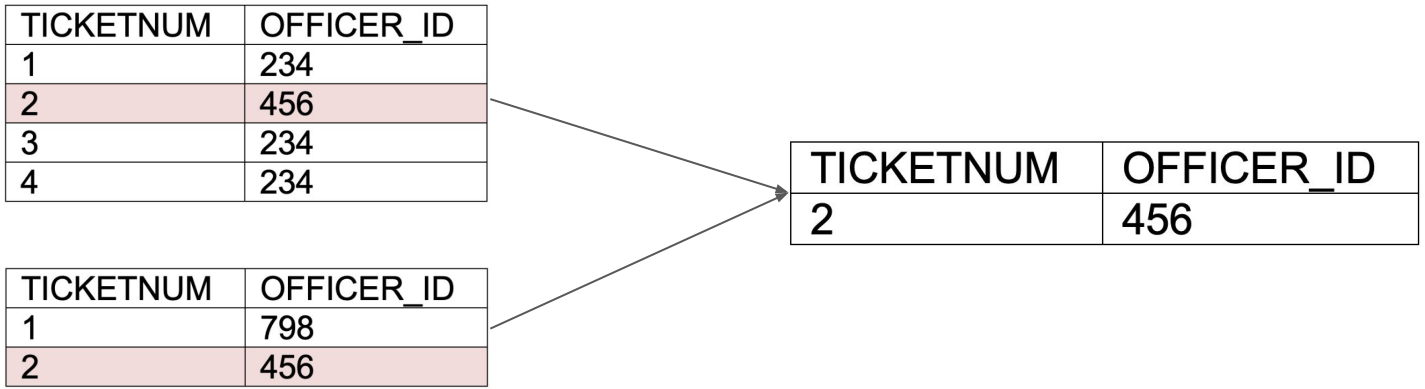
Relational Algebra - Union Example - Result

January ∪ February

TICKETNUM	OFFICER_ID	DLNUM	LOCATION	SPEED
1	234	800123	S Lamar	45
2	456	800456	W Jackson	55
3	234	800789	S Lamar	42
4	234	800111	S Lamar	52
1	798	800223	S Lamar	60
2	456	800455	W Jackson	57

Relational Algebra - Intersect Example

Requires similar fields and only looks for those in common.



Relational Algebra - Difference Example

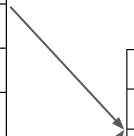
Similar to INTERSECT. Returns the ones that are different.

Tickets_January

TICKETNUM	OFFICER_ID	DLNUM	LOCATION	SPEED
1	234	800123	S Lamar	45
2	456	800456	W Jackson	55
3	234	800789	S Lamar	42
4	234	800111	S Lamar	52

Tickets_February

TICKETNUM	OFFICER_ID	DLNUM	LOCATION	SPEED
1	798	800223	S Lamar	60
2	456	800455	W Jackson	57



