COMS2002A: DATABASE FUNDAMENTALS

SEMESTER 1, 2024

LAB 4 – ADVANCED SELECT QUERIES

5th March, 2024

Lecturer: Mr. Mikhail Chirkoot

PRELIMINARIES

1. Connect to the LAMP server
2. Connect to MySQL
3. Choose the database to be used using the `USE` command.

Details on these steps are in the Lab1.pdf file.

PART 1 – PRACTICE

Last week, we looked at the SELECT command and how we can select all the data from a table. We also looked at how to select one or more columns of data.

In this lab, we'll look at how to select specific rows of data based on certain criteria. **A.**

SETTING UP OUR TABLES

To start with, we need to create and populate our tables with existing data.

1. Create a table called ***`student`*** with the following columns:

```
student_no char(8),
student_fname varchar(25),
student_lname varchar(25),
student_contact char(11));
```

2. Run the DESCRIBE command and verify the properties of the table you created.

```
mysql> desc student;
+----------------+-------------+------+-----+---------+-------+
| Field          | Type        | Null | Key | Default | Extra |
+----------------+-------------+------+-----+---------+-------+
| student_no     | char(8)     | YES  |     | NULL    |       |
| student_fname  | varchar(25) | YES  |     | NULL    |       |
| student_lname  | varchar(25) | YES  |     | NULL    |       |
| student_contact | char(11)   | YES  |     | NULL    |       |
+----------------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

3. Insert the following data (in the screenshot) into the table **`student`**. Run the SELECT command to verify the added data.

```
+------------+--------------+--------------+-----------------+
| student_no | student_fname | student_lname | student_contact |
+------------+--------------+--------------+-----------------+
| 11000010   | Henry        | Peter        | 0767263611      |
| 11000011   | Joe          | Sue          | 0738781982      |
| 11000012   | Jake         | Henry        | 0767136971      |
| 11000013   | Thabo        | Mbatha       | 0638220136      |
| 11000014   | Naledi       | Dlamini      | 0726268245      |
| 11000015   | Senzo        | Mokoena      | 0820436231      |
+------------+--------------+--------------+-----------------+
```

4. Create table called **course**, with the following columns and data type.
   `course_code char(8), course_name varchar(50), course_diagonal char(1), course_coordinator varchar (15)`

5. Describe the **course** table to see the details.

```
mysql> desc course;
+--------------------+-------------+------+-----+---------+-------+
| Field              | Type        | Null | Key | Default | Extra |
+--------------------+-------------+------+-----+---------+-------+
| course_code        | char(8)     | YES  |     | NULL    |       |
| course_name        | varchar(50) | YES  |     | NULL    |       |
| course_diagonal    | char(1)     | YES  |     | NULL    |       |
| course_coordinator | varchar(15) | YES  |     | NULL    |       |
+--------------------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

6. Insert the following data (in the screenshot) into the table **`course`**. Run the SELECT command to verify the added data.

```
mysql> SELECT * FROM course;
+-------------+----------------------+----------------+--------------------+
| course_code | course_name          | course_diagonal | course_coordinator |
+-------------+----------------------+----------------+--------------------+
| COMS2002    | Database Fundamentals | B              | Olaperi            |
| COMS2013    | Mobile Computing      | B              | Pravesh            |
+-------------+----------------------+----------------+--------------------+
2 rows in set (0.01 sec)
```

7. Create table called **registration**, with the following columns and data type.
   `course_code char(8), student_no char(8), registration_no char(3), registration_date date`
8. Describe the **registration** table to see the details.

```
mysql> DESC registration;
+-------------------+---------+------+-----+---------+-------+
| Field             | Type    | Null | Key | Default | Extra |
+-------------------+---------+------+-----+---------+-------+
| course_code       | char(8) | YES  |     | NULL    |       |
| student_no        | char(8) | YES  |     | NULL    |       |
| registration_no   | char(3) | YES  |     | NULL    |       |
| registration_date | date    | YES  |     | NULL    |       |
+-------------------+---------+------+-----+---------+-------+
4 rows in set (0.01 sec)
```

9. Insert the following data (in the screenshot) into the **`registration`** table. Run the SELECT command to verify the added data.

```
mysql> SELECT * FROM registration;
+-------------+------------+-----------------+-------------------+
| course_code | student_no | registration_no | registration_date |
+-------------+------------+-----------------+-------------------+
| COMS2002    | 11000010   | 111             | 2018-04-09        |
| COMS2002    | 11000011   | 113             | 2018-04-10        |
| COMS2002    | 11000012   | 115             | 2018-04-11        |
| COMS2013    | 11000010   | 112             | 2018-04-09        |
| COMS2013    | 11000011   | 114             | 2018-04-11        |
+-------------+------------+-----------------+-------------------+
5 rows in set (0.00 sec)
```

B. SELECTING ROWS WITH CONDITIONAL RESTRICTIONS – COMPARISON OPERATORS

Different comparison operators can be used with the WHERE clause in order to select certain rows. Read the course resource – ***Introduction to SQL – 4.pdf*** for more details on the comparison operators.

