# Practical: SQL – Basic SELECT statement

## Objectives

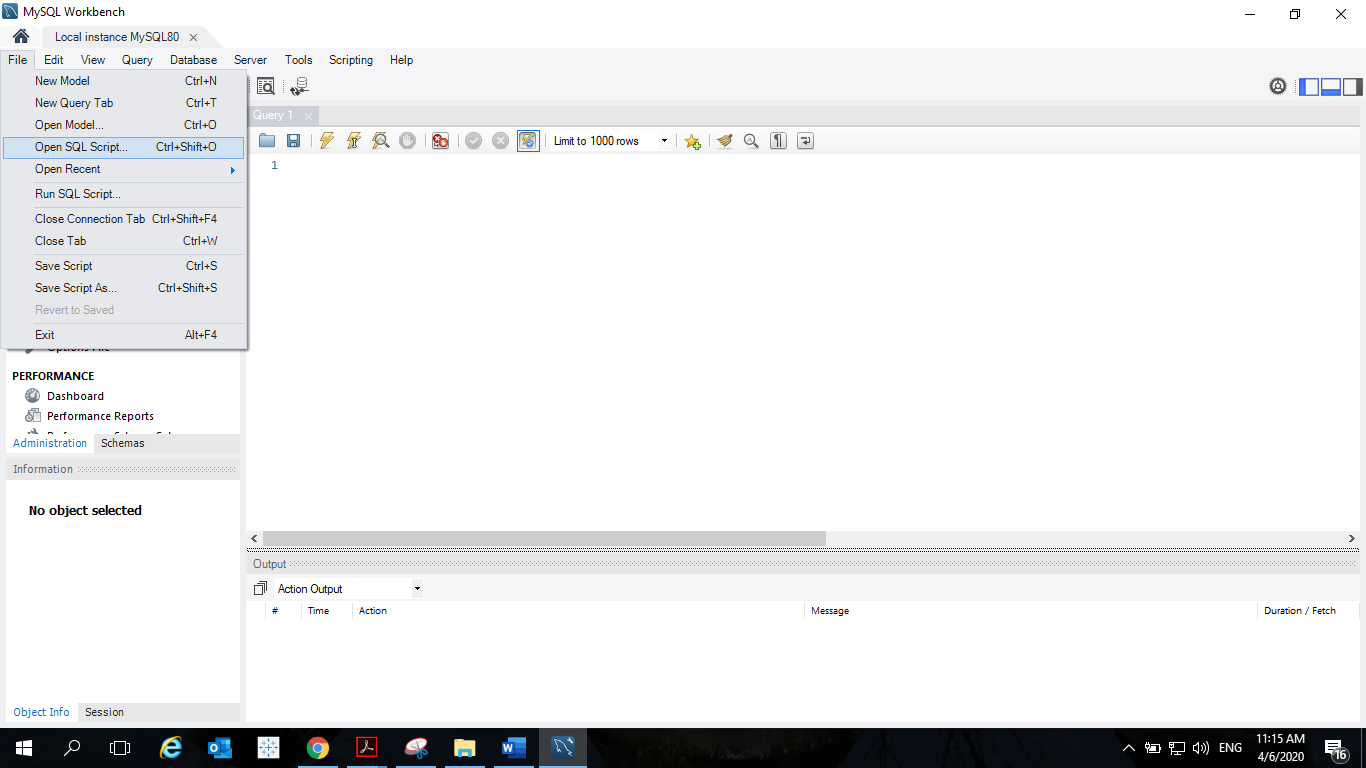
At the end of this practical, you should be able to:

* Connect to **MySQL Server** using **MySQL Workbench**.
* Issue basic **SELECT** statements to query the **ORDER** database.

