```
CDIP-UIU: Python & Data Analysis
Batch - 09

Welcome to the 1st lecture on MySQL!


Text Books (optional)
---------------------
1. Database System Concepts by Abraham Silberschatz, Henry F. Korth and S.Sudarshan
2. Database Systems: The Complete Book, by Garcia-Molina, Ullman and Widom


References
----------
1. https://dev.mysql.com/doc/refman/8.0/en/
2. http://www.mysqltutorial.org/
3. https://www.w3resource.com/sql-exercises/
4. https://www.w3schools.com/mysql/default.asp
5. https://www.coffeendcode.com/MYSQL/introduction.htm


======================================================


Database - Collection of correlated data
DBMS     - Organization/Management/Retrieval of data from database

Database system = Database + DBMS (Database Management System)

Website ---> MySQL(high level) ---> DBMS ---> database(low level)
Web server                                    Database server

PC <==> Internet <==> Web server <==> Database server

SQL (standard)    SQL --> Structured Query Language
- MySQL
- Oracle
- MS SQL
- PostgreSQL

Web server - Apache
DB server - MySQL

XAMPP (Bundle) - has apache and MySQL built-in

1. Open XAMPP and turn on Apache and MySQL server
2. Go to browser and type any of the following
    - http://localhost/phpmyadmin/
    - http://127.0.0.1/phpmyadmin/
    - http://127.0.0.1:[port_number]/phpmyadmin/
      (Example: http://127.0.0.1:80/phpmyadmin/)



MySQL - case-insensitive, keyword Uppercase

MySQL - DDL -- Data definition language
        - create/drop       database
        - create/drop/alter table

      - DML -- Data manipulation language
        - Insert data
        - Delete data
        - Search data / SELECT
        - Update data

File extension for MySQL files: .sql

======================================================

----------------------------
| Data Definition Language |
----------------------------

Database:
---------

How to create a database:
- CREATE DATABASE [database_name]
- CREATE DATABASE IF NOT EXISTS [database_name]

How to delete/drop a database:
- DROP DATABASE [database_name]
- DROP DATABASE IF EXISTS [database_name]


Table:
------

Databases are a set of tables. May contain one or more tables.

Example student table
---------------------

-------------------------------------------
| name | dob      | admission_date | cgpa |
-------------------------------------------
| A    | 01.01.95 | 01.01.15       | 3.50 |
-------------------------------------------
| B    | 04.02.95 | 02.01.15       | 3.33 |
-------------------------------------------


Python class example
--------------------

class Student:
    def __init__(self, name, dob, admission_date, cgpa):
        self.name = name
        self.dob = dob
        self.admission_date = admission_date
        self.cgpa = cgpa

student_1 = new Student('A', '01.01.95', '01.01.15', 3.50)
student_2 = new Student('B', '04.02.95', '02.01.15', 3.33)


How to create a table in a database:
-   CREATE TABLE table_name (
        column1 datatype,
        column2 datatype,
        column3 datatype,
        ....
   );

Python datatypes:
- int       [numeric]
- float     [numeric]
- string    [text]
- boolean   [boolean - true/false]
- complex   [numeric] (a + ib)

name = "Shabab" [compiler would automatically understand that 'name' is a variable and its datatype is string, looking at the datatype]

MySQL datatypes:
- INT (size) -- by default 11
- FLOAT (m, d) --> Example: 3.1416; m = 5, d = 4
  FLOAT (10, 5) --> xx, xxx . xxxxx
- DOUBLE (m, d)

- VARCHAR (size) --> Variable characters
  VARCHAR(30) --> "This is a boy"

- DATE -- format: 'YYYY-MM-DD' = '1995-01-01'
- DATETIME -- format: 'YYYY-MM-DD hh:mm:ss' = '1995-01-01 23:59:59"
- TIME -- format: 'hhh:mm:ss' = '15:07:23' >> Range: '-838:59:59' to '838:59:59'


Table creation example:
-----------------------

CREATE TABLE students (
    name VARCHAR(50),
    dob DATE,
    admission_date DATE,
    cgpa FLOAT(3, 2),
    courses INT
);

(Without using underscore. Using ``)

CREATE TABLE teachers (
    id int(10),
    name VARCHAR(50),
    `date of birth` DATE,
    `joining date` DATE
);



How to delete/drop a table:
- DROP TABLE [table_name]
- DROP TABLE IF EXISTS [table_name]


Alter table:
------------

```

```
How to alter a table:
    Alter --> change/edit

- add column
- delete column
- datatype changing

- column rename
- table rename


Alter table - add column:
-------------------------

ALTER TABLE table_name
ADD column_name datatype;

ALTER TABLE students
ADD department VARCHAR(50);

ALTER TABLE students
ADD id INT(10);


Alter table - drop column:
--------------------------

ALTER TABLE table_name
DROP COLUMN column_name;

ALTER TABLE students
DROP COLUMN admission_date;


Alter table - change column datatype:
-------------------------------------

ALTER TABLE table_name
MODIFY COLUMN column_name datatype;

ALTER TABLE students
MODIFY COLUMN department INT(15);


MySQL Constraints:
------------------
- NOT NULL - Ensures that a column cannot have a NULL value
- UNIQUE - Ensures that all values in a column are different
- DEFAULT - Sets a default value for a column if no value is specified

CREATE TABLE students (
    id INT,
    name VARCHAR(30) UNIQUE,
    dob DATE,
    admission_date DATE,
    cgpa FLOAT(3,2) DEFAULT 0.00,
    dept VARCHAR(5) NOT NULL
);

(Example of NOT NULL, Unique and DEFAULT)
```

