



ITC 556/114
WEEK 8

Introduction to Structured Query Language (SQL)

Continuation from Chapter 7

Tutorial on MySQL:

<http://youtu.be/DCgRF4KOYIY>

SQL REVIEW

- List the DDL commands in SQL

SQL Data Definition Commands

COMMAND OR OPTION	DESCRIPTION
CREATE SCHEMA AUTHORIZATION	Creates a database schema
CREATE TABLE	Creates a new table in the user's database schema
NOT NULL	Constraint that ensures that a column will not have null values
UNIQUE	Constraint that ensures that a column will not have duplicate values
PRIMARY KEY	Defines a primary key for a table
FOREIGN KEY	Defines a foreign key for a table
DEFAULT	Defines a default value for a column (when no value is given)
CHECK	Constraint used to validate data in a column
CREATE INDEX	Creates an index for a table
CREATE VIEW	Creates a dynamic subset of rows/columns from one or more tables
ALTER TABLE	Modifies a table's definition (adds, modifies, or deletes attributes or constraints)
CREATE TABLE AS	Creates a new table based on a query in the user's database schema
DROP TABLE	Permanently deletes a table (and thus its data)
DROP INDEX	Permanently deletes an index
DROP VIEW	Permanently deletes a view

What are views?

- **View** is virtual table based on SELECT query
- Create view by using CREATE VIEW command
- Special characteristics of relational view:
 - Name of view can be used anywhere a table name is expected
 - View dynamically updated(always shows up-to-date data) every time a user queries a view.
 - Restricts users to only specified columns and rows
 - Views may be used as basis for reports

```
CREATE VIEW  
view_name  
SELECT column_name(s)  
FROM table_name  
WHERE condition
```

Data Manipulation Commands

COMMAND OR OPTION	DESCRIPTION
INSERT	Inserts row(s) into a table
SELECT	Selects attributes from rows in one or more tables or views
WHERE	Restricts the selection of rows based on a conditional expression
GROUP BY	Groups the selected rows based on one or more attributes
HAVING	Restricts the selection of grouped rows based on a condition
ORDER BY	Orders the selected rows

Table 7.6 - Comparison Operators

SYMBOL	MEANING
=	Equal to
<	Less than
<=	Less than or equal to
>	Greater than
>=	Greater than or equal to
<> or !=	Not equal to

Arithmetic operators

- **The Rule of Precedence:** Establish the order in which computations are completed
- Perform:
 - Operations within parentheses
 - Power operations (1st priority by default)
 - Multiplications and divisions (2nd priority by default)
 - Additions and subtractions (3rd priority by default)

Comparison Operators: Computed Columns and Column Aliases

- SQL accepts any valid expressions/formulas in the computed columns
- **Alias:** Alternate name given to a column or table in any SQL statement to improve the readability
- Computed column, an alias, and date arithmetic can be used in a single query

Listing Table Rows

- SELECT
 - Used to list contents of table
- Syntax
 - `SELECT columnlist`
`FROM tablename`
- Columnlist represents one or more attributes, separated by commas
- Asterisk can be used as wildcard character to list all attributes

Listing Table Rows (continued)

