



National University of Sciences and Technology (NUST)
School of Electrical Engineering and Computer Science

Department of Computing

Name: Mahum Samar

CMS ID = 290647

CS220: Database Systems

Class: BSCS -9B

Lab 02: SQL Operators

Time: 2:00pm-5:00 pm

Lab Engineer: Miss Sadia Amir

CLO-2: Formulate SQL queries to retrieve information from a relational database.



Introduction

Structured Query Language (SQL) was developed at IBM San Jose Research Laboratory as a part of System R project. It is a declarative query language for querying a relational database. It also includes features for defining the structure of the data, for inserting and modifying data in the database, and for specifying security constraints. It is relational complete (it supports all six core relational algebra operations). SQL commands can be classified into three groups DDL, DML & DCL.

Objectives

After performing this lab students should be able to:

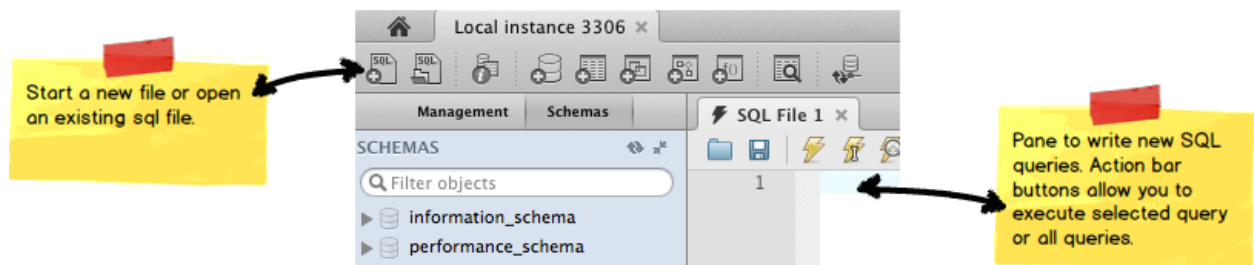
1. Design SQL queries to retrieve data using SELECT clause and various associated operators.
2. Design SQL queries with FROM & WHERE clause.
3. Execute SQL queries over MySQL using MySQL Workbench.

Tools/Software Requirement

- MySQL Community Server 5.6
- MySQL Workbench 6.1
- Sakila Database

Description

1. Open MySQL Workbench and open the default connection instance.
2. A new query window would open from where you can write and execute queries.



3. You can save the query file and can also add comments using # symbol.
4. On executing queries, results are displayed in the lower part of the screen.
5. Error or success messages are displayed in action output pane at the bottom.
6. Try running few SQL queries modeled during the lectures to get it going.
7. Continue playing with the Workbench and SQL queries till you are comfortable with the querying mechanism and have learnt the shortcuts to execute queries.



1. SQL Basic Query Structure

The SELECT clause

The most common use of the SQL commands is the selection of data from the tables located in a database. This can be achieved through SELECT command. We need to SELECT information FROM a table. Hence we have the most basic SQL query structure comprising of:

- SELECT
- FROM
- WHERE

The syntax for SELECT clause is as follows:

```
SELECT "column_name(s)" FROM "table_name(s)";
```

Always specify the name of the database in which a table/relation is present through which data is to be retrieved.

*e.g. Select column_name
 From Sakila.table_name*

There are three ways we can retrieve data from a table:

- Retrieve one column
- Retrieve multiple columns
- Retrieve all columns (Use *)

The select clause can contain arithmetic expressions involving the operation, +, −, *, and /, and operating on constants or attributes of records (tuples).

The FROM clause

The FROM clause can allow to select attributes from single table or multiple tables. When multiple tables are applied, it combines the records from the two or more tables listed and presents every possible combination of the listed attributes in SELECT clause.

This is not very useful until some filtering condition is applied. This is achieved through using the WHERE clause.

The WHERE clause

We can use the WHERE clause to filter the result set based on certain conditions. The syntax for using WHERE in the SELECT statement is as follows:

```
SELECT "column_name(s)"  
FROM "table_name(s)"  
WHERE "condition";
```

"Condition" can include a single comparison clause (called simple condition) or multiple comparison clauses combined together using AND or OR operators (compound condition). Conditions can include other operators like IN, BETWEEN, DISTINCT etc shown in Table 1:

Operator	Description	Example
=	Equal to	Author = 'Alcott'
<>	Not equal to (most DBMS also accept != instead of <>)	Dept <> 'Sales'
>	Greater than	Hire_Date > '2012-01-31'
<	Less than	Bonus < 50000.00
>=	Greater than or equal	Dependants >= 2
<=	Less than or equal	Rate <= 0.05



