

Introduction to Oracle and SQL

Learning Objective

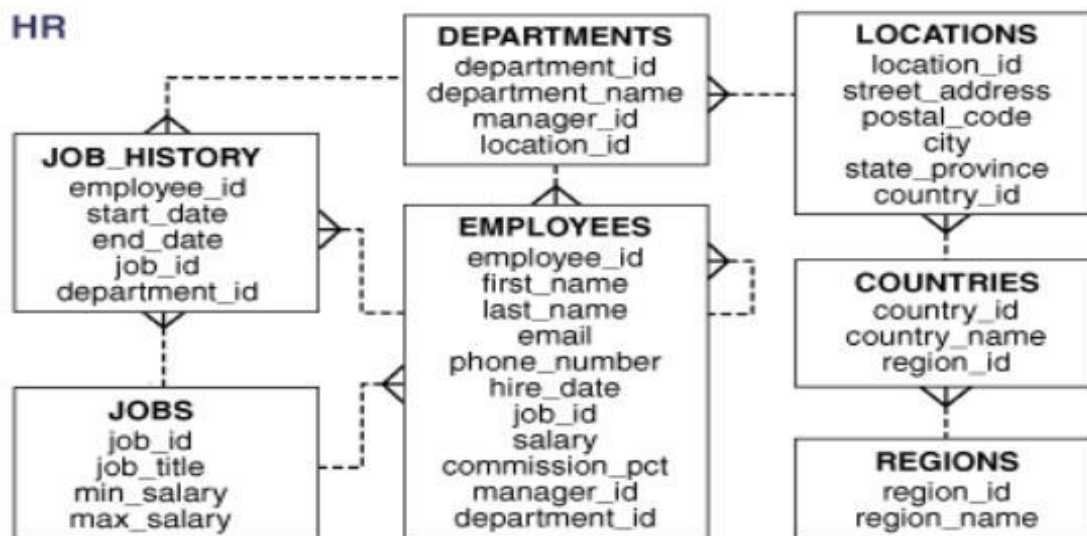
After completing this lab the student should be able to:

- Difference between single row function and multiple row function.
- Discuss Logical operator and their use.
- Describe various types of functions that are available in SQL.
- Use character, number, and date functions in SELECT statements.
- Describe the use of conversion functions.

Tools and Technologies

- Oracle 11g Express edition / enterprise edition

HR Schema for Reference



Logical operators

A logical operator combines the result of two component conditions to produce a single result based on them or to invert the result of a single condition. Three logical operators are available in SQL.

- **AND** Returns TRUE if both component conditions are TRUE
- **OR** Returns TRUE if either component condition is TRUE
- **NOT** Returns TRUE if the following condition is FALSE

You can use several conditions in one WHERE clause using the AND and OR operators.

Example

Syntax

Select column_name, all

From table Name

Where column_name=value and column_name=value;

Write a query to display employee number, first name, job id ,salary form those employees whose salary is greater than equal to 1100 and job_id is 'IT_PROG'

SELECT employee_id,first_name, job_id ,salary from employees where

salary>=1100 and job_id='IT_PROG';

Exercise

Write a query to display job_id and min salary from those jobs whose job title 'President and maximum salary is 400000

**Select job_id, min_salary
From jobs
Where job_title='President' and max_salary=400000;**

Write a query to show job start date and job end date of those employees whose department id is 110 and job title is 'AC_ACCOUNT'

**select start date, end date
from employees
where department_id=110 and job_title='AC_ACCOUNT'**

Write a query to display first_name , last_name of those employees whose job_id is 'ST_CLERK' and employee number is 131;

**select first_name, last_name
from employees
where job_id='ST_CLERK' and employee_id=131;**

OR operator

Example

Write a query to display employee number, first name, job title ,salary form those employees whose salary is greater than equal to 1100 or job_id is 'IT_PROG'

**SELECT employee_id,first_name, job_id ,salary from employees where salary>=1100
or job_id='IT_PROG';**

Lab task 1:

Write a query to display salary and job tile of those employee whose salary is greater than 4000 or job title IT_PROG

Write a query to show country code, country name of those employees whose country name 'Argentina' or region id 4

Not IN

Example

Write a query to display first_name,job title form those employees whose job title not in st_clerk and programmer.

