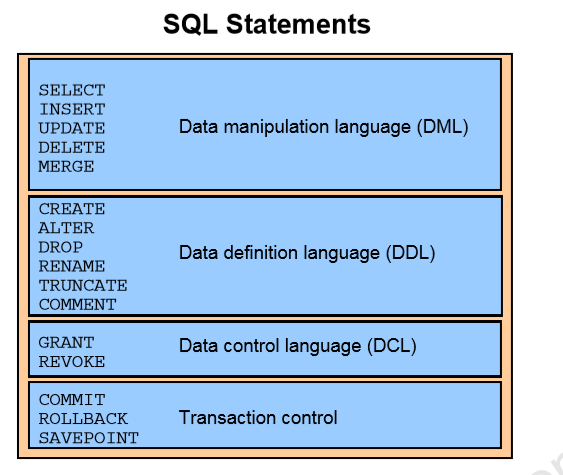
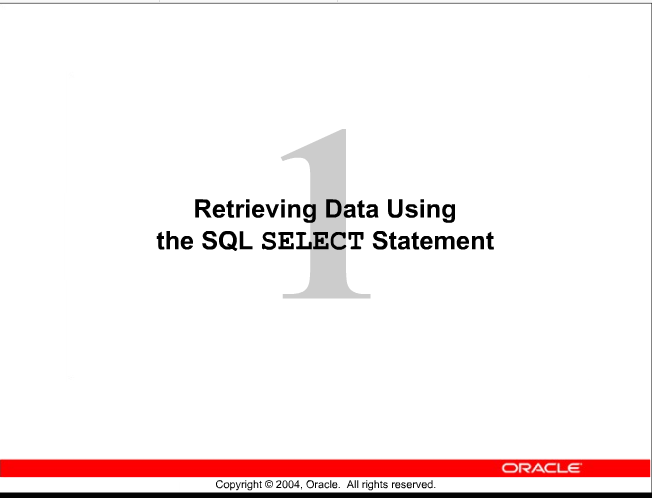
STARTING UP

1. Open your Oracle Client 32 session in MyApps and access your account. You may use SQL\*PLUS or SqlDeveloper as your Interface.
2. Username and Password are placed on BB
3. Did you load the script that populates the database? demobld.sql

Lesson 1A -- SELECT Basics Review with Oracle server

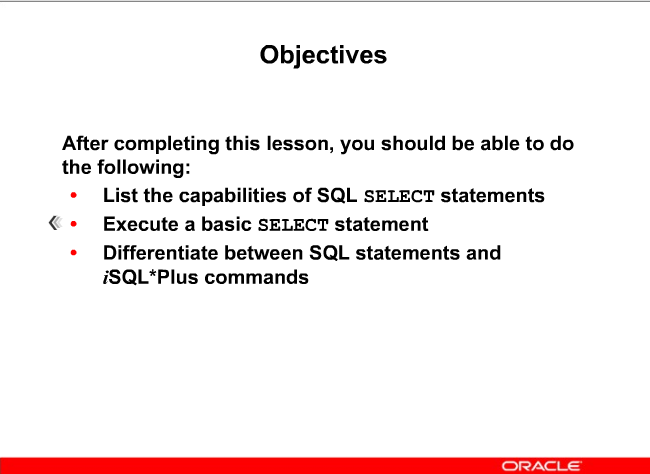
These are the SQL commands used on the course



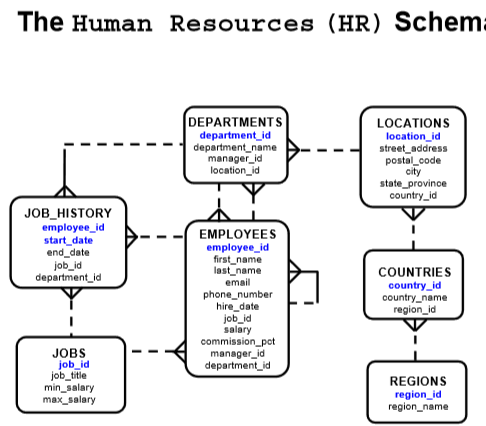


You already know this, but make a quick look to review it

2



SCHEMA – working with this semester



Our **demobld.sql** script creates only 5 of these tables, and all names are Singular,

So whenever you test the examples from these Slides, remember NOT to finish table name with S.