1. We can select a subset of a table by placing restrictions on the rows to be included in the output. For example, let's display any student whose last name is 'Henry'.

```
SELECT * FROM students
WHERE student_lname = 'Henry';
```

```
mysql> SELECT * FROM student WHERE student_lname = 'Henry';
+------------+---------------+---------------+-----------------+
| student_no | student_fname | student_lname | student_contact |
+------------+---------------+---------------+-----------------+
| 11000012   | Jake          | Henry         | 0767136971      |
+------------+---------------+---------------+-----------------+
1 row in set (0.00 sec)
```

2. By this command, we are selecting all columns of data from the table **student**, where the **student_lname** column is **Henry**. We are using the '=' (equal to) operator. Note the single equals '=' symbol.

3. Next, let's select all students where the student's last name is **not** Henry.

```
SELECT * FROM student WHERE student_lname != 'Henry';
```

```
mysql> SELECT * FROM student WHERE student_lname != 'Henry';
+------------+--------------+---------------+-----------------+
| student_no | student_fname | student_lname | student_contact |
+------------+--------------+---------------+-----------------+
| 11000010   | Henry        | Peter         | 0767263611      |
| 11000011   | Joe          | Sue           | 0738781982      |
| 11000013   | Thabo        | Mbatha        | 0638220136      |
| 11000014   | Naledi       | Dlamini       | 0726268245      |
| 11000015   | Senzo        | Mokoena       | 0820436231      |
+------------+--------------+---------------+-----------------+
5 rows in set (0.00 sec)
```

4. Remember, we can also select specific columns from the table. Next, let's select only the student's number and contact where the student number is '11000013'.

```
SELECT student_no, student_contact FROM student
WHERE student_no = '11000013';
```

```
mysql> SELECT student_no, student_contact FROM student
    -> WHERE student_no = '11000013';
+------------+-----------------+
| student_no | student_contact |
+------------+-----------------+
| 11000013   | 0638220136      |
+------------+-----------------+
1 row in set (0.00 sec)
```

5. Let's view all students who registered after the 9th of April, 2018. (2018-04-09). Here we will be using the greater than (>) symbol.

```
SELECT * FROM registration WHERE registration_date >
'2018-04-09';
```

```
mysql> SELECT * FROM registration WHERE registration_date > '2018-04-09';
+-------------+------------+-----------------+-------------------+
| course_code | student_no | registration_no | registration_date |
+-------------+------------+-----------------+-------------------+
| COMS2002    | 11000011   | 113             | 2018-04-10        |
| COMS2002    | 11000012   | 115             | 2018-04-11        |
| COMS2013    | 11000011   | 114             | 2018-04-11        |
+-------------+------------+-----------------+-------------------+
3 rows in set (0.00 sec)
```

C. SELECTING ROWS WITH CONDITIONAL RESTRICTIONS – LOGICAL OPERATOR Read the course resource – ***Introduction to SQL – 4.pdf*** for more details on the logical operators.

1. Next, let's use some logical operators. Select any student who registered into COMS2002 on the 9th of April, 2018. We will use the Logical AND operator here, since we need to check for two conditions to be met (both conditions).

```
SELECT * FROM registration
WHERE course_code = 'COMS2002' AND registration_date =
'2018-04-09';
```

```
mysql> SELECT * FROM registration
    -> WHERE course_code = 'COMS2002' AND registration_date = '2018-04-09';
+-------------+------------+-----------------+-------------------+
| course_code | student_no | registration_no | registration_date |
+-------------+------------+-----------------+-------------------+
| COMS2002    | 11000010   | 111             | 2018-04-09        |
+-------------+------------+-----------------+-------------------+
1 row in set (0.00 sec)
```