TICKETNUM	OFFICER_ID
1	234
3	234
4	234

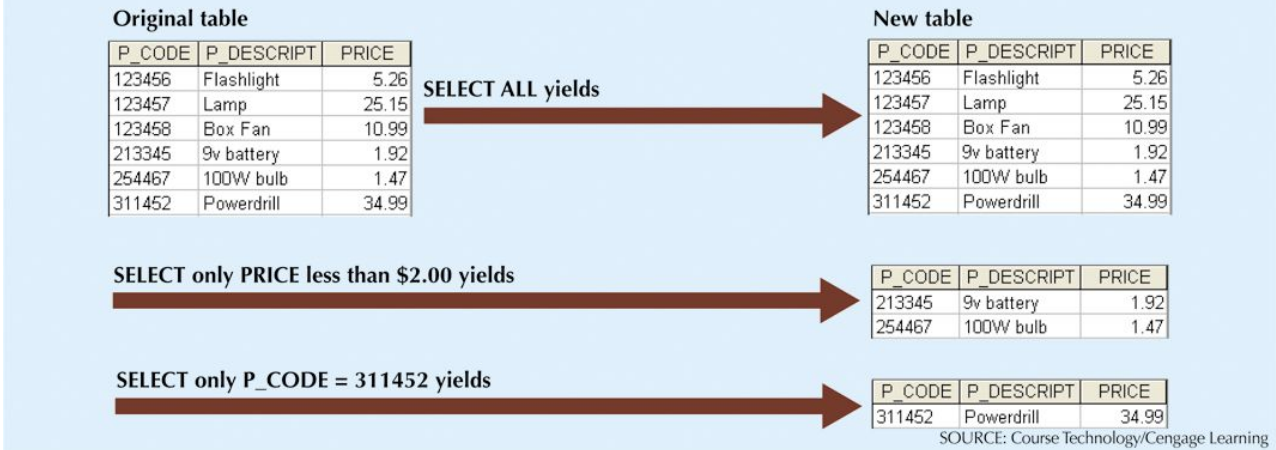
Relational Algebra - Product Example

Cartesian Product – all fields, all rows

Tickets x Officers

TICKETNUM	OFFICER_ID	DLNUM	LOCATION	SPEED	OFFICE_ID	Name
1	234	800123	S Lamar	45	456	Roy Smith
1	234	800123		45	234	John Applegate
2	456	800456	W Jackson	55	456	Roy Smith
2	456	800456		55	234	John Applegate
3	234	800789	S Lamar	42	456	Roy Smith
3	234	800789		42	234	John Applegate
4	234	800111	S Lamar	52	456	Roy Smith
4	234	800111		52	234	John Applegate

FIGURE 3.4 **SELECT**



SOURCE: Course Technology/Cengage Learning

FIGURE 3.6 UNION

P_CODE	P_DESCRIPT	PRICE
123456	Flashlight	5.26
123457	Lamp	25.15
123458	Box Fan	10.99
213345	9v battery	1.92
254467	100W bulb	1.47
311452	Powerdrill	34.99

UNION

P_CODE	P_DESCRIPT	PRICE
345678	Microwave	160.00
345679	Dishwasher	500.00
123458	Box Fan	10.99

yields

P_CODE	P_DESCRIPT	PRICE
123456	Flashlight	5.26
123457	Lamp	25.15
123458	Box Fan	10.99
213345	9v battery	1.92
254467	100W bulb	1.47
311452	Powerdrill	34.99
345678	Microwave	160
345679	Dishwasher	500

SOURCE: Course Technology/Cengage Learning

FIGURE 3.7 INTERSECT

STU_FNAME	STU_LNAME
George	Jones
Jane	Smith
Peter	Robinson
Franklin	Johnson
Martin	Lopez

INTERSECT

EMP_FNAME	EMP_LNAME
Franklin	Lopez
William	Turner
Franklin	Johnson
Susan	Rogers

yields

STU_FNAME	STU_LNAME
Franklin	Johnson

SOURCE: Course Technology/Cengage Learning

FIGURE 3.8 DIFFERENCE

STU_FNAME	STU_LNAME
George	Jones
Jane	Smith
Peter	Robinson
Franklin	Johnson
Martin	Lopez

DIFFERENCE

EMP_FNAME	EMP_LNAME
Franklin	Lopez
William	Turner
Franklin	Johnson
Susan	Rogers

yields

STU_FNAME	STU_LNAME
George	Jones
Jane	Smith
Peter	Robinson
Martin	Lopez

SOURCE: Course Technology/Cengage Learning

FIGURE 3.9 PRODUCT

P_CODE	P_DESCRIPT	PRICE
123456	Flashlight	5.26
123457	Lamp	25.15
123458	Box Fan	10.99
213345	9v battery	1.92
254467	100W bulb	1.47
311452	Powerdrill	34.99

PRODUCT

STORE	AISLE	SHELF
23	W	5
24	K	9
25	Z	6

yields

P_CODE	P_DESCRIPT	PRICE	STORE	AISLE	SHELF
123456	Flashlight	5.26	23	W	5
123456	Flashlight	5.26	24	K	9
123456	Flashlight	5.26	25	Z	6
123457	Lamp	25.15	23	W	5
123457	Lamp	25.15	24	K	9
123457	Lamp	25.15	25	Z	6
123458	Box Fan	10.99	23	W	5
123458	Box Fan	10.99	24	K	9
123458	Box Fan	10.99	25	Z	6
213345	9v battery	1.92	23	W	5
213345	9v battery	1.92	24	K	9
213345	9v battery	1.92	25	Z	6
311452	Powerdrill	34.99	23	W	5
311452	Powerdrill	34.99	24	K	9
311452	Powerdrill	34.99	25	Z	6
254467	100W bulb	1.47	23	W	5
254467	100W bulb	1.47	24	K	9
254467	100W bulb	1.47	25	Z	6

SOURCE: Course Technology/Cengage Learning