Oracle
Version 12c

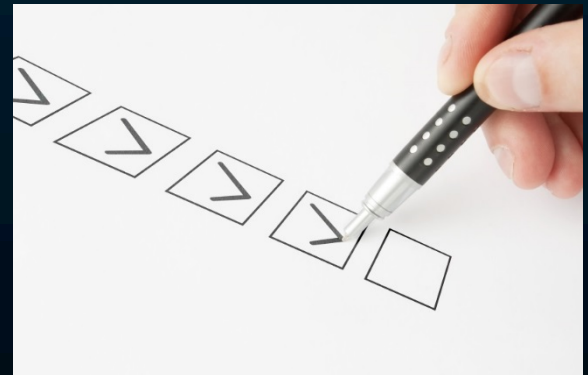**Retrieving Data Using SQL SELECT Statements**

Enabling Objectives

After completing this chapter, in the next 40 minutes you will be able to :

- Write 2 SQL select commands with or without aliasing the table names on Oracle Platform.

- Identify from the given list the commonly used iSQL*Plus commands

Key Topics

- Features of SQL
- Basic SQL Select Command
- iSQL*Plus Commands

Features of SQL

Features of SQL

- A SQL is a non-procedural, fourth generation language

- SQL processes sets of records rather than a single record at a time. The most common form of a set of records is a table

- SQL can be used by a range of users including DBAs, application Programmers and many other types of end users

SQL Command set

SQL COMMANDS	DESCRIPTION
SELECT	Data Retrieval Command
INSERT, UPDATE, DELETE, MERGE	Data Manipulation Language Commands (DML) These statements insert new rows, change existing data and delete unwanted rows.
CREATE, ALTER, DROP	Data Definition Language Commands (DDL) These statements Create, Change or Remove different database objects.
GRANT, REVOKE	Data Control Language Commands (DCL) These statements give or remove access rights to both the Oracle database and the structures within it.
COMMIT, ROLLBACK	Transaction Control Language (TCL) Theses statements are used with transaction to save or undo certain actions.

Rules for writing SQL statements

SQL statements

- Can be broken down into multiple lines

- Command words cannot be split across lines

- Are not case sensitive (unless indicated otherwise)

- May be terminated with a semi-colon (;) or a forward slash (/)

- The forward slash must be given in a new line after the SQL statement

Basic SQL Select Command

Uses of SQL Select statement

- For retrieving data from the database either selectively or collectively

- Column names can be specified or " * " may be specified for displaying all columns

- Calculated columns can be displayed

- Duplication of rows can be removed

- Sorting of rows is possible

- Summary information can be displayed

Basic SQL Statement

SELECT column_list | * FROM table_name

column_list: names of columns whose values need to be retrieved, use " *" if values from all column are needed.

Table_name: name of table being queried.

Examples: EMPLOYEES table

select * from employees;

Result:

Retrieves all the records from the employees table for 3 columns

Basic SQL Statement

Examples:

EMPLOYEES table:

select * ,e_code from employees;

Result: The following query results in a **syntax error**

select employees.e_code, employees.e_salary from employees;

Result:

The query returns employees code and salary.
Column names can be prefixed by table names for clarity in the code, specifically required when retrieving records from multiple tables.

Basic SQL Statement

Examples: EMPLOYEES table:

select e.e_code, e.e_salary from employees e;

Result :
The query returns employees code and salary. The following query exhibits table aliasing. Alias names can be used in the select list.
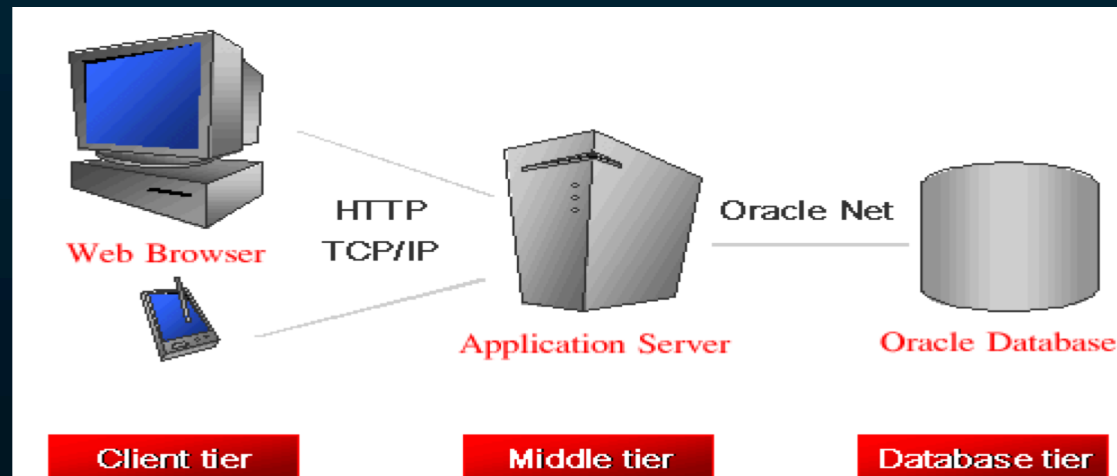
select employees.*, e_code from employees;

