



United International University

Department of CSE

CSE 222: Database Management System Laboratory

Lab Sheet

Experiment No. 1

Basic SQL Queries

1. Introduction

In DBMS concept, the SELECT statement is used to select data from a database. The data returned is stored in a result table, called the result-set.

2. Lab Equipment/Required Software

- a. XAMPP (<https://www.apachefriends.org/download.html>)

3. Objective

- a. Get familiar with the XAMPP and MySQL.
- b. Learn how to use PHPMyAdmin.
- c. Learn how to do basic SELECT operation in MySQL.

4. Workflow

- a. Download necessary software if required (Section - 2).
- b. Download the sample MySQL file and import into PHPMyAdmin.
- c. Follow the examples of SQL queries shown in class.
- d. Practice the give problems in the lab sheet.

5. SQL Statements

[Ref: <https://www.mysqltutorial.org/>]

Clause	Variant	Example
Select	-	SELECT lastName FROM employees;
	Wildcard	SELECT * FROM employees;
	Without Table	SELECT NOW(); SELECT 2+3;
	DISTINCT	SELECT DISTINCT state FROM customers;
	Multiple DISTINCT	SELECT DISTINCT state, city FROM customers; [When you specify multiple columns in the DISTINCT clause, the DISTINCT clause will use the combination of values in these columns to determine the uniqueness of the row in the result set.]
Order by	-	SELECT contactLastname, contactFirstname FROM customers ORDER BY contactLastname;
	DESC ASC	SELECT contactLastname, contactFirstname FROM customers ORDER BY contactLastname DESC;
	Multiple Order	SELECT * FROM customers ORDER BY contactLastname DESC, contactFirstname ASC;
	Expression	SELECT orderNumber, orderlinenumber, quantityOrdered * priceEach FROM orderdetails ORDER BY quantityOrdered * priceEach DESC;
	Expression with Alias	SELECT orderNumber, orderlinenumber, quantityOrdered * priceEach AS subtotal FROM orderdetails ORDER BY subtotal DESC; [AS is optional]
	Limit	SELECT customerNumber,

		customerName FROM customers ORDER BY customerName LIMIT 10;
	Offset Limit	SELECT customerNumber, customerName FROM customers ORDER BY customerName LIMIT 20, 10;
Where	-	SELECT lastname, firstname, jobtitle FROM employees WHERE jobtitle = 'Sales Rep';
	AND	SELECT lastname, firstname, jobtitle, officeCode FROM employees WHERE jobtitle = 'Sales Rep' AND officeCode = 1;
	OR	SELECT lastName, firstName, jobTitle, officeCode FROM employees WHERE jobtitle = 'Sales Rep' OR officeCode = 1 ORDER BY officeCode , jobTitle;
	BETWEEN	SELECT firstName, lastName, officeCode FROM employees WHERE officeCode BETWEEN 1 AND 3 ORDER BY officeCode;
	LIKE	SELECT firstName, lastName FROM employees WHERE lastName LIKE '%son' ORDER BY firstName;

		[To form a pattern, you use the % and _ wildcards. The % wildcard matches any string of zero or more characters while the _ wildcard matches any single character.]														
	IN	<pre>SELECT firstName, lastName, officeCode FROM employees WHERE officeCode IN (1 , 2, 3) ORDER BY officeCode;</pre>														
	NOT IN	<pre>SELECT officeCode, city, phone FROM offices WHERE country NOT IN ('USA' , 'France') ORDER BY city;</pre>														
	NULL Check	<pre>SELECT lastName, firstName, reportsTo FROM employees WHERE reportsTo IS NULL;</pre>														
	Comparison Operator	<table><tr><th>Operator</th><th>Description</th></tr><tr><td>=</td><td>Equal to. You can use it with almost any data type.</td></tr><tr><td><> or !=</td><td>Not equal to</td></tr><tr><td><</td><td>Less than. You typically use it with numeric and date/time data types.</td></tr><tr><td>></td><td>Greater than.</td></tr><tr><td><=</td><td>Less than or equal to</td></tr><tr><td>>=</td><td>Greater than or equal to</td></tr></table>		Operator	Description	=	Equal to. You can use it with almost any data type.	<> or !=	Not equal to	<	Less than. You typically use it with numeric and date/time data types.	>	Greater than.	<=	Less than or equal to	>=
Operator	Description															
=	Equal to. You can use it with almost any data type.															
<> or !=	Not equal to															
<	Less than. You typically use it with numeric and date/time data types.															
>	Greater than.															
<=	Less than or equal to															
>=	Greater than or equal to															