## Tasks

### Setup the lab environment for P11, and P12

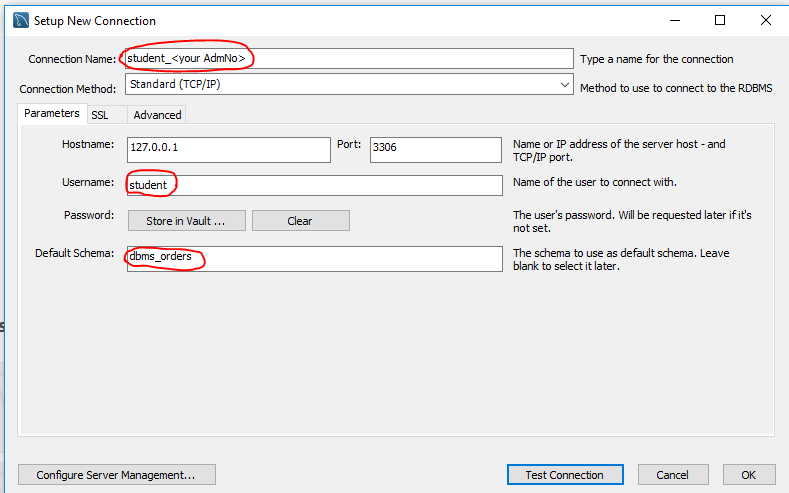
1. Download the script DBMS\_SQL\_SELECT\_Setup.sql from Blackboard Learning System.
2. Launch MySQL Workbench. Connect to the MySQL Server with MySQL Workbench using root account.
3. From the main menu bar, click File -> Open SQL Script. Navigate to Download folder open *DBMS\_SQL\_SELECT\_Setup.sql*



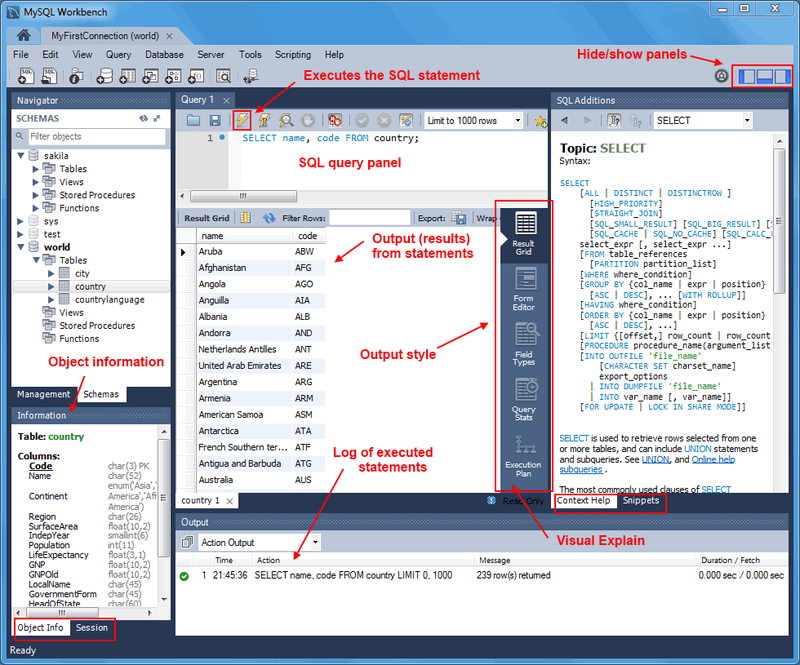
1. Run the script by clicking the button, or press Shift+Ctrl+Enter.   
   This script creates a user named *student* with password as *studPa55*. It also creates a database named *dbms\_orders*. Creates 6 tables and populates data to all the tables.



1. Create a Connection using *student* user account that access to *dbms\_orders* by default.
   * 1. Click the home button at the top left corner of MySQL Workbench.
     2. In the Welcome page, Next to MySQL Connections, click the + sign to create a new connection.
     3. In Setup New Connection dialog, enter the details as shown below and click **Test Connection**. When prompt, enter password for student: *studPa55*. If the connection is successfully made, click OK.



1. Click the *student\_<your Admno>* connection that you have just created.
2. You are now connected to the database *dbms\_orders* with *student* account. You can now enter your SQL statements in the **query** window and use this button to execute **single** query that the cursor is located.



1. Read [Visual SQL Editor Document](https://dev.mysql.com/doc/workbench/en/wb-sql-editor.html) to get yourself familiar with the MySQL Workbench.

### B. Querying the ORDER database

The **ORDER** database schema is given in page 5. This database consists of 7 tables and each table is populated with some sample records.

