

Chapter 3

Where & Order By Clauses

SQL Developer Features

SQL*Plus Commands

The WHERE Clause

```
SELECT      [DISTINCT] {*| column [alias], ...}  
FROM        table  
[WHERE      condition(s)];
```

- A condition that restricts which rows participate in a SQL statement
 - a condition will be evaluated as either TRUE, FALSE, or UNKNOWN
 - a row will be selected when the condition evaluates as TRUE
- Conditions are built with a comparison operator and a value
- A condition must compare two values of the same datatype

Comparison Operators

Operator	Meaning
=	Equal to
>	Greater than
>=	Greater than or equal to
<	Less than
<=	Less than or equal to
<>, !=	Not equal to

3

WHERE Clause Practice

- Show the title, issue and length of articles shorter than 1500 words.
- Show the title, issue and length of the business articles.
- Show the last name, first name and phone number for the freelancer writers.

4

More Comparison Operators

Operator	Meaning
BETWEEN ...AND...	Between two values (inclusive)
IN(list)	Match any of a list of values
LIKE	Match a character pattern
IS NULL	Is a null value

5

The BETWEEN Comparison Operator

- An alternative way to select rows based on a range of values

```
SELECT title, length
FROM article
WHERE length BETWEEN 500 AND 1500;
```

- Equivalent to

```
SELECT title, length
FROM article
WHERE _____;
```

6

The IN Comparison Operator

- Selects rows based on items in a predefined list

```
SELECT *  
FROM article  
WHERE type IN ('BUS', 'LAW');
```

- Equivalent to

```
SELECT *  
FROM article  
WHERE _____;
```

7

The LIKE Operator

- Selects rows based on a pattern you specify
 - pattern provides **flexibility** when exact value is not known
- Use the LIKE keyword and wildcard symbol(s)
 - **%** —

```
SELECT title, type  
FROM article  
WHERE title LIKE '%$%';
```

- **Practice Time**

- Show name and phone of all writers
- Show name and phone of writers in area code 710

8

Evaluating Null Values

- NULL represents a missing or unknown value
- IS NULL operator

- used to test for null values

```
SELECT ln, fn, phone
FROM writer
WHERE phone IS NULL;
```

- don't try compare a value to NULL

```
SELECT ln, fn, phone
FROM writer
WHERE phone = NULL;
```

- Practice Time

- Which writers don't have a contact?
- Which articles don't have a writerid?

9

Logical Operators

Operator	Meaning
NOT	Returns TRUE if the following condition is FALSE
AND	Returns TRUE if <i>both</i> conditions are TRUE
OR	Returns TRUE if <i>either</i> condition is TRUE

- Used when constructing a query containing multiple criteria
- Shown in order of precedence
 - may need parentheses to override logical operator precedence
 - $3 + 4 * 2$

10

Logical Operators: Examples

```
SELECT title, type, length
FROM article
WHERE type = 'BUS' AND length <1500;
```

```
SELECT title, type, length
FROM article
WHERE type = 'BUS' OR length <1500;
```

11

Logical Operators: Precedence

- Which are either political or business articles that are shorter than 1500 words?

9
rows

```
SELECT title, type, length
FROM article
WHERE type='POL' OR type='BUS' AND length<1500;
```

7
rows

```
SELECT title, type, length
FROM article
WHERE (type='POL' OR type='BUS') AND length<1500;
```

12

Logic Tables

AND	TRUE	FALSE	NULL	OR	TRUE	FALSE	NULL	NOT	
TRUE	TRUE	FALSE	NULL	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE
FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	NULL	FALSE	TRUE
NULL	NULL	FALSE	NULL	NULL	TRUE	NULL	NULL	NULL	NULL

13

Lab 3.2

The ORDER BY Clause

```
SELECT      [DISTINCT] {*| column [alias], ...}
FROM        table
[WHERE      condition(s)]
[ORDER BY   column [{ASC|DESC}] 1, ...];
```

- Used to **sort** the returned rows
 - can sort ascending or descending
 - can sort by multiple columns

```
SELECT ln, fn, freelancer
FROM writer
ORDER BY freelancer;
```

```
SELECT ln, fn, freelancer
FROM writer
ORDER BY freelancer, 1, fn;
```