Result: The following query returns all the fields from the employees table including e_code twice in the result.

iSQL* Plus Commands

iSQL* Plus

- Is an interactive and batch query tool on a client computer.

- It allows you to execute SQL statements against Oracle database.

iSQL* Plus Commands

Listed below are some of the commonly used iSQL*Plus commands

iSQL*Plus Commands	Description
conn[ect] username/password For example: _connect scott/tiger_	Connects the user to the Oracle server
_set_{parameter} For example : set numwidth 5	Allows you to set many parameter and options for your session.
_show_{parameter} For example: show numwidth	Echos the value of the parameter specified on the screen
desc[ribe] schema object For example: _desc employees_	List the column definition for the specified schema object

iSQL* Plus Commands

iSQL*Plus Commands	Description
@url [arg] *For example:* *@http://mac_name.domain:port/* *myscript.sql val1 val2*	Runs SQL*Plus statements in the specifies script.
@@url *Suppose there is a script named* *myscript.sql:*	Runs a script. @@ has an additional functionality of looking for the specific script in the same URL as the script from which it was called
/ (slash)	Execute the SQL or PL/SQL block currently stored in the buffer
quit	Terminates SQL*Plus and returns control to the Operating System . Same as ***exit***

- SET commands:

SET TIMING { ON | OFF }

Controls the display of timing statistics :

- ON displays timing statistics on each SQL command or PL/SQL block run
- OFF suppresses timing of each command

SET HEAD { <u>ON</u>|OFF }

Controls printing of column headings in the result:

- ON prints column headings in reports
- OFF suppresses column headings

iSQL* Plus Commands

- SET commands:

  ### SET PAGES { 24| N}

  Controls the number of lines in each page (page size)
  - If set to 0:
    - suppresses all column headings, page breaks and other formatting information in the result\

  - If set to any integer (n):
    - sets the number of lines in each page
    - Accommodates result in single page

SET LINESIZE { 80|N }

   Sets total number of characters to display in a line

SET FEEDBACK{ 6|N|ON|OFF}

Displays the number of records returned by a query, when the
result contains at least 'n' records:

- ON or OFF turns the display on or off
- If feedback is set to 0 it is same as turning it OFF

Example:

SQL> set feedback off
SQL> select * from e where dept_id =20;

CITY	E_CODE	E_NAME	E_SALARY	M_ID	JOIN_DATE	DEPT_ID	JOB_ID
	67	Smith			01-DEC-13	20	
New York	45	Samuel	20000	34	12-DEC-12	20	MGR_ADMIN
San Francisco	8	Sam	13000	45	10-JAN-11	20	SE_ADMIN
Chicago	50	Mike	21000	45	26-DEC-17	20	SE_ADMIN

SQL> set time on

   13:47:11 SQL> set timing on

   13:47:21 SQL> select * from e where dept_id=20;

CITY	E_CODE	E_NAME	E_SALARY	M_ID	JOIN_DATE	DEPT_ID	JOB_ID
	67	Smith			01-DEC-13	20	
New York	45	Samuel	20000	34	12-DEC-12	20	MGR_ADMIN
New York	8	Sam	13000	45	10-JAN-11	20	SE_ADMIN
	50	Mike	21000	45	26-DEC-17	20	SE_ADMIN

Elapsed: 00:00:00.03
13:47:30 SQL>

Example:

> **SQL> set head off**
>
> **13:48:07 SQL> select * from e where dept_id=20;**

	67 Smith			01-DEC-13	20	
New York	45 Samuel	20000	34	12-DEC-12	20 MGR_ADMIN	
New York	8 Sam	13000	45	10-JAN-11	20 SE_ADMIN	
San Francisco	50 Mike	21000	45	26-DEC-17	20 SE_ADMIN	