Syntax for the basic SELECT statement :

**SELECT** [**DISTINCT**] *select list*

**FROM** *tablename*

**{**[**INNER JOIN** *tablename* **ON** *condition*]}

[**WHERE** *condition*]

[**ORDER BY** *column list* [**DESC**]]

Write the SQL statements for the following queries (the underlined words give you hints on which table(s) to retrieve the required information):

**SELECT … FROM … ORDER BY …**

1. (i) Retrieve the full details of all the Orders. (2 methods) (Ans : 23 rows selected)

SELECT DISTINCT ORDER\_NUM FROM orders

SELECT DISTINCT PO\_NUM FROM orders

(ii) Study the output. In what order are the rows sorted ? (In ascending order of .)

ORDER\_NUM

(iii) Amend your SQL so that the output rows are arranged in ascending order of the Customer Number.

SELECT DISTINCT CUSTOMER\_NUM FROM orders ORDER BY CUSTOMER\_NUM ASC

1. (i) List the full name (i.e. fname and lname) and the full address (i.e. address1, address2, city, state\_code and zipcode) of all the Customer. (Ans : 28 rows selected)

SELECT DISTINCT FNAME, LNAME, address1, address2, city, state\_code, zipcode FROM customer

(ii) Study the output. What the values in the state\_code ? (Ans : CA, ) Hint: you may write another SQL to help you to retrieve the distinct values in state\_code.

'CA' 'NJ' 'AZ' 'DE' 'FL' 'OK' 'MA' 'CO' 'NY'

(iii) Amend your SQL in part (i) so that the output rows are arranged in ascending order of the lname.

SELECT DISTINCT FNAME, LNAME, address1, address2, city, state\_code, zipcode FROM customer ORDER BY lname ASC

**Computed/derived column, round( ) function**

1. The company has just announced a 10% increase in shipping charges for shipping the Orders.

(i) List the Order Number, the existing shipping charge (i.e. ship\_charge column) and the new shipping charge (derived from the existing shipping charge column) for all the Orders. Give the derived column a name (alias) called ‘new\_ship\_charge’. (23 rows selected)

SELECT DISTINCT order\_num, ship\_charge, round(ship\_charge \* 1.1, 2) AS new\_ship\_charge FROM orders

(ii) Amend your SQL to round the new shipping charge to the nearest dollars, i.e. 0 decimal place.

**S**ELECT DISTINCT order\_num, ship\_charge, round(ship\_charge \* 1.1, 0) AS new\_ship\_charge FROM orders

**Concatenation, substr( ) function**

1. (i) List the customer’s full name (i.e. concatenate the first name and the last name to form one field) of all the Customer. (28 rows selected)

SELECT DISTINCT CONCAT(fname, ‘ ’, lname) FROM customer

(ii) List the last 3 characters of the customer’s zip code. (Zip code is 5-characters.)

SELECT DISTINCT substr(zipcode, -3) FROM customer

**WHERE condition clause**

1. (i) Amend your SQL in question 2 (i) to retrieve only those customer *living in state ‘AZ’*. (Ans : 2 rows selected)

SELECT DISTINCT FNAME, LNAME, address1, address2, city, state\_code, zipcode FROM customer WHERE state\_code = 'AZ'

(ii) Retrieve full details of those Orders which *have not been paid yet*, i.e. Paid Date has NULL value. (Ans : 6 rows selected)

SELECT DISTINCT order\_num FROM orders WHERE paid\_date is NULL

(iii) Retrieve full details of those Products (from Product table) *supplied by supplier ‘HRO’*. (Ans : 12 rows selected)

SELECT DISTINCT prod\_num,suppl\_code, unit\_price, remarks FROM PRODUCT WHERE suppl\_code = 'HRO'

(iv) List the product number, product description (from Product\_desc table) of those products *related to ‘tennis’*, i.e. tennis racquet, tennis ball, etc. (Ans : 2 rows selected)

SELECT DISTINCT prod\_num, prod\_desc FROM product\_Desc WHERE prod\_desc LIKE 'tennis%'

(v) List supplier code, supplier name (from Supplier table) of those suppliers whose *name start with letter ‘H’*. (Hint : use substr( ) function in WHERE clause) (Ans : 2 rows selected)

SELECT DISTINCT suppl\_code, suppl\_name FROM supplier WHERE suppl\_name LIKE 'H%'

**Multiple Tables Queries (INNER JOIN .. ON)**

**For question 6 to 8, explore how you can add in an additional condition in the WHERE clause, to answer the questions prompted.**

1. One customer can place one or more orders. List the customer info (i.e. customer number, first name, last name) and his/her order info (i.e. order number, order date). (Ans : 23 rows selected)

Hint : Join Customer and Orders tables.