COUNTRY

LOCATION

DEPARTMENT

EMPLOYEE

JOB\_HISTORY

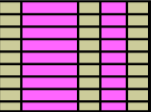
**These tables will be used for our 4 SQL labs !**

# 3 ACTIONS ON TABLES

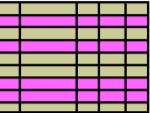
* + **1 PROJECTION**
  + **2 SELECTION**
  + **3 JOINS**
  + **Done through SELECT statement**

SELECT – Retrieving data from a table

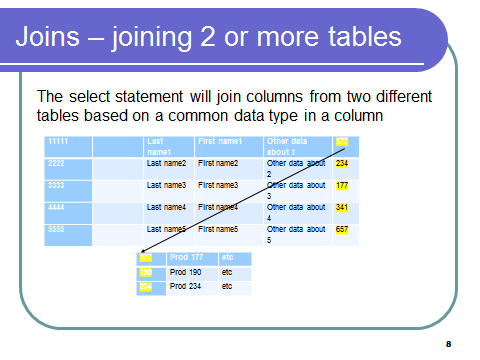
**1 PROJECTION** -- Retrieving **specific columns** of data such as ALL student names and phone numbers



**2 SELECTION** – Returns **only rows** that meet the specific restriction such as all male students from a table of students

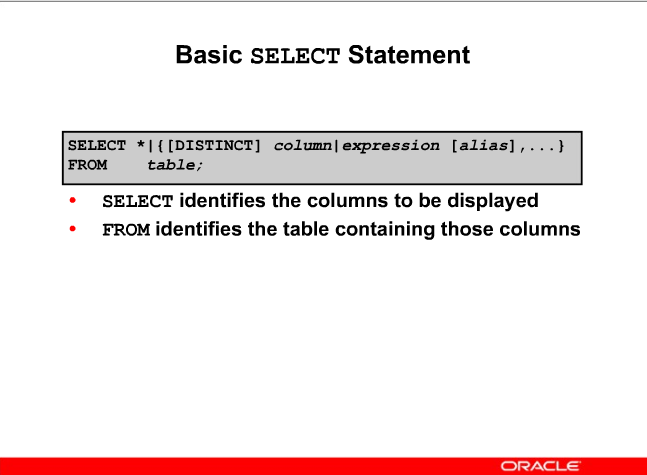


**3 JOIN** – Returning **data from 2 different** tables such as course name and the student name



More on this in a later lessons.

FORMAT of the SELECT statement



**SQL STYLE**

**It is important NOT to use run on sentences**

This and similar styles will not be marked and so will get zero

SELECT LAST\_NAME, FIRST\_NAME, ADDRESS1, ADDRESS2, CITY, PROV, PCODE FROM EMPLOYEES WHERE UPPER(CITY) = UPPER (‘TORONTO’);

Vs

SELECT LAST\_NAME, FIRST\_NAME, ADDRESS1, ADDRESS2, CITY, PROV, PCODE

FROM EMPLOYEES

WHERE UPPER(CITY) = UPPER (‘TORONTO’);

