Department of Computer Engineering
Faculty of Engineering
University of Sri Jayewardenepura

Course	Database Systems
Course Code	CO3201
Title	Introduction to SQL
Practical Number	1
Outcomes	Gain knowledge about the basics of SQL

General Instructions

- No food, drinks, backpacks, and bags are allowed to take inside the laboratory.
- Login to the computer using the 'Student' account.
- Create a folder in the D drive to save your work and name it according to the following format: *yy_ENG_xxx* (e.g. 17_ENG_135).
- Use the following format when you are naming the source files: *yy_ENG_xxx_L_n.out*, *yy_ENG_xxx* is your registration number, *L* stand for the practical number, and *n* represent the exercise number (e.g. 17_ENG_135_3_1.out)
- Please save your work frequently during the practical session to avoid data lose due to unavoidable circumstances.
- Your files will be erased after the practical session. Therefore, please keep a backup to yourself.
- Please archive all ".out" files to a zip file, upload the zip file to LMS.

What is SQL ?

SQL is a standard language for sorting, manipulating and retrieving data in databases.

What Can SQL do?

SQL can execute queries against a database
SQL can retrieve data from a database
SQL can insert records in a database SQL
can update records in a database SQL
can delete records from a database SQL
can create new databases
SQL can create new tables in a database

Data types

Text data type:

VARCHAR - A variable-length string between 1 and 255 characters in length. For example, VARCHAR(25). You must define a length when creating a VARCHAR field.

CHAR - A fixed-length string between 1 and 255 characters in length (for example CHAR(5)), right-padded with spaces to the specified length when stored. Defining a length is not required, but the default is 1.

Number data type:

INTEGER - A normal-sized integer that can be signed or unsigned. If signed, the allowable range is from -2147483648 to 2147483647. If unsigned, the allowable range is from 0 to 4294967295. You can specify a width of up to 11 digits.

FLOAT - A floating-point number that cannot be unsigned. You can define the display length (M) and the number of decimals (D). This is not required and will default to 10,2, where 2 is the number of decimals and 10 is the total number of digits (including decimals). Decimal precision can go to 24 places for a FLOAT.

DOUBLE - A double precision floating-point number that cannot be unsigned. You can define the display length (M) and the number of decimals (D). This is not required and will default to 16,4, where 4 is the number of decimals. Decimal precision can go to 53 places for a DOUBLE. REAL is a synonym for DOUBLE.

Date data type:

DATE() - A date in YYYY-MM-DD format, between 1000-01-01 and 9999-12-31. For example, December 30th, 1973 would be stored as 1973-12-30.

TIMESTAMP() - A timestamp between midnight, January 1st, 1970 and sometime in 2037. This looks like the previous DATETIME format, only without the hyphens between numbers; 3:30 in the afternoon on December 30th, 1973 would be stored as 19731230153000 (YYYYMMDDHHMMSS).

Basic Commands

1. **To log the MysQL client session and create a log file:**

   Syntax : mysql> tee <destination>/<file_name>.out;
   Ex : mysql> tee D:/practical/Lab_01.out;

2. **Create a database**

   Syntax :CREATE DATABASE <databaseName>;
   Ex :CREATE DATABASE student;

3. **Use database**

   Syntax : USE <*databaseName>;*
   Ex : USE student;

4. **Show tables in a database**

   Syntax : SHOW TABLES ;

5. **Delete database**

   Syntax :

   i. DROP DATABASE <*databaseName*> ; *(Delete the database (irrecoverable!))*

   ii. DROP DATABASE IF EXISTS <*databaseName*> ; *(Delete if it exists)*

   Ex :DROP DATABASE student; or DROP DATABASE IF EXISTS student ;

6. **Create table with different data types**

   i. Without primary key/foreign key

   Syntax : CREATE TABLE <*table_name*> (
   <*column1 datatype*>, <*column2 datatype*>,<*column3 datatype*>,..);

   Ex: CREATE TABLE Persons
   (PersonID int, LastName varchar(255), FirstName varchar(255),
   Address varchar(255), City varchar(255));

   ii. With primary key/foreign key

   Syntax :

   CREATE TABLE <*tableName*> (<*columnName columnType columnAttribute*, ...>
   PRIMARY KEY(<*columnName*>), FOREIGN KEY (<*columnNmae*>)
   REFERENCES <tableName> (<columnNmae>));
   Ex : CREATE TABLE student

```
(
    id INT unsigned NOT NULL AUTO_INCREMENT,
    name    VARCHAR(150) NOT NULL,
    course    VARCHAR(150) NOT NULL,
    birthday DATE NOT NULL,
    PRIMARY KEY         (id)
);
```

7. Alternate table structure

i. Add column

Syntax : ALTER TABLE <*table_name*> ADD <*column_name*> <*datatype*>;