- can cite a column **alias**
- can cite a column **position (number)**

14

ORDER BY and NULLS

- Can use **NULLS FIRST** or **NULLS LAST** clause to control placement of nulls

```
SELECT ln, phone
FROM writer
ORDER BY phone;
```

LN	PHONE
Cox	(210) 783-5415
Johnson	(210) 895-2046
Martinez	(544) 332-7788
Seeger	(576) 423-0932
...	
Cohen	(910) 338-1875
Waldeck	(917) 361-8181
Kim	(917) 729-5364
Nilsson	
Lawton	

```
SELECT ln, phone
FROM writer
ORDER BY phone NULLS FIRST;
```

LN	PHONE
Nilsson	
Lawton	
Cox	(210) 783-5415
Johnson	(210) 895-2046
Martinez	(544) 332-7788
Seeger	(576) 423-0932
...	
Cohen	(910) 338-1875
Waldeck	(917) 361-8181
Kim	(917) 729-5364

15

ORDER BY Practice

- Show title, type, issue, length of each article sorted by length, with longest article on top
- Show title, type, issue, length of each article sorted by type and secondarily, sorted by date with most recent on top
 - note that a **chronological** sort is performed

16

DISTINCT and ORDER BY

- In a SELECT DISTINCT query with an ORDER BY clause all ORDER BY items are limited to constants and SELECT list expressions

```
SELECT DISTINCT type, writerid
FROM article
ORDER BY type, writerid;
```

```
SELECT DISTINCT type, writerid
FROM article
ORDER BY length;
```

17

Column Alias

- By default, a field's name is used as its column heading
- Use an alias to specify a different column heading
- Multi-word aliases must be enclosed in double-quotes
- Practice Time

```
SELECT ln, phone FROM writer;

SELECT ln, phone AS Phone# FROM writer;

SELECT ln, phone Phone# FROM writer;

SELECT ln, phone "Phone Number"
FROM writer;
```

18

Comments in SQL Statements

- Notes used to document a script
- Are non-executable
 - can use to temporarily disable statements
- Two Styles
 - single-line
 - --
 - multi-line
 - /* */

```

cis119do x
Run Script (F5)
SET ECHO on
--Part 1
select * from cat;
--Part 2
select object_name, object_type, status from user_objects;
/*--Part 3 need to tweak some more
SELECT user_id, to_char(logon_day, 'mm/dd/yyyy hh:mi am') l
FROM sys.stats$user_log
WHERE logon_day >= SYSDATE-1
ORDER BY login;*/
--Part 4
select * from user_constraints;
    
```

Script

- An unformatted text file containing one or more SQL statements and/or SQL*Plus commands
 - defaults to .sql extension
 - define the default script folder
 - mySCC: set to H:\cis119do
 - home: set to C:\cis119do
- Practice Time
 - Launch Windows Explorer
 - Create folder h:\cis119do
 - Use Tools > Preferences to set the default script location

```

assignment1.sql x
SQL Worksheet History
SET ECHO on
SPPOOL h:\cis119do\assignment1.txt
--Part 1
select * from type;
--Part 2
select * from article
where length <1000;
--Part 3
descr type
    
```

Preferences


Environment

- Code Editor
- Compare and Merge
- Database
 - Advanced
 - Autotrace/Explain Plan
 - Drag And Drop
 - NLS
 - ObjectViewer
 - PL/SQL Compiler
 - Reports
 - SQL Editor Code Templ
 - SQL Formatter
 - Third Party JDBC Driver
 - User Defined Extension
 - Worksheet
- Debugger

Database: Worksheet

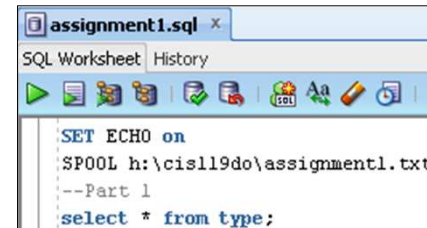
- Autocommit in SQL Worksheet
- Open a Worksheet on connect
- Close all worksheets on disconnect
- Prompt for Save file on close
- Max Rows to print in a script: 5000
- SQL History Limit: 100
- Select default path to look for scripts: H:\cis119do
- This is the directory used when running a
 - Save Bind variables to disk on exit
 - Automatically Freeze Result Tabs