OR

SELECT LAST\_NAME, FIRST\_NAME, ADDRESS1, ADDRESS2,

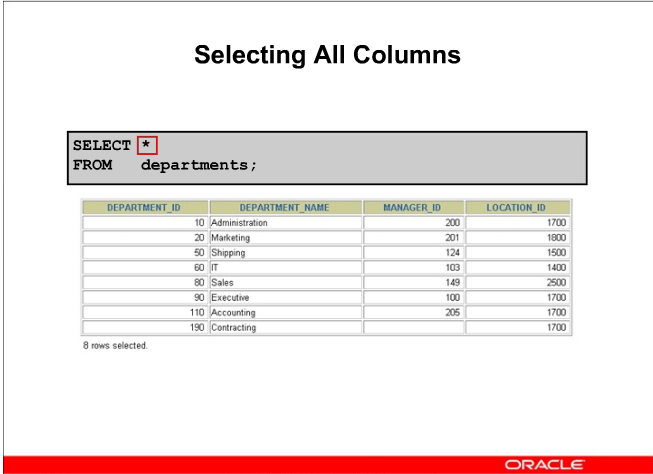
CITY, PROV, PCODE

FROM EMPLOYEES

WHERE UPPER(CITY) = UPPER (‘TORONTO’);

More data later about the style !

PROJECTION –

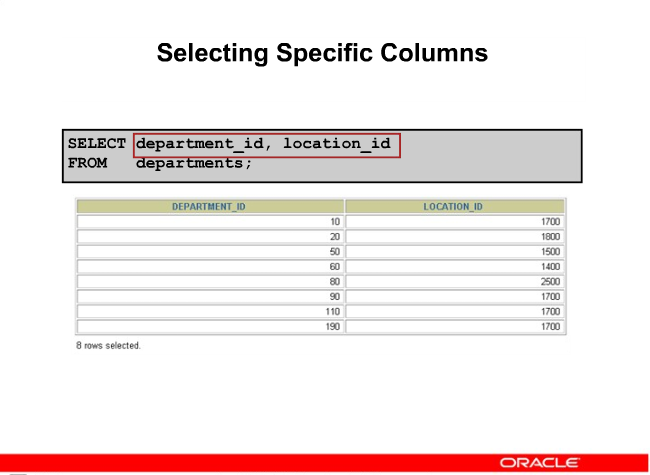


Use the DESCRIBE command to see the structure of a specific table.

DESCRIBE LOCATION or

DESCRIBE EMPLOYEE

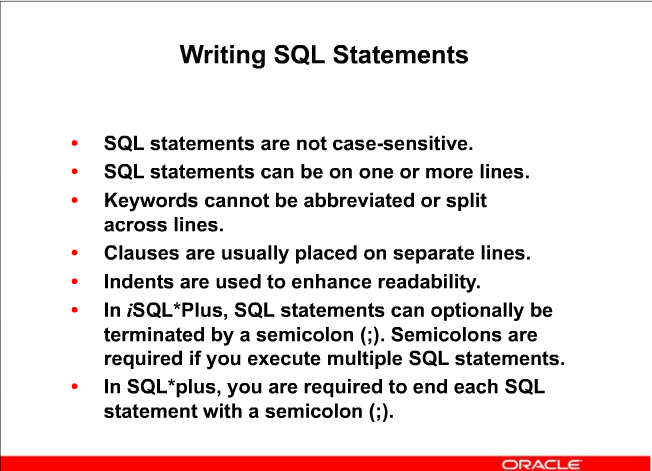
SELECTION --



SELECT department\_id, location\_id

FROM department;