2. Select all students who registered into either COMS2002 or COMS2013. Here, we'll use the LOGICAL OR operator, because, we want students who registered for either of the courses.

```
SELECT * FROM registration WHERE course_code='COMS2002'
OR COURSE_CODE='COMS2013';
```

```
mysql> SELECT * FROM registration WHERE course_code='COMS2002' OR COURSE_CODE='COMS2013';
+-------------+------------+-----------------+-------------------+
| course_code | student_no | registration_no | registration_date |
+-------------+------------+-----------------+-------------------+
| COMS2002    | 11000010   | 111             | 2018-04-09        |
| COMS2002    | 11000011   | 113             | 2018-04-10        |
| COMS2002    | 11000012   | 115             | 2018-04-11        |
| COMS2013    | 11000010   | 112             | 2018-04-09        |
| COMS2013    | 11000011   | 114             | 2018-04-11        |
+-------------+------------+-----------------+-------------------+
5 rows in set (0.00 sec)
```

3. Now let's select all rows from the registration table where the course code is not COMS2002. We can use the LOGICAL NOT operator to negate any specified condition.

```
SELECT * FROM registration
WHERE NOT (course_code = 'COMS2002');
```

```
mysql> SELECT * FROM registration
    -> WHERE NOT (course_code = 'COMS2002');
+-------------+------------+-----------------+-------------------+
| course_code | student_no | registration_no | registration_date |
+-------------+------------+-----------------+-------------------+
| COMS2013    | 11000010   | 112             | 2018-04-09        |
| COMS2013    | 11000011   | 114             | 2018-04-11        |
+-------------+------------+-----------------+-------------------+
2 rows in set (0.00 sec)
```

4. Note that you can also use one of the comparison operators for this task. Try it!

D. SELECTING ROWS WITH CONDITIONAL RESTRICTIONS – SPECIAL OPERATORS

1. Read the course resource – ***Introduction to SQL – 4.pdf*** for more details on the special operators. We'll be using the special operators alongside other operators such as the logical operators.

2. There are several special operators that can be used to restrict the rows returned by the **SELECT** command. One of them is **LIKE**. Let's see an example of how it is used. Let's select all the details from the student table where the student's first name starts with the letter 'J'.

```
SELECT * FROM student WHERE student_fname LIKE 'J%';
```

```
mysql> SELECT * FROM student WHERE student_fname LIKE 'J%';
+------------+---------------+---------------+-----------------+
| student_no | student_fname | student_lname | student_contact |
+------------+---------------+---------------+-----------------+
| 11000011   | Joe           | Sue           | 0738781982      |
| 11000012   | Jake          | Henry         | 0767136971      |
+------------+---------------+---------------+-----------------+
2 rows in set (0.00 sec)
```

The '%' symbol is a wildcard character in SQL and is used for pattern matching. In this command, it is used to match any character after the letter M. It can match zero or more characters. That means we are searching for *student_fname* values that start with '**J'** and have zero or more characters after. The query returns two rows where the first names are Joe and Jake.

3. Another example, let's select all student's details where the last name starts with the letter 'M'.

```
mysql> SELECT * FROM student WHERE student_lname LIKE 'M%';
+------------+---------------+---------------+-----------------+
| student_no | student_fname | student_lname | student_contact |
+------------+---------------+---------------+-----------------+
| 11000013   | Thabo         | Mbatha        | 0638220136      |
| 11000015   | Senzo         | Mokoena       | 0820436231      |
+------------+---------------+---------------+-----------------+
2 rows in set (0.00 sec)
```

4. Next, let's select rows where the last name does not end with the letter 'a'.

```
SELECT * FROM student WHERE student_lname NOT LIKE '%a';
```

```
mysql> SELECT * FROM student WHERE student_lname NOT LIKE '%a';
+------------+---------------+---------------+-----------------+
| student_no | student_fname | student_lname | student_contact |
+------------+---------------+---------------+-----------------+
| 11000010   | Henry         | Peter         | 0767263611      |
| 11000011   | Joe           | Sue           | 0738781982      |
| 11000012   | Jake          | Henry         | 0767136971      |
| 11000014   | Naledi        | Dlamini       | 0726268245      |
+------------+---------------+---------------+-----------------+
4 rows in set (0.00 sec)
```

