DHA SUFFA UNIVERSITY
Department of Computer Science

CS-2003 Database Systems Lab
Spring 2022

LAB 01 Introduction to Databases

OBJECTIVE(S)

- Introduction to RDBMS and SQL
- Learn MySQL Data types
- Learn DDL and DML Commands

RELATIONAL DATABASE MANAGEMENT

A relational Database Management **S**ystem is a type of database management system (DBMS) that stores data in the form of related tables.

Popular RDBMS:

- Oracle
- MySQL
- Microsoft SQL Server

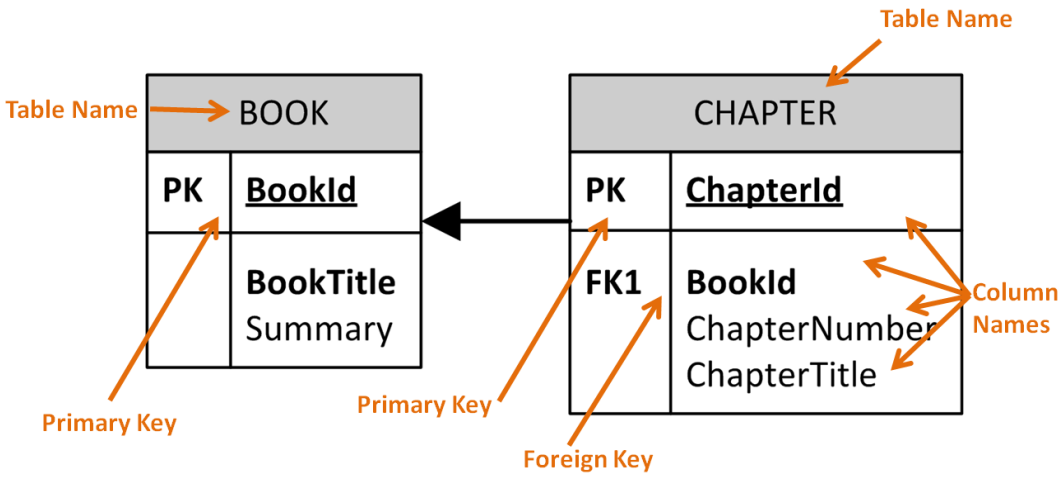
Tables

Customer Table:

CustomerID	CustomerName	Address	City	PostalCode	Country
1	Alan Turing	13 Guild Street	London	SE4 2FZ	United Kingdom
2	Dennis Ritchie	61 Farnum Road	New York	10033	United States
3	John von Neumann	Csabai kapu 4	Budapest	1033	Hungary
4	Ada Lovelace	68 Crown Street	London	SW1W 8WL	United Kingdom
5	Bjorn Stroustrup	Askelund 25	Copenhagen	1105	Denmark
6	Hideo Kojima	237-1268, Shimoigusa, Suginami-ku	Tokyo	167-0022	Japan

Database Relationship

Book has a Chapter:



STRUCTURED QUERY LANGUAGE (SQL)

Structured **Q**uery **L**anguage lets you access and manipulates databases.

DATA TYPES

DATA TYPE	DESCRIPTION
VARCHAR(size)	Holds a variable-length string (can contain letters, numbers, and special characters). The maximum size is specified in parenthesis. Can store up to 255 characters.
INT(size)	-2147483648 to 2147483647 normal. 0 to 4294967295 UNSIGNED*. The maximum number of digits may be specified in parenthesis
FLOAT(size, d)	A small number with a floating decimal point. The maximum number of digits may be specified in the size parameter. The maximum number of digits to the right of the decimal point is specified in the d parameter
DATE	A date. Format: YYYY-MM-DD

DATA DEFINITION LANGUAGE (DDL)

DDL refers to **D**ata **D**efinition **L**anguage, a subset of SQL statements that change the structure of the database schema in some way, typically by creating, deleting, or modifying schema objects such as databases, tables, and views.

CREATE:

Creates a new database, table, view of a table, or object in the database.