FIGURE 7.3 The contents of the PRODUCT table

	P_CODE	P_DESCRIPT	P_INDATE	P_QOH	P_MIN	P_PRICE	P_DISCOUNT	V_CODE
▶ +	11QER/31	Power painter, 15 psi., 3-nozzle	03-Nov-05	8	5	109.99	0.00	25595
+	13-Q2/P2	7.25-in. pwr. saw blade	13-Dec-05	32	15	14.99	0.05	21344
+	14-Q1/L3	9.00-in. pwr. saw blade	13-Nov-05	18	12	17.49	0.00	21344
+	1546-QQ2	Hrd. cloth, 1/4-in., 2x50	15-Jan-06	15	8	39.95	0.00	23119
+	1558-QWV1	Hrd. cloth, 1/2-in., 3x50	15-Jan-06	23	5	43.99	0.00	23119
+	2232/QTY	B&D jigsaw, 12-in. blade	30-Dec-05	8	5	109.92	0.05	24288
+	2232/QWVE	B&D jigsaw, 8-in. blade	24-Dec-05	6	5	99.87	0.05	24288
+	2238/QPD	B&D cordless drill, 1/2-in.	20-Jan-06	12	5	38.95	0.05	25595
+	23109-HB	Claw hammer	20-Jan-06	23	10	9.95	0.10	21225
+	23114-AA	Sledge hammer, 12 lb.	02-Jan-06	8	5	14.40	0.05	
+	54778-2T	Rat-tail file, 1/8-in. fine	15-Dec-05	43	20	4.99	0.00	21344
+	89-WRE-Q	Hicut chain saw, 16 in.	07-Feb-06	11	5	256.99	0.05	24288
+	PVC23DRT	PVC pipe, 3.5-in., 8-ft.	20-Feb-06	188	75	5.87	0.00	
+	SM-18277	1.25-in. metal screw, 25	01-Mar-06	172	75	6.99	0.00	21225
+	SWV-23116	2.5-in. wd. screw, 50	24-Feb-06	237	100	8.45	0.00	21231
+	WR3/TT3	Steel matting, 4'x8'x1/8", .5" mesh	17-Jan-06	18	5	119.95	0.10	25595

Selecting Rows with Conditional Restrictions

- Select partial table contents by placing restrictions on rows to be included in output
 - Add conditional restrictions to SELECT statement, using WHERE clause
- Syntax:
 - SELECT *columnlist*
FROM *tablelist*
[WHERE *conditionlist*] ;

Selecting Rows with Conditional Restrictions (continued)

FIGURE 7.4

Selected PRODUCT table attributes for vendor code 21344

	P_DESCRIPT	P_INDATE	P_PRICE	V_CODE
▶	7.25-in. pwr. saw blade	13-Dec-05	14.99	21344
	9.00-in. pwr. saw blade	13-Nov-05	17.49	21344
	Rat-tail file, 1/8-in. fine	15-Dec-05	4.99	21344

SQL SELECT statement

- An SQL SELECT statement can include a variety of operators and functions:

Operators : *, -, /, +, <, >, <>

Functions: Sum, Count, Average, Maximum, Minimum

We will look at these little more next week

Wildcard characters

You also can order a list in a particular order or sort in Ascending/descending order:

Example:

```
SELECT      *  
FROM      hire  
ORDER      BY cust_num;
```

Note: the ***** in this case is a wild character, same as the ' ' Or the **%**. These can represent 1 or more characters and can be used in queries.

The underscore **_** is for one single missing character while the **%** is for more unlimited unknown characters.

More Examples

```
SELECT *  
FROM emp;
```

```
SELECT *  
FROM emp  
WHERE emp_name LIKE 'R%';
```

```
SELECT emp_name, emp_ID, emp_mobile, emp_DOB  
  
FROM emp
```

```
WHERE emp_DOB <1/1/1990;
```

```
SELECT *  
FROM customer  
WHERE name LIKE '%i%';
```

```
SELECT *  
FROM customer  
WHERE name LIKE '%e%';
```

```
SELECT emp_name , emp_ID, emp_mobile,  
emp_area
```

```
FROM emp
```

```
WHERE emp_area <> '2000';
```

NULL and NOT NULL

- The is null/is not null can be used to execute a query result where a value either exists or does not exist –
- The NULL would be the optional entry (such as the zero symbol in your ERD)

Listing only records that meet a certain condition:

Example:

```
SELECT      *  
FROM        hire  
WHERE       p_code= '21344-AA'  
ORDER BY    cust_num;
```

```
SELECT *  
FROM    PRODUCT  
WHERE   V-Code IS NULL;
```

Did you
get this
right??