5. Another special operator is **BETWEEN**. Let us select all students whose student number is between 11000010 and 11000012.

```
SELECT * FROM student
WHERE student_no BETWEEN 11000010 AND 11000012;
```

```
mysql> SELECT * FROM student
    -> WHERE student_no BETWEEN 11000010 AND 11000012;
+------------+---------------+---------------+-----------------+
| student_no | student_fname | student_lname | student_contact |
+------------+---------------+---------------+-----------------+
| 11000010   | Henry         | Peter         | 0767263611      |
| 11000011   | Joe           | Sue           | 0738781982      |
| 11000012   | Jake          | Henry         | 0767136971      |
+------------+---------------+---------------+-----------------+
3 rows in set (0.00 sec)
```

6. We can also select rows, where the column values matches a set of values specified. We do this using the **IN** special operator. Let's display student registration details for the 9th and the 11th of April, 2018.

```
SELECT * FROM registration
WHERE registration_date IN ('2018-04-09', '2018-04-11');
```

```
mysql> SELECT * FROM registration
    -> WHERE registration_date IN ('2018-04-09', '2018-04-11');
+-------------+-----------+-----------------+-------------------+
| course_code | student_no | registration_no | registration_date |
+-------------+-----------+-----------------+-------------------+
| COMS2002    | 11000010  | 111             | 2018-04-09        |
| COMS2002    | 11000012  | 115             | 2018-04-11        |
| COMS2013    | 11000010  | 112             | 2018-04-09        |
| COMS2013    | 11000011  | 114             | 2018-04-11        |
+-------------+-----------+-----------------+-------------------+
4 rows in set (0.00 sec)
```

PART 2 – YOUR TASK

INSTRUCTIONS

1. Write the SQL commands to perform the following tasks. Upload all the files into the Gradescope assignment – **Lab 4**.
2. We'll be using the same tables as in Part 1 – Practice. Ensure that the data you have in the tables are similar to the screenshots below when you run the SELECT * command.

QUESTIONS

1. Using the student table, write an SQL query to select all students whose first name is not 'Thabo'. Save your SQL command in a file called **lab4_q1.txt**
2. Using the student table, write an SQL query to select all students whose last name ends with 'e' OR 'a'. Save your SQL command in a file called **lab4_q2.txt**
3. Using the student table, write an SQL query to select all students who registered on or after the 10th of April, 2018. Save your SQL command in a file called **lab4_q3.txt**
4. Using the registration table, write an SQL query to select only the student number of students who registered for COMS2002 on either the 9th or 10th of April, 2018. Save your SQL command in a file called **lab4_q4.txt**
5. Using the student table, write an SQL query to select all students who have the letter 'e' in their first name. Save your SQL command in a file called **lab4_q5.txt**
6. Using the student table, write an SQL query to select all students who BOTH have the letter 'e' in their first name and also have the letter 'a' in their last name. Save your SQL command in a file called **lab4_q6.txt**
7. Using the student table, write an SQL query to select all students who EITHER have the letter 'e' in their first name OR the letter 'a' in their last name. Save your SQL command in a file called **lab4_q7.txt**

```
mysql> select * from student;
+------------+--------------+--------------+-----------------+
| student_no | student_fname | student_lname | student_contact |
+------------+--------------+--------------+-----------------+
| 11000010   | Henry         | Peter         | 0767263611      |
| 11000011   | Joe           | Sue           | 0738781982      |
| 11000012   | Jake          | Henry         | 0767136971      |
| 11000013   | Thabo         | Mbatha        | 0638220136      |
| 11000014   | Naledi        | Dlamini       | 0726268245      |
| 11000015   | Senzo         | Mokoena       | 0820436231      |
+------------+--------------+--------------+-----------------+
6 rows in set (0.00 sec)

mysql> select * from registration;
+-------------+------------+-----------------+-------------------+
| course_code | student_no | registration_no | registration_date |
+-------------+------------+-----------------+-------------------+
| COMS2002    | 11000010   | 111             | 2018-04-09        |
| COMS2002    | 11000011   | 113             | 2018-04-10        |
| COMS2002    | 11000012   | 115             | 2018-04-11        |
| COMS2013    | 11000010   | 112             | 2018-04-09        |
| COMS2013    | 11000011   | 114             | 2018-04-11        |
+-------------+------------+-----------------+-------------------+
5 rows in set (0.00 sec)

mysql> select * from course;
+-------------+----------------------+----------------+--------------------+
| course_code | course_name          | course_diagonal | course_coordinator |
+-------------+----------------------+----------------+--------------------+
| COMS2002    | Database Fundamentals | B              | Olaperi            |
| COMS2013    | Mobile Computing      | B              | Pravesh            |
+-------------+----------------------+----------------+--------------------+
2 rows in set (0.00 sec)

mysql>
```