- **CREATE DATABASE** db_name;
  - **SHOW** databases;
  - **USE** db_name;

- **CREATE TABLE** tb_name(col1_name datatype(size), col2_name datatype(size));
  - **SHOW** tables;
  - **DESCRIBE** tb_name;

ALTER:

Modifies an existing database object such as a table.

- **ALTER TABLE** tb_name **CHANGE** old_col_name new_col_name datatype(size), **CHANGE** old_col_name new_col_name datatype(size);

- **ALTER TABLE** tb_name **MODIFY** col_name datatype(size), **MODIFY** col_name datatype(size)

- **ALTER TABLE** tb_name **ADD** col_name datatype(size), **ADD** col_name datatype(size);

- **ALTER TABLE** tb_name **DROP** col_name, **DROP** col_name;

DROP:

Delete an entire database, table, view of a table, or object in the database.

- **DROP DATABASE** db_name;

- **DROP TABLE** tb_name;

RENAME:

Renames an object in the database.

- **RENAME TABLE** old_tb_name **TO** new_tb_name;

TRUNCATE:

Removes all the records from a database table.

- **TRUNCATE TABLE** tb_name;

TASK

- Create a database studentDB.
- Display all the available databases.
- Select the created database (studentDB) to perform further operations on it.
- Create a table having the following fields and data types:
  stud_id (integer, 3), stud_name (varchar, 50), stud_phone (integer, 7)
  stud_gpa (integer, 1)
- Display all the tables in studentDB.
- Show the structure of the created table.
- Rename the table to "student". Display a list of all the tables in the database.
- Rename "stud_id" to "id".
- Rename "stud_name" to "name", "stud_phone" to "phone", "stud_gpa" to "gpa". Also, change the size of "stud_phone" to 10 and the data type of "stud_gpa" to float having size=3 and d=2. Use only one query.
- Show the structure of the table.
- Change the size of "name" to 100. Show the structure of the table.
- Delete the column "phone". Show the structure of the table.
- Add the following new columns using a single query. Show the structure of the table.
  Dob (date), semester (integer, 1), department (varchar, 2)
- Delete "phone" and "gpa" using a single query. Show the structure of the table.

DATA MANIPULATION LANGUAGE (DML)

DML refers to **D**ata **M**anipulation **L**anguage, a subset of SQL statements that deals with data manipulation.

SELECT:

Retrieves (reads) data from the table.

- **SELECT** * **FROM** tb_name;

INSERT:

Adds data into the database table.

- **INSERT INTO** tb_name(field(s)_name) **VALUES** (field(s)_data);

UPDATE:

Updates an existing record in a table.

- **UPDATE** tb_name **SET** col1_name = value1, col2_name = value2,
  WHERE col_name = col_value;

DELETE:

Deletes data from a table.

- **DELETE FROM** tb_name;
- **DELETE FROM** tb_name **WHERE** col_name = col_value;

TASK

- Insert 4 records into the table using a single query.
- Display all the records.
- Display only the second record.
- Delete the third record. Display all the records.
- Add only the id, name, and date of birth for two new records. Display all the records.
- Update the remaining fields for the two most recently added records.
- Display all the records.
- Delete records using the DELETE keyword. Display all the records.
- Add 4 new records.
- Truncate the table. Display all the records.
- Add 2 new records.
- Delete the table. Display all the tables in the database.

LAB ASSIGNMENT

1. Create two tables with at least 6 columns having 4 different data types.
2. Show the structure of tables created in (1).
3. Insert at least 4 records into the first table using column names.
4. Insert at least 4 records into the second table without using the column's name.
5. With a separate query for each table, show the table data.
6. Add two columns into any one table.
7. Alter the data type of any two columns from any table.
8. Truncate the first table data.
9. Delete the third record of the second table.
10. Update the first record of the second table.

SUBMISSION GUIDELINES

- Take a screenshot of each task.
- Place all the screenshots in a single word file labeled with Roll No and Lab No. e.g. **'cs181xxx_Lab01'**.
- Convert the file into PDF.
- Submit the PDF file at LMS
- **-100%** policies for plagiarism.