Functions	String	<table><tr><th>Name</th><th>Description</th></tr><tr><td>CONCAT</td><td>Concatenate two or more strings into a single string</td></tr><tr><td>INSTR</td><td>Return the position of the first occurrence of a substring in a string</td></tr><tr><td>LENGTH</td><td>Get the length of a string in bytes and in characters</td></tr><tr><td>LEFT</td><td>Get a specified number of leftmost characters from a string</td></tr><tr><td>LOWER</td><td>Convert a string to lowercase</td></tr><tr><td>LTRIM</td><td>Remove all leading spaces from a string</td></tr><tr><td>REPLACE</td><td>Search and replace a substring in a string</td></tr><tr><td>RIGHT</td><td>Get a specified number of rightmost characters from a string</td></tr><tr><td>RTRIM</td><td>Remove all trailing spaces from a string</td></tr><tr><td>SUBSTRING</td><td>Extract a substring starting from a position with a specific length.</td></tr><tr><td>SUBSTRING_INDEX</td><td>Return a substring from a string before a specified number of occurrences of a delimiter</td></tr><tr><td>TRIM</td><td>Remove unwanted characters from a string.</td></tr><tr><td>FIND_IN_SET</td><td>Find a string within a comma-separated list of strings</td></tr><tr><td>FORMAT</td><td>Format a number with a specific locale, rounded to the number of decimals</td></tr><tr><td>UPPER</td><td>Convert a string to uppercase</td></tr></table>		Name	Description	CONCAT	Concatenate two or more strings into a single string	INSTR	Return the position of the first occurrence of a substring in a string	LENGTH	Get the length of a string in bytes and in characters	LEFT	Get a specified number of leftmost characters from a string	LOWER	Convert a string to lowercase	LTRIM	Remove all leading spaces from a string	REPLACE	Search and replace a substring in a string	RIGHT	Get a specified number of rightmost characters from a string	RTRIM	Remove all trailing spaces from a string	SUBSTRING	Extract a substring starting from a position with a specific length.	SUBSTRING_INDEX	Return a substring from a string before a specified number of occurrences of a delimiter	TRIM	Remove unwanted characters from a string.	FIND_IN_SET	Find a string within a comma-separated list of strings	FORMAT	Format a number with a specific locale, rounded to the number of decimals	UPPER	Convert a string to uppercase
		Name	Description																																
		CONCAT	Concatenate two or more strings into a single string																																
		INSTR	Return the position of the first occurrence of a substring in a string																																
		LENGTH	Get the length of a string in bytes and in characters																																
		LEFT	Get a specified number of leftmost characters from a string																																
		LOWER	Convert a string to lowercase																																
		LTRIM	Remove all leading spaces from a string																																
		REPLACE	Search and replace a substring in a string																																
		RIGHT	Get a specified number of rightmost characters from a string																																
		RTRIM	Remove all trailing spaces from a string																																
		SUBSTRING	Extract a substring starting from a position with a specific length.																																
		SUBSTRING_INDEX	Return a substring from a string before a specified number of occurrences of a delimiter																																
		TRIM	Remove unwanted characters from a string.																																
		FIND_IN_SET	Find a string within a comma-separated list of strings																																
	FORMAT	Format a number with a specific locale, rounded to the number of decimals																																	
	UPPER	Convert a string to uppercase																																	
	Math	<table><tr><th>Name</th><th>Description</th></tr><tr><td>ABS()</td><td>Returns the absolute value of a number</td></tr><tr><td>CEIL()</td><td>Returns the smallest integer value greater than or equal to the input number (n).</td></tr><tr><td>FLOOR()</td><td>Returns the largest integer value not greater than the argument</td></tr><tr><td>MOD()</td><td>Returns the remainder of a number divided by another</td></tr><tr><td>ROUND()</td><td>Rounds a number to a specified number of decimal places.</td></tr><tr><td>TRUNCATE()</td><td>Truncates a number to a specified number of decimal places</td></tr></table>		Name	Description	ABS()	Returns the absolute value of a number	CEIL()	Returns the smallest integer value greater than or equal to the input number (n).	FLOOR()	Returns the largest integer value not greater than the argument	MOD()	Returns the remainder of a number divided by another	ROUND()	Rounds a number to a specified number of decimal places.	TRUNCATE()	Truncates a number to a specified number of decimal places																		
		Name	Description																																
		ABS()	Returns the absolute value of a number																																
CEIL()		Returns the smallest integer value greater than or equal to the input number (n).																																	
FLOOR()		Returns the largest integer value not greater than the argument																																	
MOD()		Returns the remainder of a number divided by another																																	
ROUND()		Rounds a number to a specified number of decimal places.																																	
TRUNCATE()	Truncates a number to a specified number of decimal places																																		