> **Elapsed: 00:00:00.03**
>
> **13:48:09 SQL>**

SPOOL [file_name] [OFF|OUT]:

- Is not available in iSQL*Plus
- Stores query result in a file or sends the file to a printer
- OFF indicates stop spooling
- OUT stops spooling and sends the file to the computers default printer
- If the file_name specified already exists then it would get overwritten
- SPOOL with no clause, lists the current spooling status

Example:

```
SQL> spool t1.txt ------------------------ 1
SQL> select * from e; -------------------------2
CITY              E_CODE E_NAME    E_SALARY   M_ID   JOIN_DATE   DEPT_ID JOB_ID
----------------- ------------ ------------   ----------  --------  ---------------  ------------- -----------
                   67     Smith                           01-DEC-13   20
San Francisco      12     Jay       13000      34    11-JAN-01   10      JE_SALES
New York city      45     Samuel    20000      34    12-DEC-12   20      MGR_ADMIN
New York city      8      Sam       13000      45    10-JAN-12   20      SE_ADMIN
San Francisco      50     Mike      21000      45    26-DEC-17   20      SE_ADMIN
San Francisco      60     Morgan    25000      34    02-JAN-03   10      SE_SALES
6 rows selected. ---------------------3
SQL> spool off ------------------- 4
```

The file (t1.txt) would contain the contents from lines indicated 2 to 4
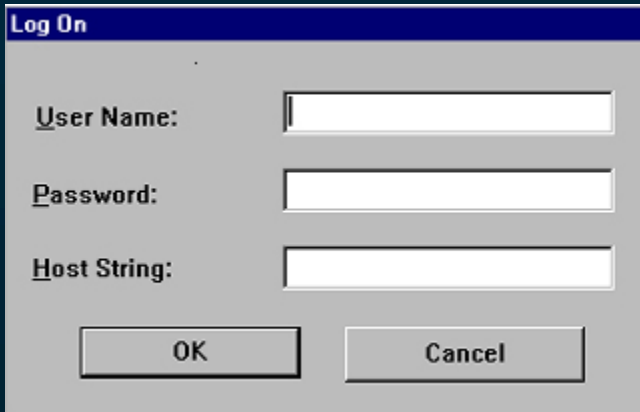
Difference between SQL and iSQL*Plus
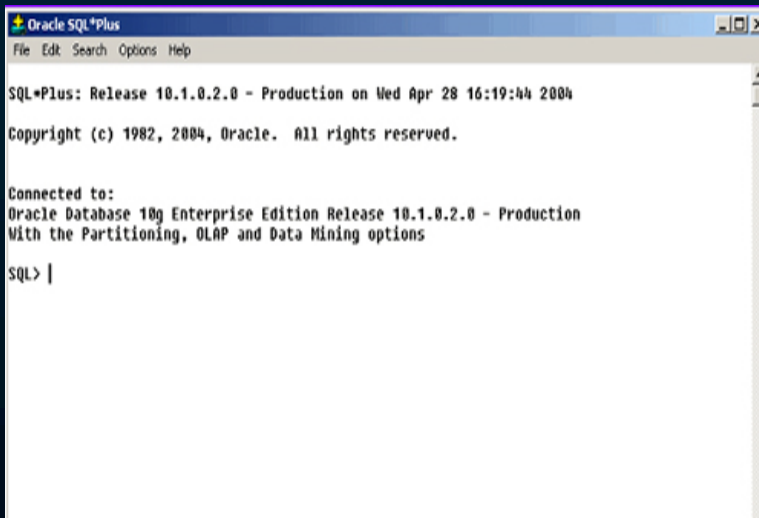
SQL	iSQL*Plus
Is a language	Is an Oracle interface
Is based on ANSI standard	Oracle's proprietary product
Abbreviation of key words not allowed	Abbreviation of key words allowed
Can manipulate data stored in database	Cannot manipulate values in the database
Termination character is required	Termination character is not required

Demonstration

- **Executing SQL statements Using SQL*Plus tool**

Steps:

1. Launch SQL plus Application

2. Provide Username and Password in appropriate boxes.

3. Host String –

   If database is on local machine
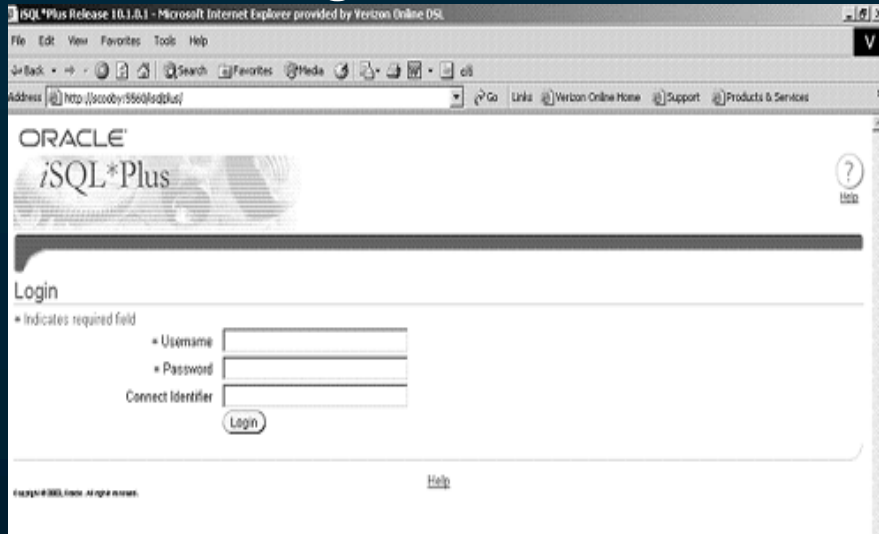
   Host String – Don't Fill any value

   Else

   Host String - Oracle Net connection string.

4. When connected enter the SQL queries in the SQL prompt for execution.

Demonstration

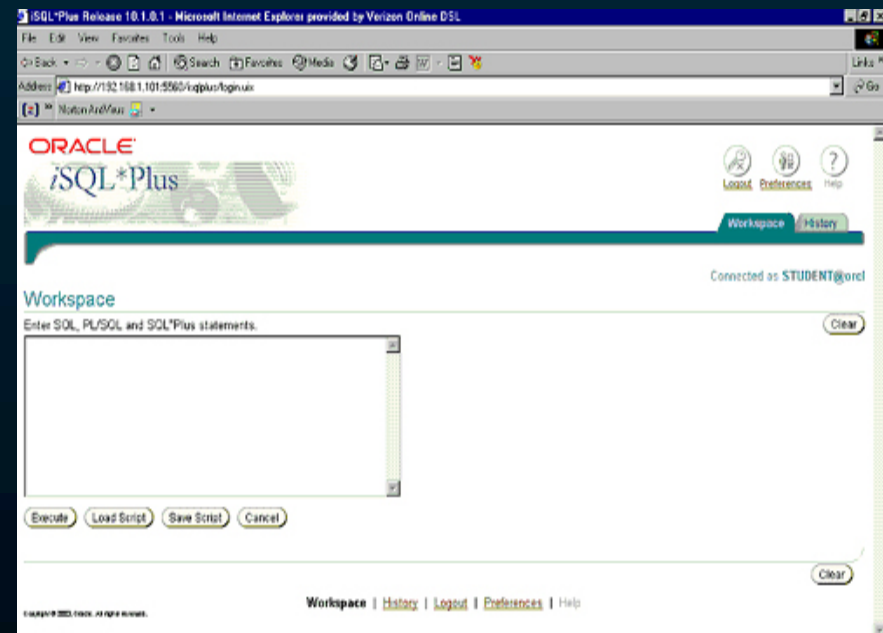- **Executing SQL statements Using iSQL*Plus browser interface**

Steps:

1. In the browser enter the URL in the following format:
   "**http://machine_name.domain:port/isqlplus**"

2. Provide Username and Password in appropriate boxes

3. When connected enter the SQL queries in the Workspace and click the execute button.

Test your understanding

1. What is the result of below query?

   select * **,e_code** **from** **employees**;

2. iSQL *Plus commands manipulate values in the database. State True or False.

3. Which of the following is not an iSQL*Plus command:
   a) connect
   b) Run
   c) Select
   d) save

Recap

In this session you have learned the following:

- Write 2 SQL select commands with or without aliasing the table names on Oracle Platform.

- Identify from the given list the commonly used iSQL*Plus commands

- List at least 2 differences between SQL and iSQL*Plus commands

You have successfully completed –
Retrieving Data Using SQL SELECT Statements