EX: ALTER TABLE Customers ADD Email varchar(255);

ii. Delete a column

Syntax :ALTER TABLE <table_name> DROP COLUMN

<column_name>; Ex : ALTER TABLE Customers DROP COLUMN Email;

iii. Modify column

Syntax : ALTER TABLE <*table_name*> MODIFY COLUMN <*column_name*> datatype;

Ex : ALTER TABLE student MODIFY COLUMN gender CHAR;

iv. Add primary key

Syntax : ALTER TABLE <table name> ADD PRIMARY KEY (<column name>);

EX : ALTER TABLE Persons ADD PRIMARY KEY (ID)

v. Add foreign key

Syntax : ALTER TABLE <table name> ADD FOREIGN KEY (column name) REFERENCES <reference table>(<reference table primary key column name>);

Ex: ALTER TABLE Orders
ADD FOREIGN KEY (PersonID) REFERENCES Persons(PersonID);

8. Describe table

Syntax : DESCRIBE <table name>;
Ex :DESCRIBE student;

9. Insert data

a. Using insert into command with correct sequence of columns

Syntax :INSERT INTO <*tablename*> VALUES <*...Values in order >*

Ex : INSERT INTO student VALUES (1,'Sandy', 'Lennon', '2015-01-03'), (2,'Cookie', 'Casey', '2013-11-13'),(3,'Charlie', 'River', '2016-05-21');

b. Using insert into command with specific columns in a table

Syntax :INSERT INTO *<tablename> <specific column name>* VALUES *<…Values in order >;*

Ex : INSERT INTO student (course,name, DOB) VALUES ('Lennon', 'Sandy', '2015-01-03'), ('Casey', 'Cookie','2013-11-13');

c. Using insert in to command with TEXT file

Syntax:

i. LOAD DATA LOCAL INFILE ' <file path>' INTO TABLE <table name> COLUMNS TERMINATED BY '\t';

Ex : LOAD DATA LOCAL INFILE ' C :\studentTb.txt ' INTO TABLE student COLUMNS TERMINATED BY '\t';

ii. BULK INSERT <table name> FROM <file path> WITH (FIELDTERMINATOR=',', ROWTERMINATOR='\n') GO;

Ex: BULK INSERT student FROM 'C :\stuDb.txt' WITH (FIELDTERMINATOR=',', ROWTERMINATOR='\n') GO;

10. Delete data in a table

a. Delete all data in table

Syntax:DELETE FORM <table name>;

Ex: DELETE FROM student;

b. Delete specific data in table

Syntax:DELETE FORM <table name> WHERE <column name>=<specific value>;

Ex: DELETE FROM student WHERE id=1;

11. Update data in a table

Syntax:UPDATE <table name> SET <column1>=<value1>, <column2>=<value2>,… WHERE <condition>;

Ex: UPDATE student SET name='Adam' WHERE id=1;

12. Drop table

Syntax:DROP TABLE <table name>;

Ex:DROP TABLE student;

13. Select data in a table

a. All columns

Syntax: SELECT * FROM <table name>;

Ex: SELECT * FROM student;

b. Specific column

Syntax: SELECT <column1>,<column2>,… FROM <table name>;

Ex: SELECT course,name FROM student;

c. Distinct

Syntax: SELECT DISTINCT <column name> FROM <table name>;

Ex: SELECT DISTINCT course FROM student;

d. Where

Syntax: SELECT <column name> FROM <table name>;

Ex: SELECT DISTINCT course FROM student;

e. AND, OR, NOT

Syntax: SELECT <specific column> FROM <table name>
WHERE <condition1> AND <condition2>

Ex: SELECT name,birthday FROM student WHERE id > 10 AND course ='IT';

f. Order by

Syntax: SELECT <specific column> FROM <table name> ORDER BY id DESC;

Ex: SELECT name,birthday FROM student ORDER BY id DESC;

g. Minimum and maximum

Syntax: SELECT MIN(<specific column>) FROM <table name>;

Ex: SELECT MIN(salary) FROM;

h. Like

Syntax: SELECT <specific column> FROM <table name> WHERE <column name> LIKE <patern>;

Ex: SELECT name,course,birthday FROM student WHERE name LIKE %a%;

i. Group by

Syntax : SELECT *<column_name>* FROM *<table_name>*
GROUP BY *<column_name>;*
Ex : SELECT name FROM student
GROUP BY faculty;

j. Having

Syntax : SELECT *<column_name>* FROM *<table_name>*
GROUP BY *<column_name>*
HAVING *<condition>;*
Ex : SELECT name FROM student
GROUP BY faculty
HAVING age > 17;

k. Between

Syntax: SELECT <specific column> FROM <table name> WHERE <condition1> AND <condition2>;

Ex: SELECT name,birthday FROM student WHERE age BETWEEN 5 AND 15;

Question: 01

The following are the requirements for the ABC University. Consider the requirements to create a Database for the university.