Select first_name,job_id

From employees

Where job_id not in ('ST_CLERK', 'PROGRAMMER')

Lab task 2:

Write a query to display employee_id,salary from those employees where department_id not 50 and 80.

Write a query to show all records in job_history except those records whose Job_id ('IT_PROG', 'AC_ACCOUNT')

CS2134-Introduction to Database System

SQL – Data Manipulation Language (DML)

Rules of Precedence

Order Evaluated	Operator
1	All comparison operators
2	NOT
3	AND
4	OR

Override rules of precedence by using parentheses

Check these two examples

**SELECT first_name, job_id ,salary from employees where job_id='ST_CLERK'
or job_id ='IT_PROG' and salary >6000;**

**SELECT first_name, job_id ,salary from employees where (job_id='ST_CLERK'
or job_id ='IT_PROG') and salary >6000;**

Order by clause

The order by clause can be used to sort the rows. If it is used then it must be placed in last. The default sort order is ascending. To reverse the order you specify the keyword DESC after column name in ORDE BY clause.

e.g

Syntax

**Select comlumn_name
From Table_name
Order by column_name**

**SELECT first_name, job_id ,department_id,hire_date from employees
order by hire_date asc ,first_name asc;**

**SELECT first_name, job_id ,department_id,hire_date from employees order
by hire_date desc;**

You can use a column alias in the order by clause.