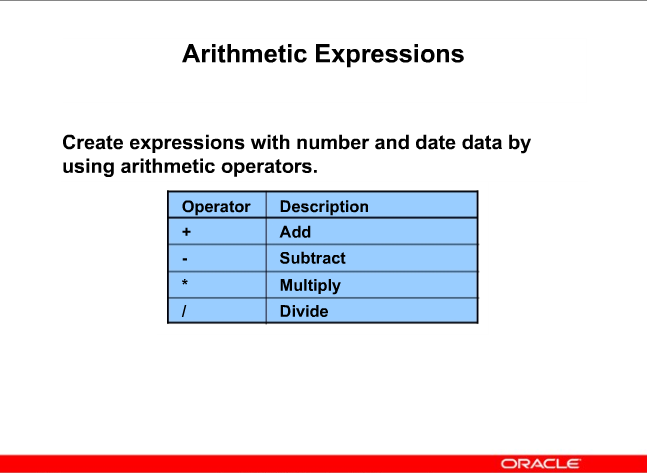
Cdo UPPER and LOWER case like this



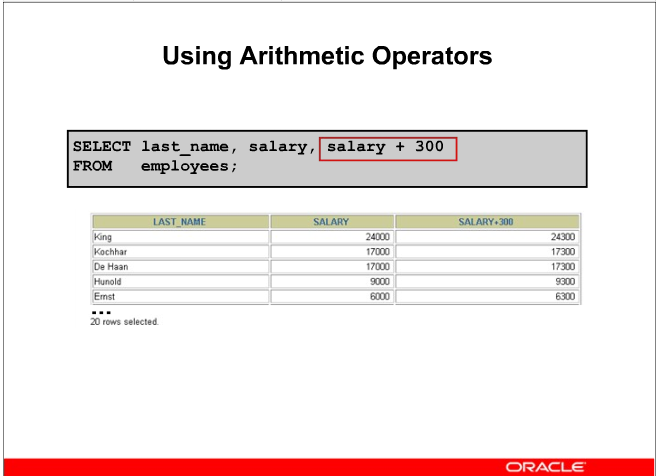
In SQL\*PLUS you need always to finish your command with ;

In SqlDeveloper you do that only when having multiple commands in a sequence.

9



Same as any programming language



Note how it adds to the salary an additional 300

As in ALL languages it is important to know the order of operation when there are multiple operators

SELECT last\_name, salary, salary + 300

FROM employees;

**Look at the EMPLOYEE table**

**We will assume the salary value for now is monthly**

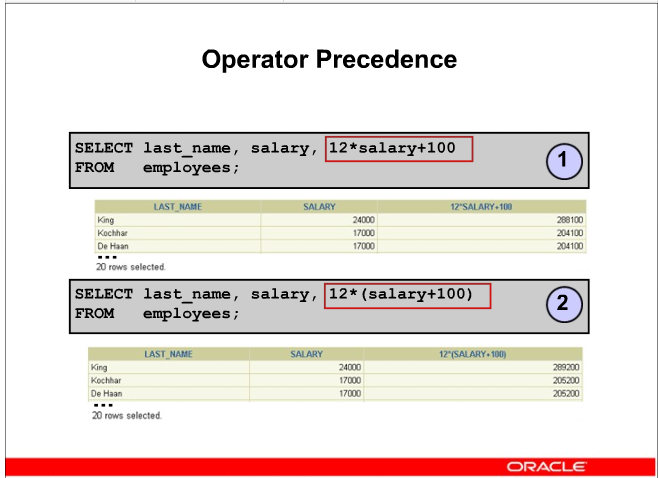
**Show**

1. **yearly salary.**
2. **Raise everyone’s salary by 100 dollars per month, then multiply it out by 12 to see yearly salary**

Try it now in SQL\*PLUS and SqlDeveloper, both.

11

Raise everyone’s salary (which we will assume is monthly) by 100 dollars then multiply it out by 12 to see yearly salary



289,200

288,100 is the result

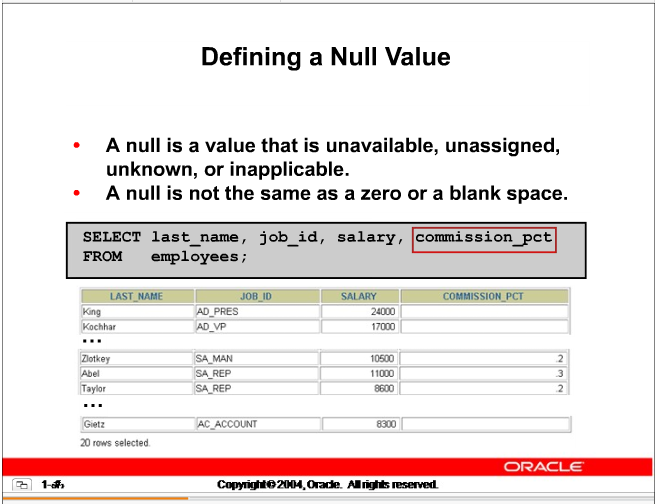
Not a large difference for EACH employee.

