Introduction to Database

Lab Manual Midterm

American International University-Bangladesh

Lab 01: Writing Basic SQL Statements

Objectives*:*

To extract data from the database, you need to use the Structured Query Language (SQL) SELECT statement. You may need to restrict the columns that are displayed. This lesson describes all the SQL statements that you need to perform these actions. You may want to create SELECT statements that can be used time and time again. This lesson also covers the use of SQL*Plus commands to execute SQL statements.

Topics to be discussed:

- Capabilities of SQL SELECT Statements
- Basic SELECT Statement
- Writing SQL Statements
- Executing SQL Statements
- Selecting All Columns, All Rows
- Arithmetic Expressions
- Arithmetic Operators
- Using Arithmetic Operators
- Operator Precedence
- Using Parentheses
- Null Values
- Column Aliases
- Concatenation Operator
- Literal Character Strings
- Duplicate Rows
- Displaying Table Structure

Exercise:

1. Write a query to display the name, department number for all employees.
2. Create a unique listing of all jobs that are in department 30.
3. Include the location of department 30 in the output.
4. Write a query to display the employee name, employee no of all employees who earn a commission.
5. Display the employee name and department no for all employees who have an A in their name.

Lab 02: Restricting and Sorting Data

Objectives:

While retrieving data from the database, you may need to restrict the rows of data that are displayed or specify the order in which the rows are displayed. This lesson explains the SQL statements that you will use to perform these actions.

Topics to be discussed:

- Limiting Rows Using a Selection
- Limiting Rows Selected
- Using the WHERE clause
- Character Strings and Dates
- Comparison Operators
- Using the Comparison Operators
- Other Comparison Operators
- Logical Operators
- Rules of Precedence
- The ORDER BY Clause
- Default Ordering of Data
- Reversing the Default Order
- Sorting by Multiple Columns

Exercise:

1. Create a query to display the name and salary of employees earning more than $2850.
2. Create a query to display the employee name and department number for employee number 7566.
3. Display the employee name, job, and start date of employees hired between February 20, 1981, and May 1, 1981. Order the query in ascending order by start date.
4. Display the employee name and department number of all employees in departments 10 and 30 in alphabetical order by name.
5. Write a query to list the name and salary of employees who earn more than $1500 and are in department 10 or 30. Label the columns Employee and Monthly Salary, respectively. Resave your SQL statement to a file named *p2q6.sql*. Rerun your query.
6. Display the name and hire date of every employee who was hired in 1982.
7. Display the name and job title of all employees who do not have a manager.
8. Display the name, salary, and commission for all employees who earn commissions. Sort data in descending order of salary and commissions

Lab 03: Single-Row Function

Objective:

Functions make the basic query block more powerful and are used to manipulate data values. This is the first of two labs that explore functions. You will focus on single-row character, number, and date functions, as well as those functions that convert data from one type to another—for example, character data to numeric.

Topics to be discussed:

- SQL Functions
- Types of SQL Functions
- Single-row Functions
- Types of Single-row Functions
- Nesting Functions
- DUAL

Exercise:

1. Write a query to display the current date. Label the column Date.
2. Display the employee number, name, salary, and salary increase by 15% expressed as a whole number. Label the column New Salary.
3. Modify your previous query to add a column that will subtract the old salary from the new salary. Label the column Increase. Rerun your query.
4. Display the employee's name, hire date, and salary review date, which is the first Monday after six months of service. Label the column REVIEW. Format the dates to appear in the format similar to "Sunday, the Seventh of September, 1981."
5. 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 results by the number of months employed. Round the number of months up to the closest whole number.
6. Write a query that produces the following for each employee: <employee name> earns <salary> monthly but wants <3 times salary>. Label the column Dream Salaries.
7. Write a query that will display the employee's name with the first letter capitalized and all other letters lowercase and the length of their name, for all employees whose name starts with J, A, or M. Give each column an appropriate label.
8. Create a query that will display the employee name and commission amount. If the employee does not earn commission, put "No Commission." Label the column COMM.
9. Create a query that displays the employees' names and indicates the amounts of their salaries through asterisks. Each asterisk signifies a hundred dollars. Sort the data in descending order of salary. Label the column EMPLOYEE_AND_THEIR_SALARIES.