What are the orders placed by customer 104 (Anthony Higgins) ? (1001, , , )

SELECT DISTINCT c.customer\_num, fname, lname, order\_num, order\_date FROM customer c

INNER JOIN orders o ON c.customer\_num = o.customer\_num

WHERE c.customer\_num = 104

ORDER BY c.customer\_num;

1. One order can consists of one or more products. List the order info (i.e. order number, order date) and the products bought in the orders (list product number and quantity). (Ans : 67 rows selected)

Hint : Join Orders and Order\_detail tables.

What are the products bought in order 1022 ? (309, , )

SELECT DISTINCT o.order\_num, order\_date, quantity, prod\_num FROM orders o

INNER JOIN order\_detail od ON od.order\_num = o.order\_num

WHERE o.order\_num = 1022

ORDER BY o.order\_num;

1. One product (e.g. tennis ball) can be supplied by many suppliers (refer records in Product table). List the product info (product number, product description) and its supplier’s info (supplier code, supplier name). (Ans : 74 rows selected)

Hint : Join Product to Product\_desc, Supplier tables.

Who are the suppliers for ‘football’ ? (Hero, )

**C. OPTIONAL EXERCISE**

**For question 9 to 10, explore how you can add in an additional condition in the WHERE clause, to answer the questions prompted.**

**Multiple Tables Queries (SELF-JOIN)**

1. Some customers are referred by other customers. List the customer fullname (with alias as “Customer Name”) and the corresponding referral fullname (with alias as “Referral”) in ascending order of the customer fullname.

Hint : Join Customer to Customer tables (Self-join)

Which customers are referred by customer 102 (Carole Sadler)? (Jason Wallack, )

**Multiple Tables Queries (LEFT/RIGHT OUTER JOIN .. ON)**

1. The company marketing team wants to identify which states in the country they need to reach out in their upcoming marketing campaign. List all states, with any existing customers.

Which are the state names starting with ‘A’ where there are no customers yet?

(Alaska, , )

### APPENDIX: ORDER database schema

|  |  |
| --- | --- |
| **Column Name** | **Data Type** |
| order\_num | number(6) |
| item\_num | number(3) |
| prod\_num | number(3) |
| suppl\_code | varchar2(3) |
| quantity | number(3) |
| total price | number(9,2) |

|  |  |
| --- | --- |
| **Column Name** | **Data Type** |
| customer\_num | number(5) |
| fname | Varchar2(15) |
| lname | Varchar2(15) |
| company | Varchar2(20) |
| address1 | Varchar2(20) |
| address2 | Varchar2(20) |
| city | varchar2(15) |
| state\_code | varchar2(2) |
| zipcode | varchar2(5) |
| phone | varchar2(18) |
| referred\_by | number(5) |

**customer**

**orders**

**order\_detail**

|  |  |
| --- | --- |
| **Column Name** | **Data Type** |
| order\_num | number(6) |
| order\_date | Date |
| customer\_num | number(5) |
| ship\_instruct | varchar2(40) |
| backlog | varchar2(1) |
| po\_num | varchar2(10) |
| ship\_date | date |
| ship\_weight | number(9,2) |
| ship\_charge | number(9,2) |
| paid\_date | date |

**product\_desc**

**product**

|  |  |
| --- | --- |
| **Column Name** | **Data Type** |
| prod\_num | number(3) |
| suppl\_code | varchar2(3) |
| unit\_price | number(6,2) |
| remarks | varchar2(15) |

|  |  |
| --- | --- |
| **Column Name** | **Data Type** |
| prod\_num | number(3) |
| prod\_desc | varchar2(15) |

**state**

|  |  |
| --- | --- |
| **Column Name** | **Data Type** |
| state\_code | varchar2(2) |
| state\_name | varchar2(15) |

**supplier**

|  |  |
| --- | --- |
| **Column Name** | **Data Type** |
| suppl\_code | varchar2(3) |
| suppl\_name | varchar2(15) |
| lead\_time\_in\_days | number(3) |

Foreign key references

|  |
| --- |
|  |

Primary keys