COMPUTER ORGANIZATION AND ARCHITECTURE

Lecture 8

Introduction to Data Definition Language

Dr Victoria Mukami

INTRODUCTION

During the last lecture, we did an introduction to SQL and a brief history as well. In this lecture, we now start writing SQL statements and we begin with the data definition statements. We will mainly focus on the creation of database tables and how to manipulate and modify existing tables and their structures.

Learning objectives

By the end of this topic, you should be able to:

1. Understand the role of ERD diagrams when creating a database
2. Create a database and its various tables

OVERVIEW

If I can take you back through what we have done, we have created an ERD for Bustani College in lecture 5. The ERD diagram could then be used to create a database. While previously we have dealt with design, we are now on the development of the database. We will create some not all tables from the ERD diagram. In lecture 7 we installed MySQL Workbench. Part of this lecture will include screenshots of where to type code and a view of the results. The next lecture will focus on altering and deleting tables and fields created during this lecture.

INTRODUCTION TO DDL

As seen in the last lecture, DDL stands for Data Definition Language. These are statements that are used to create database objects such as tables and views. This first lecture will focus on the creation of tables and views. The statements used to create tables include CREATE and DROP. Before we start working with the creation of tables let us first review the Bustani ERD we created from Lecture 5. The ERD diagram as shown in Figure 1 has 6 tables. For the lecture, we cannot work with all the tables. We will therefore create the instructor and employee tables. We will add new fields to the tables as shown in Tables 1 and 2.

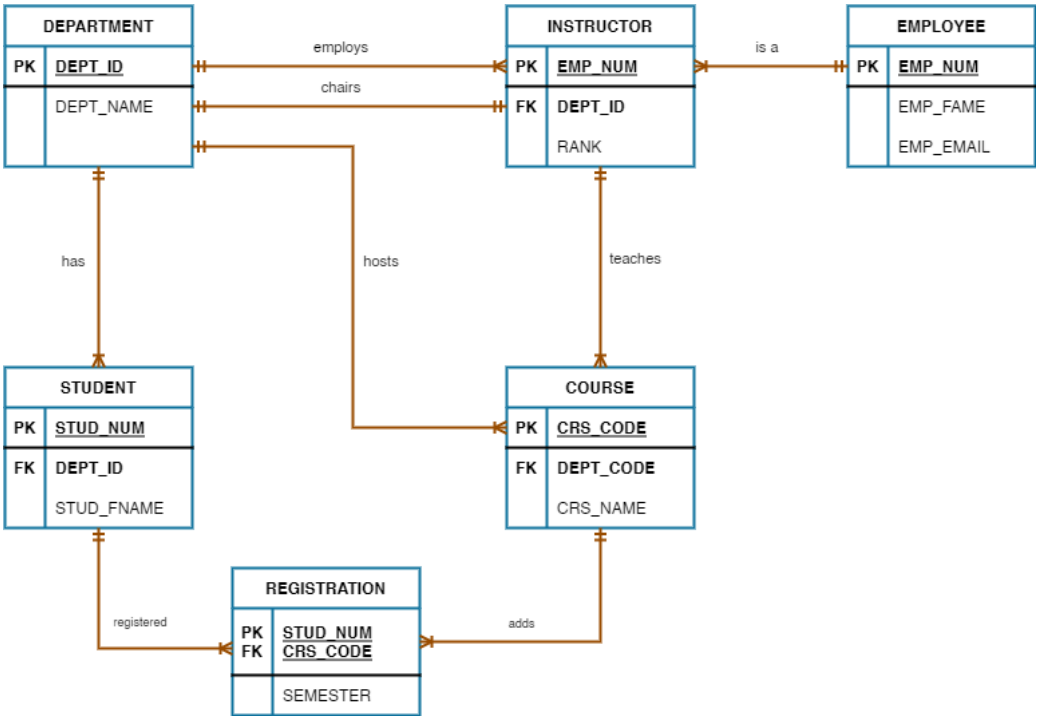


Figure 1: Bustani College ERD Diagram

Employee Table

The employee table as shown in table 1 will contain 6 fields namely: Employee Number, First Name, Last Name, Gender, Date of Birth and Marital Status. Note shorter one-word names have been used within the table.

