# Databases Lab 2

We are going to continue to practice some SQL queries. Ensure that the DBMS you have chosen has the following tables:

DEMO\_CUSTOMERS, DEMO\_ORDERS, DEMO\_ORDER\_ITEMS, DEMO\_PRODUCT\_INFO, DEPT, EMP

If not use the supplied scripts to upload them as described in Lab 1.

### Exercise

Enter the following SQL commands into the SQL Editor and check you see the results depicted; ensure you understand the queries:

1. Display all details of all products:

**Select \* from demo\_product\_info**

1. Display product ids and product names in the format ‘id: name*’* with heading “Product” and product description with heading “Description”:

**ORACLE**

**select product\_id || ': ' || product\_name "Product", product\_description "Description" from demo\_product\_info**

**MYSQL**

**select concat(product\_id, ': ', product\_name) "Product", product\_description "Description" from demo\_product\_info**

1. Display product id, product name and description for products with the category of ‘Mens’:

**ORACLE**

**select product\_id || ': ' || product\_name "Product", product\_description "Description" from demo\_product\_info**

**where category='Mens'**

**MYSQL**

**select concat(product\_id, ': ', product\_name) "Product", product\_description "Description" from demo\_product\_info**

**where category='Mens'**

1. Display product id, product name and description for products where the list price is 50 or less:

**ORACLE**

**select product\_id || ': ' || product\_name "Product", product\_description "Description" from demo\_product\_info**

**where list\_price <= 50**

**MYSQL**

**select concat(product\_id, ': ', product\_name) "Product", product\_description "Description" from demo\_product\_info**

**where list\_price <= 50**

1. Display product id, product name, description and list price for products where the list price is greater than 100 with highest list price first.

**ORACLE**

**select product\_id || ': ' || product\_name "Product",**

**product\_description "Description", list\_price "Price"**

**from demo\_product\_info**

**where list\_price > 100**

**order by list\_price desc**

**MYSQL**

**select concat(product\_id, ': ', product\_name) "Product", product\_description "Description" from demo\_product\_info**

**where list\_price > 100**

**order by list\_price desc**

1. Adjust the previous query to limit list price to less than 125:

**ORACLE**

**select product\_id || ': ' || product\_name "Product", product\_description "Description", list\_price "Price"**

**from demo\_product\_info**

**where list\_price between 101 and 124**

**order by list\_price desc**

**MYSQL**

**select concat(product\_id, ': ', product\_name) "Product", product\_description "Description", list\_price "Price" from demo\_product\_info**

**where list\_price** **between 101 and 124**

**order by list\_price desc**

1. Display product id, product name, description and list\_price for products with the category of ‘Mens’ and list price at least 80 with highest list price first.

**ORACLE**

**select product\_id || ': ' || product\_name "Product", product\_description "Description", list\_price "Price"**

**from demo\_product\_info**

**where list\_price >= 80 and category = 'Mens'**

**order by list\_price desc**

**MYSQL**

**select concat(product\_id, ': ', product\_name) "Product", product\_description "Description", list\_price "Price" from demo\_product\_info**

**where list\_price >= 80 and category = 'Mens'**

**order by list\_price desc**

1. Display the product id, name, description and list price of the Blouse and Skirt products.

**ORACLE**

**select product\_id || ': ' || product\_name "Product", product\_description "Description", list\_price "Price"**

**from demo\_product\_info**

**where product\_name in ('Blouse', 'Skirt')**

**MYSQL**

**select concat(product\_id, ': ', product\_name) "Product", product\_description "Description", list\_price "Price" from demo\_product\_info**

**where product\_name in ('Blouse', 'Skirt')**

1. Display details of Shoes products:

**ORACLE**

**select product\_id || ': ' || product\_name "Product", product\_description "Description", list\_price "Price"**

**from demo\_product\_info**

**where product\_name like '%Shoes'**

**MYSQL**

**select concat(product\_id, ': ', product\_name) "Product", product\_description "Description", list\_price "Price" from demo\_product\_info**

**where product\_name like '%Shoes'**

1. Display details of products considered suitable for women:

**ORACLE**

**select product\_id || ': ' || product\_name "Product", product\_description "Description", list\_price "Price"**

**from demo\_product\_info**

**where product\_description like '%women%'**

**or category = 'Womens'**

**MYSQL**

**select concat(product\_id, ': ', product\_name) "Product", product\_description "Description", list\_price "Price" from demo\_product\_info**

**where product\_description like '%women%'**

**or category = 'Womens'**

1. Display all non-business products:

**ORACLE**

**select product\_id || ': ' || product\_name "Product", product\_description "Description", list\_price "Price"**

**from demo\_product\_info**

**where product\_description not like '%business%'**

**MYSQL**

**select concat(product\_id, ': ', product\_name) "Product", product\_description "Description", list\_price "Price" from demo\_product\_info**

**where product\_description not like '%business%'**

1. Display product details where the penultimate (second last) character of the category is ‘n’:

ORACLE

**select product\_id || ': ' || product\_name "Product", product\_description "Description", list\_price "Price"**

**from demo\_product\_info**

**where category like '%n\_'**

MYSQL

**select concat(product\_id, ': ', product\_name) "Product", product\_description "Description", list\_price "Price" from demo\_product\_info**

**where category like '%n\_'**

### Exercise

Write and test SQL queries to answer each of the requests below:

1. Display all details of all employees (14 rows)

Select \* from emp

1. Display employee no, name and job - in the format ‘no: space name space (job)’ ordered by name in ascending order (14 rows)

SELECT concat('No: ', ENAME , ' ', JOB) "Emp info" from emp order by ENAME ASC

1. Display name and department no of all employees whose salary is greater than 2000 (6 rows)

SELECT ENAME , DEPTNO from emp WHERE SAL > 2000

1. Display name and department no of employees whose salary is within the range 2000 and 3000 inclusive i.e. including the boundary values (5 rows)

SELECT ENAME , DEPTNO from emp WHERE SAL between 2000 and 3000

1. Display name and department no of employees whose salary is within the range 2000 and 3000 exclusive i.e. not including the boundary values (3 rows)

SELECT ENAME , DEPTNO from emp WHERE SAL between 2001 and 2999

1. Display name and job of employees who work in departments 20 and 30 (11 rows)

SELECT ENAME , JOB from emp WHERE DEPTNO = 20 OR DEPTNO = 30

1. Display name and job of employees who do not work in departments 20 and 30 (3 rows)

SELECT ENAME , JOB from emp WHERE NOT DEPTNO = 20 AND NOT DEPTNO = 30

1. Display details of employees in department 20 with salary greater than 2000 (3 rows)

SELECT ENAME , JOB from emp WHERE DEPTNO = 20 and SAL > 2000

1. Display details of employees whose salary is greater than 2000 and whose manager has id 7839 but do not work in department 10 (2 rows)

SELECT \* FROM `emp` WHERE SAL > 2000 AND MGR = 7839 AND NOT DEPTNO = 10

1. Display the names of all employees hired in 1981 (10 rows)

SELECT ENAME from emp where year(HIREDATE) = 1981

1. Display the names and salaries of all employees who can earn commission in employee no order within descending order of salary (4 rows)

SELECT `ENAME`,`SAL` from emp WHERE `COMM`> 0

ORDER BY `SAL` DESC , `EMPNO` ASC