Note : save your sql queries in following file format using 'tee' command.

Ex : <Index number>.out

Part 1 – Creating a Database

1. Create a New database "ABC_Uni ".

2. View all the available databases in the server.

3. Select the newly created "ABC_Uni "database to implement other details given below.

Part 2 – Creating tables

4. University needs to save data regarding faculties, lecturers, students and courses. Create following tables in order to save data belongs to each.

5. Show table structure of each table created in the database.

6. Add primary key and foreign keys for tables.

Faculties

Faculty ID	**Faculty name**	**Location**
1	Medicine	Gampaha
2	Engineering	Ratmalana
3	Science	Kotte
4	Technology	Nugegoda
5	Business and finance	Maharagama

Lecturers

Lecturer ID	Title	Lecturer name	FacultyID	Email
10317	Prof	Nandana	3	nandana @sci.abc.lk
10318	Dr	Perera	2	perera @eng.abc.lk
10319	Mr	Asanka	1	asanka @med.abc.lk
10320	Ms	Nipuni	4	nipuni @ar.abc.lk
10321	Ms	Gunasekara	1	gunasekara @med.abc.lk
10322	Dr	Disanayake	4	disanayake @.ar.abc.lk

Students

Student Id	FName	Gender	DOB	Age	FacultyID	Town	Email
1000	Kamal	M	2000-01-10	18	003	Galle	kamal@mail.com
1001	Amal	M	1999-12-25	19	002	Kandy	amal@mail.com
1002	Nimal	M	1997-04-01	21	001	Jaffna	nimal@yao.com
1003	Amali	F	1998-02-14	20	004	Kandy	amali@mail.com
1004	Shamal	M	1997-04-14	21	003	Matara	sham97@yao.com
1005	Nimali	F	1996-02-29	22	002	Colombo	nimmi@yao.com

Courses

CourseID	Course name	FacultyID	noCredits	Semester
M001	Pharmacology	1	4	4
E001	Electronics	2	3	1
S001	Organic	3	3	1
A001	Literature	4	3	2
E002	Robotics	2	4	6
M002	Psychology	1	2	5
S002	Mathematics	3	4	3
E004	Computing	2	4	5
B001	Commerce	5	2	1
S002	Programing	3	3	3
S003	Statistics	3	4	3

Part 3 – Inserting Data into tables and table modifications

7. Insert the data in the tables as given above. Use text file to insert "Student" and "Course" data into tables.(create a text file using **tab** as a column separator)
8. Update the Location of 'Medicine' faculty in Faculty table as 'Gangodawila'.
9. Modify Lecture table by inserting new column called "salary" with float data type and describe the table structure.
10. Add following salaries into Lecturer table.
    a. Lecturer ID =10317 → salary = 120000.00
    b. Lecturer ID =10318 → salary = 115000.00
    c. Lecturer ID =10319 → salary = 100000.00
    d. Lecturer ID =10320 → salary = 100000.00
    e. Lecturer ID =10321 → salary = 100000.00
    f. Lecturer ID =10322 → salary = 115000.00
11. Delete 'age' column in student table.
12. Modify student table by inserting 'Last_name' column.
13. Update following students details.
    a. Student Id = 1000 →Last_name = 'Ranasinghe'
    b. Student Id = 1001 →Last_name = 'Silva'
    c. Student Id = 1002 →Last_name = 'Fernando'
    d. Student Id = 1003 → Last_name = 'Peris'
    e. Student Id = 1004→ Last_name = 'Joshep'
    f. Student Id = 1005→Last_name = 'Alwis'

14. Insert following student's details to Student table

    Madushi Rnasinghe studying in Technology faculty. Her DOB is 1996-03-15 and lives in Wattala. Email address is madushi@gmail.com

15. Write SQL queries to retrieve following data from the above tables.

I. View all the details in Lecturer table
II. View names of all the lecturers in the lecturer table
III. View all the student's full names and the faculty from the student table
IV. View all the salaries in ascending order
V. View all the lecturers who works in the medical faculty
VI. View the lecturer ID of the lecturer who teaches Robotics
VII. View all courses with less than 3 credits
VIII. View the names of the students who come from Galle
IX. View all the lecturers who work in the medical faculty
X. View the count of the students in each faculty with the faculty name and faculty ID
XI. View courses offering in semester 3 and 5.
XII. Find maximum and minimum salary of lecturers.
XIII. View all the lecturers who having salary between 100000 and 120000.
XIV. View all the courses group by number of credits allocation.
XV. View all the student last names start with letter 'A'.
XVI. Delete a student who has studentID = 1004.