Python-mysql

Besant Technologies

Database:

- A structured set of data stored in computer.
- Supports data manipulation.
- Efficient retrieval, insertion and deletion of data from database.
- Ex: Telephone directory, Employee details, Electricity bill details.

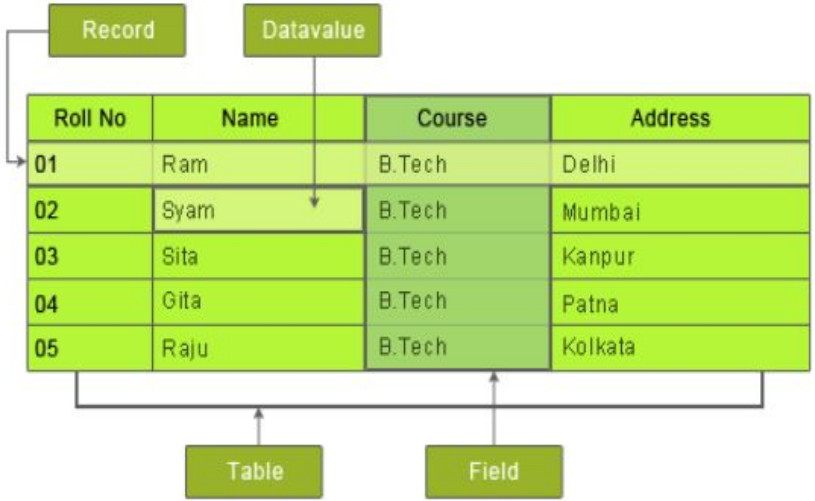
Database Management System (DBMS):

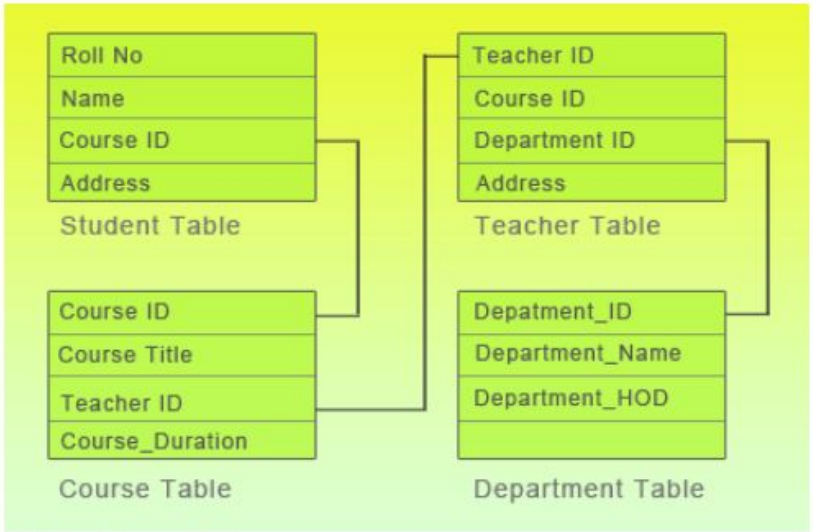
- Software used to manage data.
- A collection of programs.
- For accessing, manipulating and reporting of data.
- Tasks:
  - Data definition
  - Data updation
  - Data Retrieval
  - User Administration

Relational Database:

- In relational database, everything is represented as a table.
- record/entity - each row of table
- Each row has same type of data.
- Attributes - columns of the table.
- Each column is unique and has a predefined data type.
- Primary key - a field in the table which uniquely identifies each record. Primary key must contain unique values. It cannot have NULL values and each table will have only one primary key.

Ex:





Ralation Among Table

SQL:

- Structured Query Language
- A standard language for dealing with relational databases.
- Used to add, update, delete database elements.
-

mysql:

- Open source software for relational databases.
- Install: sudo apt-get install mysql-server (ubuntu)
- Download: https://www.mysql.com/downloads/
- mysql-connector - sudo pip install mysql-connector-python (ubuntu)
- Download: https://dev.mysql.com/downloads/connector/python/
- Mysql-connector is used for programs only.

Commands:

- **All mysql commands are uppercase and end with ‘;’**
- Accessing mysql shell: mysql -u root -p
- Checking available databases: SHOW DATABASES;
- Creating a database: CREATE DATABASE database_name;
- Deleting a database: DROP DATABASE database_name;
- Opening an existing database: USE database_name;
- Checking available tables: SHOW tables;
- Creating a table: CREATE TABLE tablename (<options>);

- Ex: CREATE TABLE potluck (id INT NOT NULL PRIMARY KEY AUTO_INCREMENT,

  name VARCHAR(20),

  food VARCHAR(30),

  confirmed CHAR(1),

  signup_date DATE);

- Checking the table organization: DESCRIBE potluck;
- Insert information in rows: INSERT INTO `potluck` (`id`,`name`,`food`,`confirmed`,`signup_date`) VALUES (NULL, "John", "Casserole","Y", '2019-08-31');

- Looking at the entries: SELECT * FROM potluck;
- SELECT * from potluck WHERE `potluck`.`id`="1";
- Updating the information: UPDATE `potluck`
- -> SET
- -> `confirmed`='N'
- -> WHERE `potluck`.`id`='1'
- Adding an extra field: ALTER TABLE potluck ADD email VARCHAR(40);
- ALTER TABLE potluck ADD email VARCHAR(40) AFTER name;
- Deleting a field: ALTER TABLE potluck DROP email;
- Deleting an entry: DELETE FROM [table name] WHERE [column name]=[field text];

Creating users and giving permissions:

- CREATE USER 'newuser'@'localhost' IDENTIFIED BY 'password';
- GRANT ALL PRIVILEGES ON * . * TO 'newuser'@'localhost';
- * indicates grant permissions on all databases and all tables.
- FLUSH PRIVILEGES; - to make the changes into effect.
- Giving specific permission: GRANT type_of_permission ON database_name.table_name TO ‘username’@'localhost’;
- Revoke existing permission: REVOKE type_of_permission ON database_name.table_name FROM ‘username’@‘localhost’;
- Checking permissions: SHOW GRANTS username;
- Deleting the user: DROP USER ‘username’@‘localhost’;

- Types of permissions:
  - ALL PRIVILEGES
  - CREATE
  - DROP
  - DELETE
  - INSERT
  - SELECT
  - UPDATE
  - GRANT OPTION

* Coming out of mysql shell: quit

Procedure To Follow In Python To Work With MySQL:

1. Connect to the database.
2. Create an object for your database.
3. Execute the MYSQL query.
4. Fetch records from the result.
5. Informing the Database if you make any changes in the table.

- import mysql.connector

Connecting to database:

- db = mysql.connect(host = "localhost", user = "root", passwd = "dbms")
- db - mysql connection object.
- cursor = db.cursor() - cursor is a class in mysql-connector which is used to execute the 'SQL' statements in 'Python'.
- 'execute()' method is used to compile a 'SQL' statement.
- 'fetchall()' method fetches all the rows from the last executed statement.
- Ex: Creating a database.
  - cursor.execute("CREATE DATABASE datacamp")