For the company with 20 employees that is 24,000

Now if you have 1000 employees, that raise is 24 million.

Would be nice if we could total the salaries to see the results.

12



**PROBLEM for you to solve**

Show

1. Last Name
2. Multiply the monthly salary by 12 to get yearly salary.
3. Then multiply it by the percent to get the commission earned

13

**PROBLEM**

Multiply the monthly salary by 12 to get yearly salary.

Then multiply it by the percent to get the commission earned

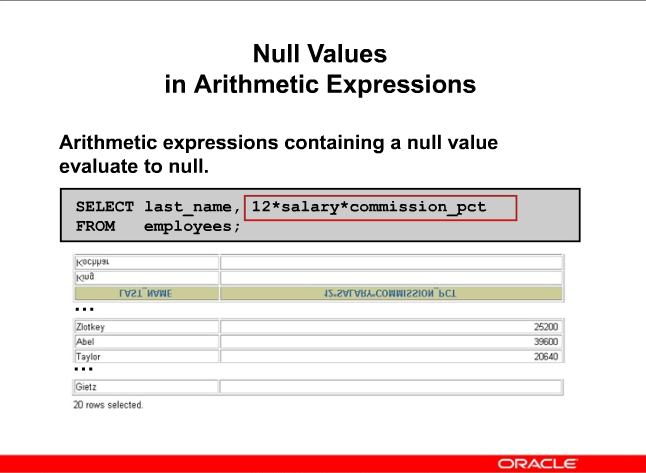
SELECT last\_name, salary, salary \* 12 \* commission\_pct

FROM employee;

THIS PROBLEM WILL HAPPEN TO YOU MANY TIMES

What Happens …. Your result is Blank (Null)

When we learn about UPPER, LOWER functions, we will handle it correctly.



SELECT last\_name, salary \* 12 \* commission\_pct

FROM employee

# NOTE: Column Names are not nice – need to fix it with an alias

LAST\_NAME SALARY\*12\*COMMISSION\_PCT

------------------------- ------------------------

King

Kochhar

De Haan

Hunold

Ernst

Lorentz

Mourgos

Rajs

Davies

Matos

Vargas

Zlotkey 25200

Abel 39600

Taylor 20640

Grant 12600

Whalen

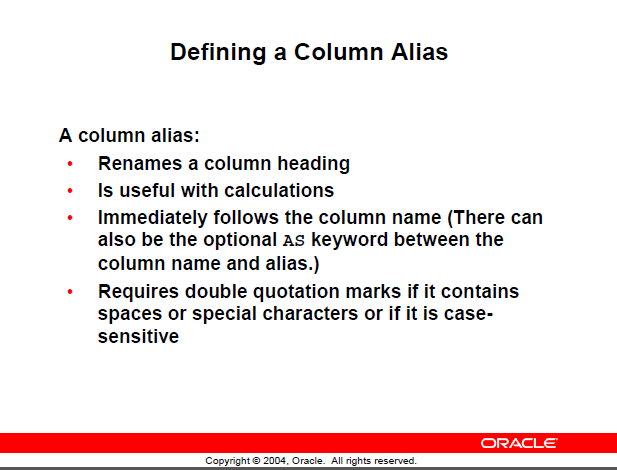
Hartstein

Fay

Higgins

Gietz

20 rows selected



**NOTE: Alias names maximum length of 30**

Try these.