e.g

**SELECT employee_id,first_name,salary *12 annualsalary from employees
order by annualsalary ;**

Write a query to show all country name is descending order

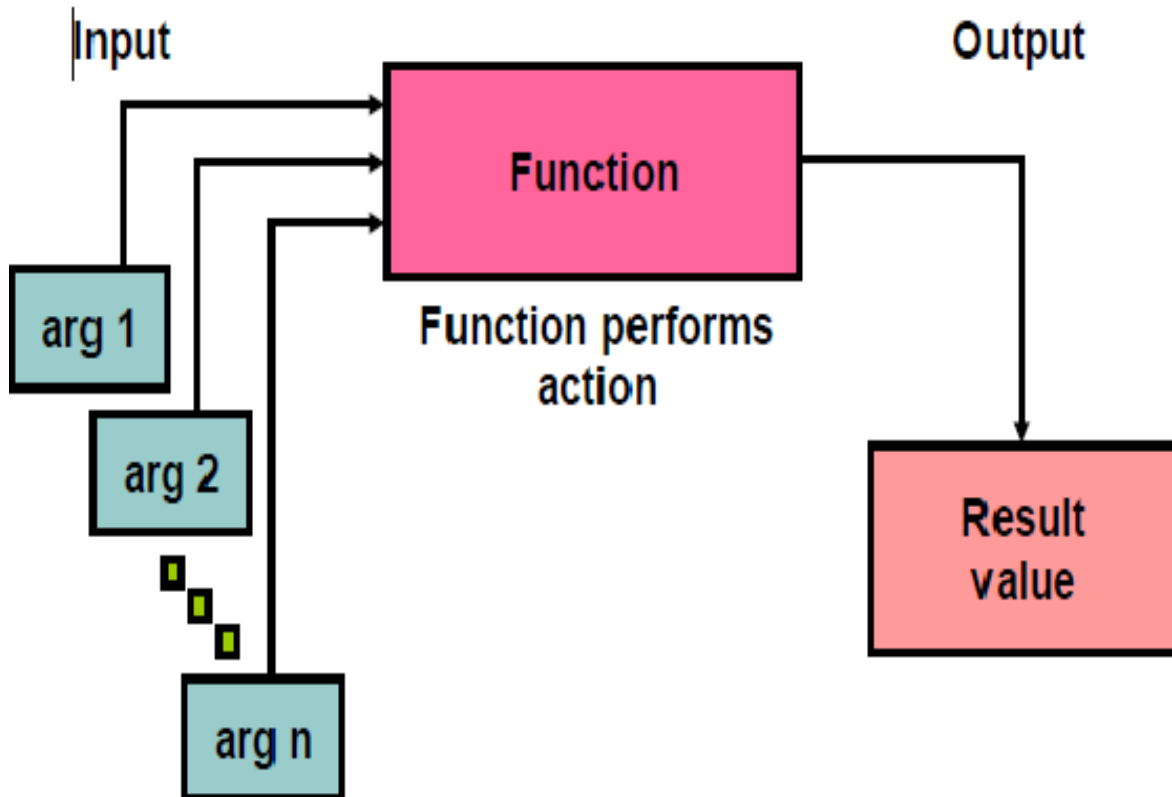
Sorting by multiple columns

You can sort query results by more than one column. The sort limit is the number of columns in the given table.

e.g

SELECT first_name,department_id,salary from employees order by department_id , salary desc;

SQL Functions



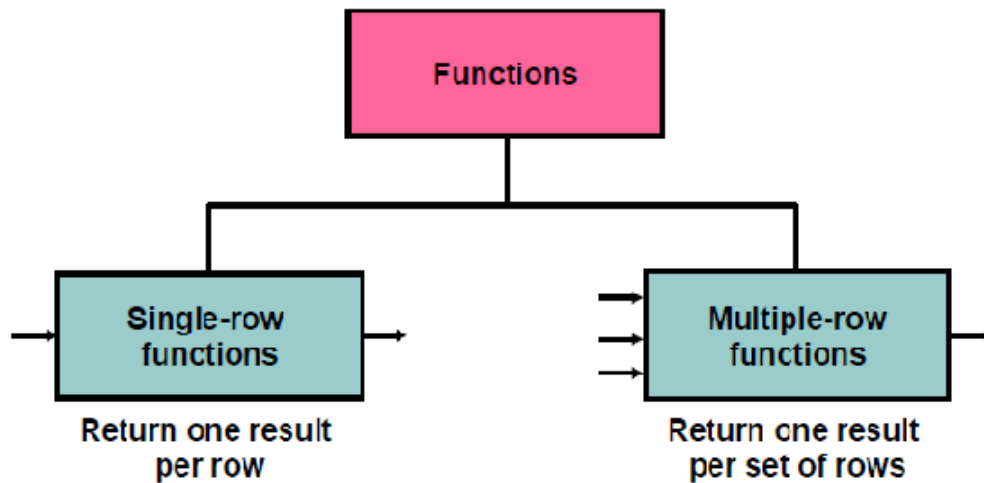
Functions are a very powerful feature of SQL and can be used to do the following:

- Perform calculation on data
- Modify individual data items
- Manipulate output for groups of rows
- Format dates and numbers for display
- Convert column data types

CS2134-Introduction to Database System

SQL – Data Manipulation Language (DML)

Two Types of Function



Character Manipulation Function

CONCAT, SUBSTR, LENGTH, INSTR, and LPAD are the five character manipulation functions

- CONCAT: Joins values together
- SUBSTR: Extracts a string of determined length
- LENGTH: Shows the length of a string as a numeric value
- INSTR: Finds numeric position of a named character
- LPAD: Pads the character value right-justified
- RPAD: Pads the character value left-justified

Functions	Result
CONCAT ('Good','String')	GoodString
SUBSTR ('String',1,3)	Str
LENGTH ('String')	6
INSTR ('String', 'r')	3
LPAD (salary , 10, '*')	*****5000

E.g

To display employee name and job_id joined together, length of the employee name, and the numeric position of the letter A in the employee name, for all employees who are in department 110

```
SELECT first_name, CONCAT (first_name, job_id ), LENGTH (first_name), INSTR  
      (first_name,'A')  
FROM employees  
WHERE department_id=110;
```

Lab Task

Write a query to display full name of that employees whose first name contains 'a' and salary is 6000, 8400, 8600 and 6100.