Datetime	Function	Description
	CURDATE	Returns the current date.
	DATEDIFF	Calculates the number of days between two DATE values.
	DAY	Gets the day of the month of a specified date.
	DATE_ADD	Adds a time value to date value.
	DATE_SUB	Subtracts a time value from a date value.
	DATE_FORMAT	Formats a date value based on a specified date format.
	DAYNAME	Gets the name of a weekday for a specified date.
	DAYOFWEEK	Returns the weekday index for a date.
	EXTRACT	Extracts a part of a date.
	LAST_DAY	Returns the last day of the month of a specified date
	NOW	Returns the current date and time at which the statement executed.
	MONTH	Returns an integer that represents a month of a specified date.
	STR_TO_DATE	Converts a string into a date and time value based on a specified format.
	SYSDATE	Returns the current date.
	TIMEDIFF	Calculates the difference between two TIME or DATETIME values.
	TIMESTAMPDIFF	Calculates the difference between two DATE or DATETIME values.
	WEEK	Returns a week number of a date.
	WEEKDAY	Returns a weekday index for a date.
	YEAR	Return the year for a specified date

6. Hand's on Practice

✓✓Set-I

- ✓a. Write a query to display the names (first_name, last_name) using alias name "First Name", "Last Name"
- ✓b. Write a query to get unique department ID from employee table.
- ✓c. Write a query to get all employee details from the employee table order by first name, descending.
- ✓d. Write a query to get the full name (first_name + " " + last_name), salary, PF of all the employees (PF is calculated as 15% of salary).

✓✓Set-II

- ✓a. Write a query to get the employee ID, names (first_name, last_name), salary in ascending order of salary.
- ✓b. Write a query to select first 10 records.
- ✓c. Write a query to select the highest paid employee.
- ✓d. Write a query to select the second lowest paid employee.

✓✓Set-III

- ✓a. Write a query get all first name from employees table in upper case.
- ✓b. Write a query to get the length of the employee full name (first_name, last_name) from employees table.
- ✓c. Write a query to get the first 3 characters of first name from employees table.
- ✓d. Write a query to first name from employees who's name starts with 'B'.

Set-IV

- a. Write a query to first name from employees who's name starts with a vowel.
- b. Write a query to get the first name from employees table where the length of the first name is less than 5.

- c. Write a query to get the first name from employees table where the first name has the keyword 'on' in it.
- d. Write a query to get monthly salary (round 2 decimal places) of every employee.

7. Homework

- a. For more exercise, visit <https://www.w3resource.com/mysql-exercises>
- b. Complete the HW given in e-LMS.