Table 1: Employee Data Table

Emp_Num	FName	LName	Gender	DOB	MStatus
EMP001	Agnes	Njue	F	12/09/1997	Single
EMP002	Will	Omondi	M	24/08/1989	Married
EMP003	Carol	Njagi	F	09/01/1993	Single
EMP004	Mary	Munene	F	06/08/1998	Married
EMP005	Cathy	Resley	M	16/04/1989	Married
EMP006	Abdul	Muita	M	30/07/1988	Single
EMP007	Steve	Kimuri	M	02/03/1999	Married
EMP008	Ruth	Kimuli	F	07/04/1996	Single

Instructor Table

The instructor table as shown in table 2 will contain 4 fields namely: Instructor ID, Employee Number, Department ID, and Post.

Table 2: Instructor Data Table

ID	Emp_Num	Dep_ID	Post
1	EMP001	CIT	Lecturer
2	EMP002	MTH	Chairman
3	EMP003	CIT	Dean
4	EMP004	CIT	Lecturer
5	EMP005	CIT	Lecturer
6	EMP006	CIT	Senior Lecturer
7	EMP007	MTH	Lecturer
8	EMP008	MTH	Senior Lecturer

The simplified ERD diagram is shown in figure 2.

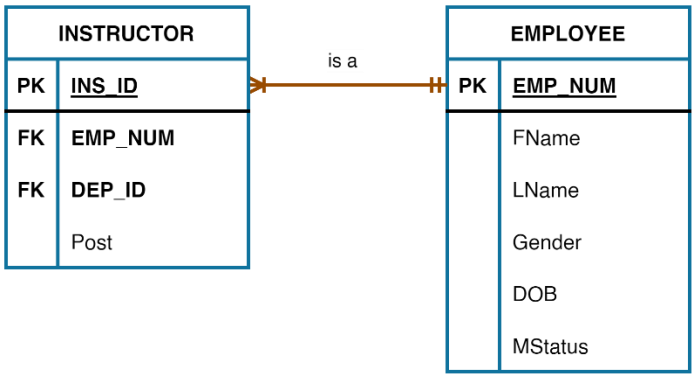
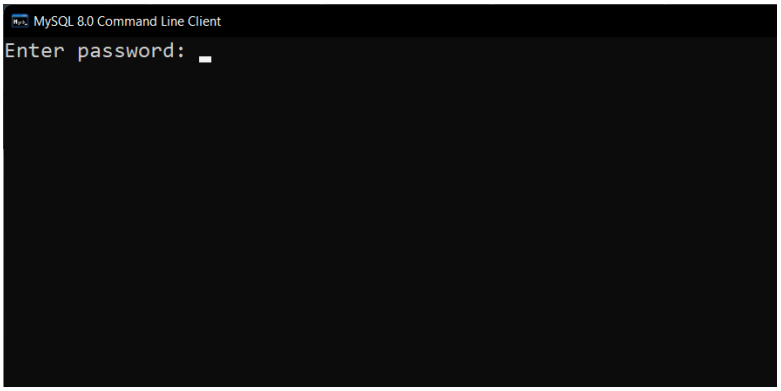