Display the full name of each employee, and please note each name is right justify.

Case Conversion Function

LOWER, UPPER and INITCAP are the three case conversion functions.

- LOWER: Converts mixed case or uppercase character string to lowercase.
- UPPER: Converts mixed case or lowercase character string to uppercase.
- INITCAP: Converts first letter of each word to uppercase and remaining letter to lowercase.

Function	Result
LOWER ('SQL Course')	sql course
UPPER ('SQL Course')	SQL COURSE
INITCAP ('SQL Course')	Sql Course

Example

Display the employee number, last name, and department number for employee higgins.

```
SELECT employee_id,  
last_name,department_id FROM employees  
WHERE LOWER (last_name)='higgins';
```

Lab Task 2

Write a query to show all country name in small letter and display all records in country table

Write a query to show all employees-name in capital letters of those employee whose salary range in 6000 to 12000.

CS2134-Introduction to Database System

SQL – Data Manipulation Language (DML)

Number Function

Number function accepts numeric input and returns numeric values.

- Round: Rounds value to specified decimal
Round (45.926,2) → 45.93
- Trunc: Truncates value to specified decimal
Trunc(45.926,2) → 45.92
- Mod: Returns remainder of division
Mod (1600,300) → 100

DUAL

Dual is a dummy table owned by the user SYS and can be accessed by all users to view results from functions and calculations

E.g

SELECT ROUND (45.923,2), ROUND (45.923,1) from dual;

Calculate the remainder of the ratio of salary to commission for all employees whose job_id title is sale representative.

```
SELECT first_name, salary, MOD (salary ,5000) FROM
employees
WHERE job_id ='SA_REP';
```

Lab Task 3

Write a query to show all those employee who is getting an even number salary.

Write a query round off, and truncate these values after the decimal up to 3 digits.

SYSDA

SYSDATE is a date function that returns the current date and time. You can use sysdate just as you would use any other column name. For example you can display the current date by selecting sysdate from a table. It is customary to select sysdate from a dummy table called dual

Example

Display the current date of the system.

Select sysdate from dual

CS2134-Introduction to Database System

SQL – Data Manipulation Language (DML)

Oracle Date Format

Oracle stored dates in an internal numeric format, representing the century, year, month, day, hours, minutes, and seconds.

Date Functions

Date functions operate on Oracle dates. All date functions returns a value of DATE datatypes except MONTHS_BETWEEN, which return a numeric value.

- MONTHS_BETWEEN (date1, date2): Finds the number of months between date 1 and date 2.

e.g

MONTHS_BETWEEN ('01-SEP-95','11-JAN-94') → 19.6774194

- ADD_MONTHS (date, n): Adds n number of calendar months to date. The value of n must be an integer and can be negative.

e.g

ADD_MONTHS ('11-JAN-94', 6) → '11-JUL-94'

- NEXT_DAY (date, 'char'): Finds the date of the next specified day of the week ('char') following date. The value of char may be a number representing a day or a character string.

e.g

NEXT_DAY ('01-SEP-95', 'FRIDAY') → '08-SEP-95'

- LAST_DAY (date): Finds the date of the last day of the month that contains date.

e.g

LAST_DAY ('01-SEP-95') → '30-SEP-95'

E.g

For all employees employed for fewer than 400 months, display the employee number, hire_date, number of months employed, six-month review date, first Friday after hire date, and last day of the month when hired.

```
SELECT employee_id, hire_date,
MONTHS_BETWEEN (SYSDATE, hire_date) TENURE,
ADD_MONTHS (hire_date, 6) REVIEW,
NEXT_DAY (hire_date, 'FRIDAY'), LAST_DAY(hire_date)
FROM employees
WHERE MONTHS_BETWEEN (SYSDATE, hire_date) <400;
```

CS2134-Introduction to Database System

SQL – Data Manipulation Language (DML)

Lab Task 4

For each employee display the employee name and calculate the number of months between today and the date the employee was hired. Label the column MONTHS_WORKED. Order your result by the number of months employed. Round the number of month up to the closest whole number.