BETWEEN	Between an inclusive range	Cost BETWEEN 100.00 AND 500.00
LIKE	Match a character pattern	First_Name LIKE 'Will%'

Table 1: List of Basic Operators available in SQL

2. Ordering data

The order of rows returned in a query result is undefined. The ORDER BY clause can be used to sort the rows. This clause comes last in the SELECT statement. ASC at the end of the ORDER BY clause specifies ascending order where as DESC specifies descending order. ASC is the default order. The syntax for an ORDER BY statement is as follows:

```
SELECT "column_name"  
FROM "table_name"  
[WHERE "condition"]  
ORDER BY "column_name" [ASC, DESC];
```

3. The Wildcard operators

There are times when we want to match on a string pattern. To do that, we will need to employ the concept of wildcard. In SQL, there are two wildcards:

- % (percent sign) represents zero, one, or more characters.
- _ (underscore) represents exactly one character.

Wildcards are used with the LIKE keyword in SQL.

Below are some wildcard examples:

- 'A_Z': All string that starts with 'A', another character, and end with 'Z'. For example, 'ABZ' and 'A2Z' would both satisfy the condition, while 'AKKZ' would not (because there are two characters between A and Z instead of one).
- 'ABC%': All strings that start with 'ABC'. For example, 'ABCD' and 'ABCABC' would both satisfy the condition.
- '%XYZ': All strings that end with 'XYZ'. For example, 'WXYZ' and 'ZZXYZ' would both satisfy the condition.
- '%AN%': All strings that contain the pattern 'AN' anywhere. For example, 'LOS ANGELES' and 'SAN FRANCISCO' would both satisfy the condition.
- '_AN%': All strings that contain a character, then 'AN', followed by anything else. For example, 'SAN FRANCISCO' would satisfy the condition, while 'LOS ANGELES' would not satisfy the condition.

Lab Task

Using Sakila Database

Formulate SQL queries for the following needs and execute them on Sakila database.

1. Retrieve names of movies starting with S.

Code:

```
SELECT title FROM sakila.film WHERE title LIKE "s%";
```

Output:



National University of Sciences and Technology (NUST)

School of Electrical Engineering and Computer Science

MySQL Workbench

Lab02 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

sakila

- Tables
 - actor
 - address
 - category
 - city
 - country
 - customer
 - film
 - film_actor
 - film_category
 - film_text
 - inventory
 - language
 - payment
 - rental
 - staff
 - store
- Views
- Stored Procedures
- Functions
- sys

Query 1 x

Limit to 1000 rows

```
1 select title from sakila.film where title like "s%";
```

Result Grid

title
SABRINA MIDNIGHT
SADDLE ANTI TRUST
SAGEBRUSH CLUELESS
SAINTS BRIDE
SALUTE APOLLO
SAMURAI LION
SANTA PARIS
SASSY PACKER
SATISFACTION CONFIDENTIAL
SATURDAY LAMBS
SATURN NAME

Information

Table: film

Columns: film_id

Output

Action Output

#	Time	Action	Message
7	15:10:58	select title from sakila.film where title like "s%" LIMIT 0, 1000	119 row(s) returned

2. Select release year of movies starting with M?

Code:

SELECT release_year FROM sakila.film WHERE title LIKE "m%";

Output:



- Retrieve data of all actors whose names are not ending on T.

```
SELECT last_name FROM sakila.actor WHERE last_name NOT LIKE "%t";
```

CS220: Database Systems



National University of Sciences and Technology (NUST)

School of Electrical Engineering and Computer Science

MySQL Workbench

Lab02 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

sakila

Tables

- actor
- address
- category
- city
- country
- customer
- film
- film_actor
- film_category
- film_text
- inventory
- language
- payment
- rental
- staff
- store

Views

Stored Procedures

Functions

sys

Query 1 x

```
1 select last_name from sakila.actor where last_name not like "%t";
```

Result Grid

last_name
AKROYD
AKROYD
AKROYD
ALLEN
ALLEN
ALLEN
ASTAIRE
BACALL
BAILEY
BAILEY
BALE
...

actor 15 x

Information

Table: actor

Columns:

- actor_id smallint UN
- first_name AI PK
- last_name varchar(45)
- last_update varchar(45) timestamp