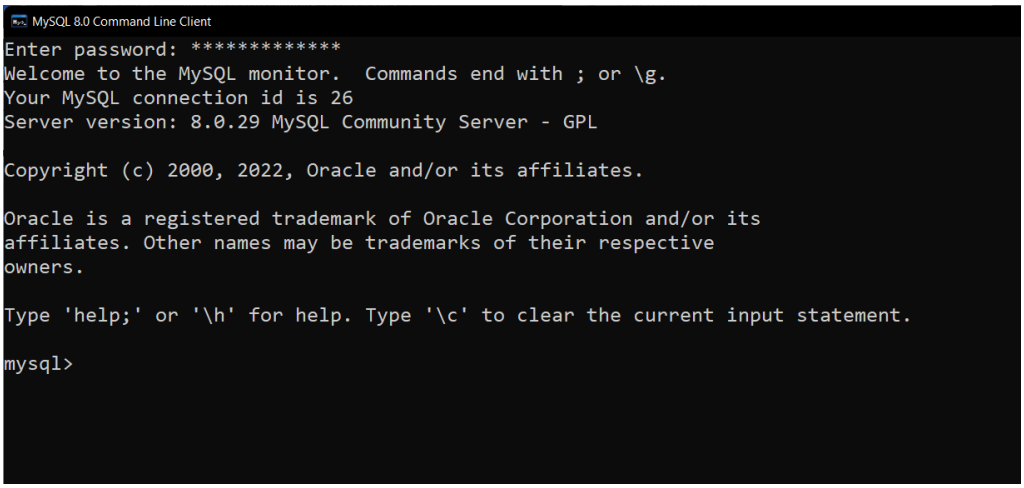
Figure 2: Simplified ERD

CREATE DATABASE

Before we can create the database, we will need to log in to MySQL Command Line Client. Remember to access you will click on the MySQL folder from the Start menu. When it opens you will be asked to insert the password.



Once you have inserted your password, you will be presented with the following screen where we will type all our SQL commands.



We start by first creating a database. To create a database, we use

```
CREATE DATABASE databasename;
```

The database name in our case would be called Bustani. The code to create the database would be:

```
CREATE DATABASE Bustani;
```

The output

```
mysql> CREATE DATABASE Bustani;
Query OK, 1 row affected (0.01 sec)
```

To be able to see the databases created you use the statement below.

```
SHOW DATABASES;
```

The output is shown below where we can see there are several default databases: information_schema, mysql, performance_schema and sys.

```
mysql> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| bustani            |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.01 sec)
```

Once we create the database, then we will need to select the database so that we can create the tables within the database. To be able to use the Bustani database we use:

```
USE Bustani;
```

```
mysql> USE Bustani;
Database changed
mysql>
```

CREATE TABLE

Now that we have created a database, we will create the two tables: Employee and Instructor. Before we create the table we need to discuss the syntax. The syntax for creating a table is shown [1].

```
CREATE TABLE tablename (
        fieldname1      datatype    constraint(s),
        fieldname1      datatype    constraint,
        PRIMARYKEY      (fieldname));
```

From the syntax above additional lines of code could include the foreign key and additional constraints. The various names above are explained below:

tablename: these are user-defined table names like employee or instructor

fieldname1: these are our fields or column headers. For instance, the employee table contains 6 fields e.g., Emp_Num

datatype: this is a valid data type definition. The data types used are listed in table 3.

Constraint: this are valid constraints. The constraints are listed in table 4.

Table 3: SQL Data Types

DATA TYPE	DESCRIPTION
CHAR( )	A string that can contain letters, numbers, and special characters. Length is 0 to 255 and is placed inside the brackets
VARCHAR( )	Like a CHAR but has a longer size. The length is 0 to 65535 and is placed inside the brackets
INT( )	This is a medium-sized integer. Allows for a max of 255 digits and this is placed inside the brackets
FLOAT( )	This is a floating-point number. This is similar to a fraction in mathematics. The length is placed inside the brackets
DECIMAL( )	This is a fixed-point number. Within the brackets, the size of the digits are placed and the number of decimal points i.e., (25, 2) where a max of 25 digits with 2 decimal points
DATE	This is a date that has the format starting with year then month then day (YYYY-MM-DD)
YEAR	This is a 4-digit year e.g., 2022

Table 4: SQL Constraints

CONSTRAINT	DESCRIPTION
NOT NULL	A column cannot be empty. Some data has to be input
UNIQUE	Makes sure that all the values are not repeated
PRIMARY KEY	A primary key from previous lectures is not empty nor repeated
FOREIGN KEY	This shows the column as being a primary key from a different table

Now that we have defined the various sections of the create table statement, let us create the employee table. The Employee table had 6 fields: Emp_Num, FName, LName, Gender, DOB, MStatus.

```
CREATE TABLE Employee (
    Emp_Num     CHAR (6)          NOT NULL   UNIQUE,
    FName       VARCHAR (25)      NOT NULL,
    LName       VARCHAR (25)      NOT NULL,
    Gender      CHAR (1)          NOT NULL,
```

```
    DOB         DATE                NOT NULL,
    MStatus     VARCHAR(15)         NOT NULL,
    PRIMARY KEY     (Emp_Num)
);
```