Conversion Functions

In addition to Oracle datatypes, columns of tables in an Oracle database can be defined using ANSI, DB2, and SQL/DS datatypes. However, the oracle Server internally converts such datatypes to Oracle datatypes.

In some cases, Oracle server allows data of one datatypes where it expects data of a different datatype. This is allowed when Oracle server can automatically convert the data to the expected datatypes. This datatypes conversion can be done implicitly by Oracle server or explicitly by the user.

Explicit datatypes conversion are done by using the conversion functions. Conversion functions convert a value from one datatype to another.

Generally the form of the function names follows the convention *datatype TO datatyps*. The first datatype is the input datatype and the last datatype is the output.

Implicit Datatypes Conversion

For assignments, the oracle can automatically convert the following;

From	To
VARCHAR2	NUMBER
VARCHAR2 or CHAR	DATE
NUMBER	VARCHAR2
DATE	VARCHAR2

CHAR to NUMBER conversion succeed only if the character string represents a valid number.

CHAR to DATE conversion succeed only if the character string has the default format DD-MON-YY.

Explicit Datatypes Conversion

SQL provides three functions to convert a value from one datatype to another.

The format model must be enclosed in single quotation marks and is case sensitive.

Function	Purpose
TO_CHAR (number/date, ['fmt'])	Converts a number or date value to a VARCHAR2 character string with format model <i>fmt</i>
TO_NUMBER (char, ['fmt'])	Converts a character string containing digits to a number with the optional format model <i>fmt</i>
TO_DATE (char,['fmt'])	Converts a character string representing a date to date value according to the fmt

CS2134-Introduction to Database System

SQL – Data Manipulation Language (DML)

	specified. (If <i>fmt</i> is omitted, format is DD-MON-YY.)
--	---

E.g

To display the name and hire dates as this format 17 November 1981 for all the employees

```
SELECT first_name, TO_CHAR (sysdate,'fmDD Month YYYY') hiredate  
FROM employees;
```

The NVL Function

To convert a null value to an actual value we use the NVL function.

Syntax

NVL (expr1,
expr2) Where:

Expr1 is the source value or expression that may contain
null Expr2 is the target value for converting

You can use the NVL function to convert any datatype but return value is always the same as the datatype of expr1.

E.G

To calculate the annual compensation of all employees, you need to multiply the monthly salary by 12 and then add the commission to it. But here the salary will be calculated only for those employees who earn a commission. If any column value is an expression is null, the result is null.

So to calculate values for all employees, you must convert the null value to a number before applying the arithmetic operator.

```
SELECT first_name, salary , comm, (salary *12)+NVL(commission_pct,0)  
FROM employees;
```

Lab Task 5

Create a query that will display the employee name and commission amount. If the employee does not earn commission put 99 label the column commission_pct.

Decode Function

The DECODE function decodes an expression in a way similar to the IF-THEN-ELSE logic used in various languages. The DECODE function decodes expression after comparing it to each search value. If the expression is the same as search, result is returned. If the default value is omitted a null value is returned where a search value does not match any of the result values.

E.g

```
SELECT job_id ,salary ,
```

CS2134-Introduction to Database System

SQL – Data Manipulation Language (DML)

```
        DECODE ( job_id , 'PROGRAMMER', SALARY *1.1,  
                    'ST_CLERK', SALARY *1.15, 'SALES  
                    MANAGER', SALARY *1.20,  
                    SALARY )  
        REVISED_SALARY  
FROM employees;
```

Note: Complete all Lab Tasks.

Lab Instructor:

Qazi Shuja Ud Din, Riphah International University
qazi.shujauddin@riphah.edu.pk