Lab 04: Aggregate Function

Objective:

Aggregate functions return a single result row based on groups of rows, rather than on single rows. Aggregate functions can appear in select lists and in ORDER BY and HAVING clauses. They are commonly used with the GROUP BY clause in a SELECT statement, where Oracle Database divides the rows of a queried table or view into groups. In a query containing a GROUP BY clause, the elements of the select list can be aggregate functions, GROUP BY expressions, constants, or expressions involving one of these. Oracle applies the aggregate functions to each group of rows and returns a single result row for each group.

Topics to be discussed:

- Aggregate Functions
- Frequently Used Aggregate Functions
- GROUP BY clause
- HAVING clause

Exercise:

1. Find average, maximum, minimum salary of the employees.
2. Find average, maximum, minimum salary of the employees according to department number.
3. Find average, maximum, minimum salary of the employees according to job category.
4. Find the name of lowest paid manager. (Manager is not Job).
5. Find the location where maximum number of employee is located
6. Find out job group having highest amount of total salary. (Sal + comm)
7. Suppose you need to know the name and department no. of the employee who earns the highest salary. Write a SQL query to return this information.

Lab 05: Constraints

Objective:

Constraints are the rules enforced on data columns on table. These are used to limit the type of data that can go into a table. This ensures the accuracy and reliability of the data in the database. Constraints could be column level or table level. Column level constraints are applied only to one column, whereas table level constraints are applied to the whole table. The constraints can be added when we create table. You can also add or change constraint after table has been created.

Topics to be discussed:

- Constraints
- Commonly used constraints available in SQL
- Constraint specification at the time of table creation
- Constraint specification after table creation
- Delete a constraint
- Disable a constraint
- Enable a constraint
- Viewing columns associated with constraints

Exercise:

1. Create following **department** table according to given data type and constraints:

Column name	Data type	Constraint
deptid	number(3)	primary key
dept_name	varchar(6)	only CSE, EEE, BBA, Eng, Ach allowed
budget	number(6)	default value 0

2. Create following **course** table according to givent data type and constraint:

Column name	Data type	Constraint
crs_id	number(4)	primary key
crs_name	varchar2(20)	not null
dept_id	number(3)	foreign key from department table

3. Create the table below according to given data types.

Column Name	Data Type
s_id	Number
s_name	Varchar2(20)
phone	Number
address	Varchar2(50)
email	Varchar2(30)
credit_completed	Number(3)
course_completed	Number(2)
cgpa	Number
deptno	number(5)
gender	Varchar2(6)

4. Set **s_id** as primary key of the table.
5. Set constraint not null on the column **s_name.**
6. Make **email** unique.
7. Make **deptno** as foreign key taking reference from **department** table which you have made in previous lab.
8. Add a constraint to **gender** so that it only allows the value **‘M’** and **‘F’**.
9. Disable the constraint of **s_id**.
10. Drop the constraint from **gender**.
11. View the columns associated with constraints.
12. Enable the constraint of **s_id**.

Lab 06: Lab Exam Guidelines

Objectives*:*

Students must now have an overall understanding of the lab based topics covered during the midterms. For this course the basics of SQL has been covered in details until now. During the final term more advanced topics of SQL will be taught in class.

Guidelines*:*

- Lab Exam is mandatory
- Copying/ Cheating is not allowed and if proved guilty will result in F grade for this course
- Lab Exam questions must be solved using Oracle 10g
- After solving write the SQLs in a text document
- The name of the text document MUST be your ID
- The SQLs MUST be numbered accordingly
- Upload the text document in the link provided in your VUES account