ID	name	dob	admission_date	cgpa	dept	
1	A	01.01.95	01.01.15	3.50	CSE	
2	B	04.02.95	02.01.15	3.33	NULL	(Error --> NULL in dept)
3	C	18.02.95	01.01.15	3.63	BBA	
4	A	01.04.95	01.01.15	3.46	BBA	(Error --> unique name)
5	E	08.05.96	01.01.16	0.00	Civil	(Default --> cgpa = 0.00)

- PRIMARY KEY - A combination of a NOT NULL and UNIQUE. Uniquely identifies each row in a table
- FOREIGN KEY - Prevents actions that would destroy links between tables

What is a primary key?
- A unique identifier for a table's rows.

Students table:
===============

ID (pk)	name	dob	cgpa	dept (fk)	
1	A	01.01.95	3.50	1	--> a student's info
2	B	04.02.95	3.33	2	
3	C	18.02.95	3.63	3	
4	A	01.04.95	3.46	3	

What is a foreign key?
- A unique identifier that uniquely identifies an object/row/instance of another table

Departments table:
==================

ID (pk)	name	head	no_of_faculties
1	CSE	Dr. A	20
2	EEE	Dr. B	18
3	BBA	Dr. C	22
4	Civil	Dr. D	23

How to add primary key:

```
-----------------------
- Using 'PRIMARY KEY' keyword   (Also example of default & unique)
CREATE TABLE students (
    id int PRIMARY KEY,
    name VARCHAR(50) UNIQUE,
    dob DATE,
    admission_date DATE,
    cgpa FLOAT(3,2) DEFAULT 0.00
);

- Using constraints
CREATE TABLE students (
    id int,
    name VARCHAR(50) UNIQUE,
    dob DATE,
    admission_date DATE,
    cgpa FLOAT(3,2) DEFAULT 0.00,

    CONSTRAINT pk_students PRIMARY KEY (id)
);


CANDIDATE KEY
-------------
- Unique identifier
- Primary key is selected from candidate keys


Example of NOT NULL:
--------------------
CREATE TABLE departments (
    id int PRIMARY KEY,
    name VARCHAR(10),
    head VARCHAR(50) NOT NULL,
    no_of_faculties int
);


How to add foreign key:
-----------------------

Note:
When we want to refer Table B in Table A,
Foreign key datatype of Table A = Primary key datatype of Table B --> (MUST BE SAME)

- While altering table
ALTER TABLE students
ADD dept int,
ADD CONSTRAINT fk_students_department FOREIGN KEY (dept) REFERENCES departments(id)


- While creating table
CREATE TABLE students (
    id int AUTO_INCREMENT,
    name VARCHAR(50) UNIQUE,
    dob DATE,
    admission_date DATE,
    cgpa FLOAT(3,2) DEFAULT 0.00,
    dept int,

    CONSTRAINT pk_students PRIMARY KEY (id),
    CONSTRAINT fk_students_department FOREIGN KEY (dept) REFERENCES departments(id)
);


======================================================


------------------------------
| Data Manipulation Language |
------------------------------


How to insert data in table:

- INSERT INTO [table_name] (col1, col2, col3., ...)
  VALUES (val1, val2, val3, ...)

INSERT INTO `students`(`id`, `name`, `dob`, `admission_date`, `cgpa`, `dept`)
VALUES (1, 'A', '1995-01-01', '2015-01-01', '3.50', 1)


How to select data from table: (Show data)

SELECT col1, col2, ...,
FROM table_name,
WHERE condition

Example:
- SELECT name
  FROM students

- SELECT name, cgpa
  FROM students

How to select all columns: Using (*)
------------------------------------
- SELECT *
  FROM students


Example of WHERE:
- SELECT name, cgpa
  FROM students
  WHERE id = 1

- SELECT name, dob
  FROM students
  WHERE cgpa = 0.00


Multiple conditions:
--------------------

When you have 2 or more conditions in WHERE clause.

- AND = Must satisfy both/all conditions
- OR = Must satisfy at least one condition
```

```
Example students table:
-----------------------
----------------------------------------------------------
| ID | name | dob        | admission_date | cgpa | dept  |
----------------------------------------------------------
| 1  | A    | 1995-01-01 | 2015-01-01     | 3.50 | 1     |
----------------------------------------------------------
| 2  | B    | 1995-01-01 | 2015-01-01     | 0.00 | 1     |
----------------------------------------------------------
| 3  | C    | 1995-01-01 | 2015-01-01     | 0.00 | 1     |
----------------------------------------------------------

conditions:                            Output:
-----------                            ---------------------
- SELECT name, dob                     | name | dob        |
  FROM students                        ---------------------
  WHERE cgpa = 0.00 AND id = 2         | B    | 1995-01-01 |
                                       ---------------------


- SELECT name, dob                     | name | dob        |
  FROM students                        ---------------------
  WHERE cgpa = 0.00 OR id = 2          | B    | 1995-01-01 |
                                       ---------------------
                                       | C    | 1995-01-01 |

```