Another example – DVD hire:

```
SELECT *  
FROM hire  
WHERE date_ret IS NOT NULL;
```

Condition

BETWEEN, IN

```
SELECT      *  
FROM        customer  
WHERE       name BETWEEN 50 AND 100;
```

IN is used to check whether an attribute value matches any value within a value list – Example:

```
SELECT      *  
FROM        customer  
WHERE       cust_num IN ('1001' , '1002' , '1003','1005');
```

More conditions: 'AND' and 'OR'

Choosing the rows to display using more complex conditions

Any number of conditions can be combined using the words 'AND' and 'OR'.

```
SELECT *  
FROM hire  
WHERE date_ret IS NULL  
      AND cust_num = '1001'  
      AND cust_name LIKE 'A%';
```

```
SELECT *  
FROM hire  
WHERE date_ret IS NULL  
      OR cust_num = '1001';
```

Figure 7.12 - Selected PRODUCT Table Attributes: The logical OR

```
SELECT P_DESCRIPT,P_INDATE,P_PRICE, V_CODE  
FROM PRODUCT  
WHERE V_CODE=21344 OR V_CODE=24288
```

P_DESCRIPT	P_INDATE	P_PRICE	V_CODE
7.25-in. pwr. saw blade	13-Dec-13	14.99	21344
9.00-in. pwr. saw blade	13-Nov-13	17.49	21344
B&D jigsaw, 12-in. blade	30-Dec-13	109.92	24288
B&D jigsaw, 8-in. blade	24-Dec-13	99.87	24288
Rat-tail file, 1/8-in. fine	15-Dec-13	4.99	21344
Hicut chain saw, 16 in.	07-Feb-14	256.99	24288

Figure 7.13 - Selected PRODUCT Table Attributes: The Logical AND

```
SELECT P_DESCRIPT,P_INDATE,P_PRICE, V_CODE
FROM PRODUCT
WHERE P_PRICE<50
AND P_INDATE>'15-Jan-2010';
```

P_DESCRIPT	P_INDATE	P_PRICE	V_CODE
B&D cordless drill, 1/2-in.	20-Jan-14	38.95	25595
Claw hammer	20-Jan-14	9.95	21225
PVC pipe, 3.5-in., 8-ft	20-Feb-14	5.87	
1.25-in. metal screw, 25	01-Mar-14	6.99	21225
2.5-in. w/d. screw, 50	24-Feb-14	8.45	21231

Figure 7.14 - Selected PRODUCT Table Attributes: The Logical AND and OR

```
SELECT P_DESCRIPT,P_INDATE,P_PRICE, V_CODE
FROM PRODUCT
WHERE (P_PRICE<50 AND P_INDATE>'15-Jan-2010')
OR V_CODE=24288;
```

P_DESCRIPT	P_INDATE	P_PRICE	V_CODE
B&D jigsaw, 12-in. blade	30-Dec-13	109.92	24288
B&D jigsaw, 8-in. blade	24-Dec-13	99.87	24288
B&D cordless drill, 1/2-in.	20-Jan-14	38.95	25595
Claw hammer	20-Jan-14	9.95	21225
Hicut chain saw, 16 in.	07-Feb-14	256.99	24288
PVC pipe, 3.5-in., 8-ft	20-Feb-14	5.87	
1.25-in. metal screw, 25	01-Mar-14	6.99	21225
2.5-in. wtd. screw, 50	24-Feb-14	8.45	21231

More on conditions

```
SELECT      *  
FROM        hire  
WHERE (cust_num = '1001' OR cust_num = '1003')  
AND (stock_num = '2001' OR stock_num = '2005');
```


More on conditions

What does the following return?

```
SELECT    V_NAME , V_CONTACT, V_PHONE  
FROM      VENDOR  
WHERE     UPPER (V_CONTACT) LIKE 'SMITH%';
```