SELECT last\_name AS name, commission\_pct as comm

FROM employee;

SELECT last\_name AS Last Name, commission\_pct as comm

FROM employee;

-- This last one will not work as there is a space in the alias name

**PROBLEM:**

Display last name and job id from the employees table and look at the result

SELECT last\_name, job\_id AS "Employees"

FROM employee;

LAST\_NAME Employees

----------------------- ----------------------

Abel SA\_REP

Davies ST\_CLERK

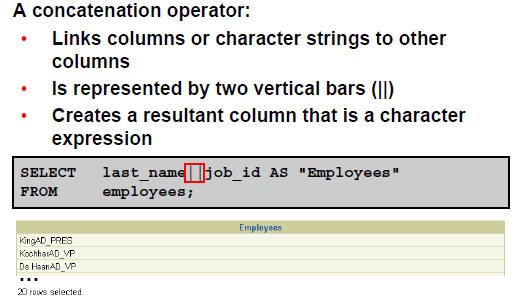
De Haan AD\_VP

Ernst IT\_PROG

Fay MK\_REP

Gietz AC\_ACCOUNT

Change the look with a **CONCATENATE OPERATOR**



Notice this works whereas on the iSeries it "appeared" to not work.

select last\_name || job\_id as "Employees"

from employee;

STILL NOT NICE LOOKING ………. Need to improve it.

**How ????? do it in question 3**

ASIDE: There is a concatenate function available later

ANSWER:

select last\_name || ' ' || job\_id as "Employees"

from employee;

Employees

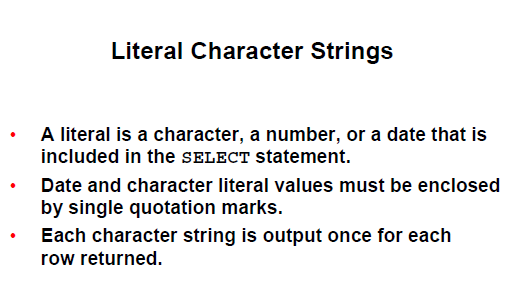
------------------------------------

Abel SA\_REP

Armarillo SA\_REP

Bergsteige SA\_REP

Brigade SA\_REP



**NOTE:**

**Single quotes for literals**

**Double quotes for alias names**

**PROBLEM Question 4:**

**Revise previous SQL 🡪 require Last name and job id with the ‘is a’ between it but nicely displayed**

select last\_name || ' is a ' || job\_id as "Employees"

from employeee;

🡸 note the spacing in the quotes

Always look at your output carefully

(answer on next page)

**PROBLEM Question 5:**

**A question from the consultant hired to get more sales.**

**How many departments does the company have?**

**Using the employees table, display all the departments with people?**

**ANSWER Question 5**

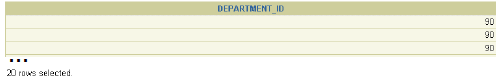
**SELECT department\_id 🡨 could use department names but then need a join**

**FROM employee;**

**Did you do this.**

**Look carefully at the results, is that what the consultant really wanted?**

See the duplicates as there are 20 rows displayed



**SELECT DISTINCT department\_id**

**FROM employee;**

DEPARTMENT\_ID

-------------

10

20

50

60

80

90

110

🡨 this is NULL value here

8 rows selected.

PROBLEM:

Show the distinct salary and job id within department. Meaning if two people in department 20 have the same salary and job\_id only show one of them.

SELECT DISTINCT (salary, job\_id), department\_id

FROM employee

ERROR 🡺 see message … does not look like the error is correct

Error starting at line : 1 in command -

SELECT DISTINCT (salary, job\_id), department\_id

FROM employees

**Error at Command Line : 1 Column : 24 🡸 this usually will help you**

Error report -

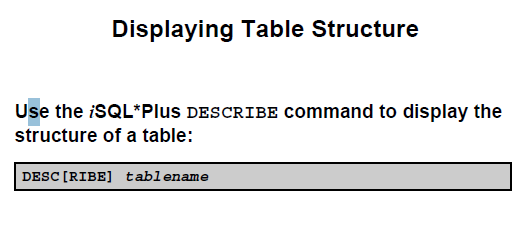
SQL Error: ORA-00907: missing right parenthesis

00907. 00000 - "missing right parenthesis"

WHY?

If you use DISTINCT, it must be for ALL selected columns

SQLPLUS command and NOT SQL



This is not SQL it is a SQL\*Plus command and does not need a semicolon