Object Info Session

Query Completed

Output

#	Time	Action	Message
12	15:11:01	select title from sakila.film where title like "s%" LIMIT 0, 1000	119 row(s) returned
13	15:13:13	select release_year from sakila.film where title like "m%" LIMIT 0, 1000	71 row(s) returned
14	15:20:25	select last_name from sakila.actor where last_name like "l%" LIMIT 0, 1000	0 row(s) returned
15	15:21:15	select last_name from sakila.actor where last_name not like "m%" LIMIT 0, 1000	185 row(s) returned
16	15:21:36	select last_name from sakila.actor where last_name not like "t%" LIMIT 0, 1000	187 row(s) returned
17	15:21:43	select last_name from sakila.actor where last_name not like "a" LIMIT 0, 1000	187 row(s) returned

4. List one or more languages in which any movie is available.

Code:

```
SELECT name FROM sakila.language, sakila.film WHERE  
language.language_id = film.language_id;
```

Output:



```
SELECT name FROM sakila.category, sakila.film, sakila.film_category
WHERE
film.film_id = film_category.film_id
AND
category.category_id = film_category.category_id
AND sakila.film.title = "ADAPTATION HOLES" ;
```

CS220: Database Systems



MySQL Workbench

Lab02 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

sakila

Tables

- actor
- address
- category
- city
- country
- customer
- film
- film_actor
- film_category
- film_text
- inventory
- language
- payment
- rental
- staff
- store

Views

Stored Procedures

Functions

sys

Administration Schemas

Information

Table: film_category

Columns:

- film_id smallint UN PK
- category_id tinyint UN PK
- last_update timestamp

Object Info Session

Query 1 x

```
1 • select name from sakila.category, sakila.film, sakila.film_category
2 where
3 film.film_id = film_category.film_id
4 and
5 category.category_id = film_category.category_id
6 and sakila.film.title = "ADAPTATION HOLES" ;
7
8
```

Result Grid

name
Documentary

Result 25 x

Read Only

Output

#	Time	Action	Message
35	15:55:49	select name from sakila.category, sakila.film, sakila.film_category w...	1 row(s) returned
36	15:56:43	select * from sakila.film LIMIT 0, 1000	1000 row(s) returned
37	15:57:00	select * from sakila.film where title = "ADAPTATION HOLES" LIM...	1 row(s) returned
38	15:57:32	select * from sakila.film_category where film_id = 3 LIMIT 0, 1000	1 row(s) returned
39	15:57:55	select * from sakila.category where category_id = 6 LIMIT 0, 1000	1 row(s) returned
40	15:58:07	select name from sakila.category, sakila.film, sakila.film_category w...	1 row(s) returned

6. Retrieve first and last name of actors who played in ALONE TRIP.

Code:

```
SELECT first_name, last_name FROM sakila.actor, sakila.film_actor, sakila.film
WHERE
actor.actor_id = film_actor.actor_id
AND
film_actor.film_id = film.film_id
AND
sakila.film.title = "ALONE TRIP";
```

Output:



MySQL Workbench

Lab02 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

sakila

Tables

- actor
- address
- category
- city
- country
- customer
- film
- film_actor
- film_category
- film_text
- inventory
- language
- payment
- rental
- staff
- store

Views

Stored Procedures

Functions

sys

Administration Schemas

Information

Table: actor

Columns:

- actor_id smallint UN AI PK
- first_name varchar(45)
- last_name varchar(45)
- last_update timestamp

Object Info Session

Query 1 x

```
1 • select first_name, last_name from sakila.actor, sakila.film_actor, sakila.film
2 where
3 actor.actor_id = film_actor.actor_id
4 and
5 film_actor.film_id = film.film_id
6 and
7 sakila.film.title = "ALONE TRIP";
8
```

Result Grid

first_name	last_name
ED	CHASE
KARL	BERRY
UMA	WOOD
WOODY	JOLIE
SPENCER	DEPP
CHRIS	DEPP
LAURENCE	BULLOCK
RENEE	BALL

Result 30 x

Read Only

Output

Action Output

#	Time	Action	Message
✓ 40	15:58:07	select name from sakila.category, sakila.film, sakila.film_category w...	1 row(s) returned
✓ 41	16:09:18	select first_name, last_name from sakila.actor, sakila.film_actor, sak...	8 row(s) returned
✓ 42	16:10:04	select * from film where title = "ALONE TRIP" LIMIT 0, 1000	1 row(s) returned
✓ 43	16:10:51	select * from film_actor where film_id = 17 LIMIT 0, 1000	8 row(s) returned
✓ 44	16:11:13	select * from film where title = "ALONE TRIP" LIMIT 0, 1000	1 row(s) returned
✓ 45	16:11:55	select first_name, last_name from sakila.actor, sakila.film_actor, sak...	8 row(s) returned

Deliverable

Submit a PDF document including the SQL queries to answer above-mentioned information needs as well as snapshot of their outcome when executed over MySQL using the Workbench.