Joining Database Tables

- Joining tables is the most important distinction between relational database and other DBs
 - Join is performed when data are retrieved from more than one table at a time
 - Equality comparison between foreign key and primary key of related tables
 - Join tables by listing tables in FROM clause of SELECT statement
 - DBMS creates Cartesian product of every table
- PS: A Cartesian join is when you join every row of one table to every row of another table. You can also get one by joining every row of a table to every row of itself (Recursive Relationships)

Outer Joins

- Two types of outer join
 - Left outer join
 - Right outer join

Check out this website and learn more about joins:

http://w3schools.com/sql/sql_join.asp

Go to http://w3schools.com/sql/sql_join.asp

And click on the **Try it yourself** option and experiment by adding some conditions



1. Change from inner join to using where and see what results you will get
2. Add a date criteria for example to see only the orders on 9/18/1996

Outer Joins

- Returns rows matching the join condition
- Also returns rows with unmatched attribute values for tables to be joined
- Three types
 - Left
 - Right
 - Full
- Left and right designate order in which tables are processed

Outer Joins (continued)

- Left outer join
 - Returns rows matching the join condition
 - Returns rows in left side table with unmatched values
 - Syntax: `SELECT column-list FROM table1 LEFT [OUTER] JOIN table2 ON join-condition`
- Right outer join
 - Returns rows matching join condition
 - Returns rows in right side table with unmatched values

Outer Joins (continued)

- Full outer join
 - Returns rows matching join condition
 - Returns all rows with unmatched values in either side table
 - Syntax:

```
SELECT      column-list
FROM  table1 FULL [OUTER] JOIN table2
      ON join-condition
```

Order by

- Sorts the records in ascending order by default. To sort the records in a descending order, you can use the DESC keyword.

syntax:

```
SELECT column_name,column_name  
FROM table_name  
ORDER BY column_name,column_name  
ASC|DESC;
```

Summary

- SQL commands can be divided into two overall categories:
 - Data definition language commands
 - Data manipulation language commands
- Basic data definition commands allow you to create tables, indexes, and views
- Many SQL constraints can be used with columns

Using MySQL Workbench – View Video

http://youtu.be/X_umYKqKaFo

<http://youtu.be/9AZJ4aS56dE>

Annotated by: Antoinette Cevenini

Section-1

All the following problems use the Thirlemere Thoroughbreds (TT) database. Before you answer any of these, you will need to load that database.

- 1.1 List the name and address of each show.
- 1.2 List the name of each horse that was born in 1990.
- 1.3 List the name and address of each show that has '13' as part of its address.
- 1.4 List the name, year of birth and colour for each mare (mares have gender of 'M') that is either grey or white.
- 1.5 List the name and year of death of each dead horse. Sort the results into descending order on year of death.
- 1.6 List the name of each type of event. Do not list duplicate event names.
- 1.7 List the event_code of each event which has a first prize of between 100 and 200 dollars.
- 1.8 List the names of all horses whose colour is "white".
- 1.9 List the horse_id and name of every horse that is still alive and was born before 1990. Order by horse_id.
- 1.10 List the name and year of birth of each horse that is now dead, and was born in either 1976, 1978, 1980 or 1981.

Tutorial Activities SQL

Assignment 3

Assignment 3 - Online Quiz 2

Note:

- This assignment will be opened in the last 30 minutes of the class time
- You need to be in class to do this quiz

Assignment 3 will cover information from weeks 1-6 inclusive – Mainly on modelling, Normalisation and design concepts

Today's Tutorial Questions

1. Please attempt all the questions in Tutorial 8 – that is **set 1** in creating SQL queries based on tt2000 database.
2. Refer to the handout under week 8 resources on how to import the database.

*If time allows attempt these review questions from chapter 7 in your textbook:

Questions 1, 2, 5, 9, 15

Assignment 2

Discuss Assignment 2 requirements

SQL Activities - set 1

Answer to Q 1

```
SELECT show_name, show_address  
FROM show1;
```