Saving SQL Statements

- Saves worksheet's contents as an unformatted text file
 - i.e., a script
- Running a script
 - **START** command (@)
 - must specify full path to script if not in the default path
 - enclose in quotes if there are spaces in the folder name or filename
- Opening a script 
 - to view/edit/run it

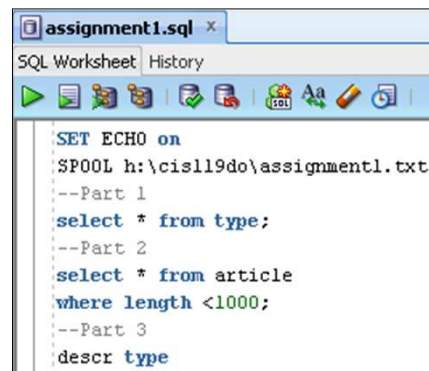


```
start "s:\cis119\scripts\createstudent"
```



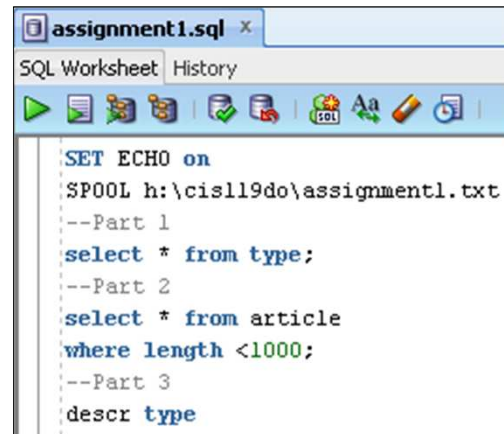
SQL*Plus SPOOL Command

- Creates a text file containing a **script's** output
 - if **ECHO** is on, commands are included with the results
 - we'll use in upcoming Applying SQL assignments
 - default extension is .LST
 - specifying .TXT makes opening easier
 - if a file with same name already exists it is overwritten **without warning**



Practice Time

- Create this script
- Save it as assignment1.sql
- Close the pane
- Open the assignment1.sql script
- Run the script (F5)
- Review the Results pane
 - SQL statements and results
- Open assignment1.txt
 - scan its contents
 - commands and results



```
assignment1.sql x
SQL Worksheet History
SET ECHO on
SPOOL h:\cis119do\assignment1.txt
--Part 1
select * from type;
--Part 2
select * from article
where length <1000;
--Part 3
descr type
```

23

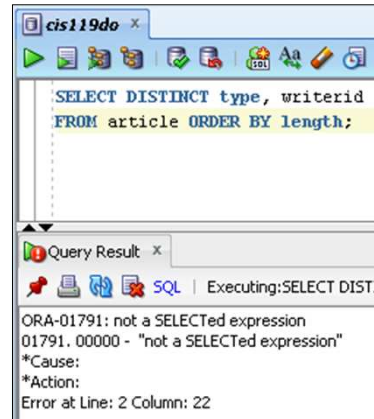
SQL Statements vs. SQL*Plus Commands

- | | |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none">■ SQL Statement<ul style="list-style-type: none">□ Statements manipulate data and objects in the database<ul style="list-style-type: none">■ examples
□ An ANSI standard language□ Keywords cannot be abbreviated□ Results appear in a Query Result pane | <ul style="list-style-type: none">■ SQL*Plus Command<ul style="list-style-type: none">□ Commands to control the client software's appearance and behavior<ul style="list-style-type: none">■ examples
□ Oracle proprietary□ Keywords can be abbreviated□ Results appear in a Statement Output pane |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

24

Understanding Oracle Error Messages

- Most error messages are helpful
 - if you understand the terms and concepts
- Resolve one error at a time
- Double-check
 - statement syntax
 - object names
 - punctuation



25

Online Documentation

- Available at <http://cisdev.sc.maricopa.edu>
- Oracle Documentation Library
 - Books tab
 - Error Messages
 - SQL Reference
 - SQL*Plus User's Guide and Reference
 - HTML and PDF for each
 - Hyperlinks for
 - Index
 - Contents
- Oracle SQL Developer Reference Guide

26

Practice Time

- What does error number **ORA-02292** mean?
- What does the **ECHO** command do? `SET ECHO {ON | OFF}`
- What does **DISTINCT** do when used in **SELECT** statement?

27