The output will look as follows:

```
mysql> CREATE TABLE Employee (
    -> Emp_Num CHAR(6) NOT NULL UNIQUE,
    -> FName VARCHAR(25) NOT NULL,
    -> LName VARCHAR(25) NOT NULL,
    -> Gender CHAR(1) NOT NULL,
    -> DOB DATE NOT NULL,
    -> MStatus VARCHAR(15) NOT NULL,
    -> PRIMARY KEY (Emp_Num)
    -> );
Query OK, 0 rows affected (0.03 sec)
```

Note that we place each code on its line for readability and to identify any errors that may come up. If you write the entire code on one line, it will be hard to identify and correct any errors. To view the table that has been created you use the SHOW or DESCRIBE statement:

```
SHOW COLUMNS FROM tablename;
```

We replace tablename with our table.

```
SHOW COLUMNS FROM Employee;
```

Describe works in the same manner and the output will be the same.

```
DESCRIBE Employee;
```

The output:

```
mysql> SHOW COLUMNS FROM Employee;
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| Emp_Num | char(6)     | NO   | PRI | NULL    |       |
| FName   | varchar(25) | NO   |     | NULL    |       |
| LName   | varchar(25) | NO   |     | NULL    |       |
| Gender  | char(1)     | NO   |     | NULL    |       |
| DOB     | date        | NO   |     | NULL    |       |
| MStatus | varchar(15) | NO   |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
6 rows in set (0.01 sec)
```

The code for the instructor table is provided below. Please note the instructor table has a foreign key Emp_Num and Dep_ID from the Employee table and the Department table as per the ERD diagram from Figure 1. The fields are four and include ID,

Emp_Num, Dep_ID and Post. The Dep_ID Foreign Key link cannot be added yet as it is a foreign key for a table that does not exist.

```
CREATE TABLE Instructor (
        Ins_ID          INTEGER             NOT NULL    UNIQUE,
        Emp_Num         CHAR (6),
        Dep_ID          CHAR (3),
        Post            VARCHAR (15)        NOT NULL,
        PRIMARY KEY         (INS_ID),
        FOREIGN KEY         (Emp_Num)   REFERENCES      Employee(Emp_Num),
);
```

The output:

```
mysql> CREATE TABLE Instructor (
    -> Ins_ID INTEGER NOT NULL UNIQUE,
    -> Emp_Num CHAR(6),
    -> Dep_ID CHAR(3),
    -> Post VARCHAR(15) NOT NULL,
    -> PRIMARY KEY (INS_ID),
    -> FOREIGN KEY (Emp_Num) REFERENCES Employee(Emp_Num)
    -> );
Query OK, 0 rows affected (0.05 sec)
```

The tables are shown by typing

```
SHOW TABLES;
```

```
mysql> SHOW TABLES;
+-------------------+
| Tables_in_bustani |
+-------------------+
| employee          |
| instructor        |
+-------------------+
2 rows in set (0.00 sec)
```

SUMMARY

We have started coding in this section. We started with instructions on how to log in to MySQL Command-Line. We also learnt how to create a database and show the databases that exist within MySQL. Next, we created various tables and updated their relevant relationships. For the next topic, we will learn how to alter and delete the tables we just created.

DISCUSSION TOPIC

During Lecture 5, we came up with an ERD for the Lands Office. Using the Entity Relationship Diagram, create a database called Shamba that will hold all the tables that you identified in your ERD diagram. Ensure that you create all the relationships. Take into consideration the data types to be used and the primary keys. Ensure to engage your instructor if you get stuck.

REFERENCES

[1] Database systems: design, implementation, and management, Coronel, C., & Morris, S, Cengage Learning, 2019.

[2] Database Systems: A Practical Approach to Design, Implementation, and Management, Connolly, T., & Begg, C., Pearson, 2015.

[3] Fundamentals of database systems, Elmasri, R., & Navathe, S. B., Pearson Education Limited, 2016.