Chapter 8: Connecting MySQL database with Python

Introduction to Database

Data is raw collection of any information that consists of numbers, figures, facts, measurements, quantities etc. that are stored, extracted, manipulated and processed for further analysis purpose.

Database is nothing but a container where we store meaningful data/information. It is the collection of data from where we can access, modify, and do any kind of manipulation to it.

How database works

Let's take an example of Bank ATM. When we are operating an ATM what we do?
First, we enter our card then give the valid details as per the screen direct us. Then we give our PIN and our cash comes out.
So, here how a machine knows all your information and gives you out the cash. The data we entered is already stored in the database of the respective bank we have our account. Once it will get checked after entering our PIN that has previously stored in the backend it gives out the cash.

Here, on the daily basis we are going to come across 2 types of databases.

1. SQL Database
2. NOSQL Database

Introduction to SQL

SQL stands for Structured Query Language. It is the preferred language which is used to stores the data in the database in an organized way. It is a communicating medium through which we can communicate to the database.

SQL was developed by Donald D. Chamberlin and Raymond F.Boyce during 1970's. It was developed at IBM Research Centre. It is an open - source relational data management system.

Installation Of MYSQL and Connector to Python

1. Go to the official website of MYSQL and click on the MYSQL Community Server Edition.
2. Next, there are two options available to download the setup. Choose the version number for the MySQL community server, which you want. If you have good internet connectivity, then choose the mysql-installer-web-community. Otherwise, choose the other one.

Installing MYSQL On Windows

Step 1- After downloading the setup, unzip it anywhere and double click the MSI installer .exe file.
Step 2- In the next screen, choose the Setup Type. There are 5 types of setup type, choose to FULL option to install appropriate MYSQL product and features. Then click on NEXT button.

Step 3- Once we click on the Next button, it may give information about some features. Click for our confirmation of a few products not getting installed. Here, we have to click on the Yes button. Then click on the Execute Button.
Step 4- In the next wizard, we will see a dialog box that asks for our confirmation of a few products not getting installed. Here, we have to click on the Yes button. Then click on the Execute Button.
Step 5- Once we click on Execute button it will start downloading all the installations of the Product. After completion, Click on Next Button.
Step 6- In the next wizard we need to configure MYSQL Server and Router. Click on the Next Button.
Step 7- Now, choose the Standalone MySQL Server/Classic MySQL Replication option and click on Next. Here, you can also choose the InnoDB Cluster based on your needs.
Step 8- Keep the default values for "Type and Networking" section. Config type as 'Development Machine', Connectivity as TCP/IP, Port number is 3306. Then click on NEXT Button.
Step 9- Finish the installation by selecting the First option 'Use Strong Password Encryption For Authentication. Then click on NEXT.
Step 10- The next wizard will come up with MYSQL Root Password. After giving the password details, click on NEXT button.
Step 11- Then click on Further to complete the Installation.

Constraints

Constraints are knowns as the conditions we give in a SQL query.
Constraints in SQL means we are applying certain conditions or restrictions on the database. This further means that before inserting data into the database, we are checking for some conditions. If the condition we have applied to the database holds true for the data which is to be inserted, then only the data will be inserted into the database tables.

Constraints in SQL can be categorized into two types:

1. **Column Level Constraint:**
   Column Level Constraint is used to apply a constraint on a single column.
2. **Table Level Constraint:**
   Table Level Constraint is used to apply a constraint on multiple columns.

Some of the real-life examples of constraints are as follows:

Whenever we set a password for any system, there are certain constraints that are to be followed. These constraints may include the following:

- There must be one uppercase character in the password.
- Password must be of at least eight characters in length.
- Password must contain at least one special symbol.

Constraints available in SQL are:

1. NOT NULL
2. UNIQUE
3. PRIMARY KEY

4. FOREIGN KEY
5. CHECK
6. DEFAULT
7. CREATE INDEX

NOT NULL

- NULL means empty, i.e., the value is not available.
- Whenever a table's column is declared as NOT NULL, then the value for that column cannot be empty for any of the table's records.
- There must exist a value in the column to which the NOT NULL constraint is applied.

UNIQUE

- Duplicate values are not allowed in the columns to which the UNIQUE constraint is applied.
- The column with the unique constraint will always contain a unique value.
- This constraint can be applied to one or more than one column of a table, which means more than one unique constraint can exist on a single table.
- Using the UNIQUE constraint, you can also modify the already created tables.

PRIMARY KEY

- PRIMARY KEY Constraint is a combination of NOT NULL and Unique constraints.
- NOT NULL constraint and a UNIQUE constraint together forms a PRIMARY constraint.
- The column to which we have applied the primary constraint will always contain a unique value and will not allow null values.

FOREIGN KEY

- A foreign key is used for referential integrity.
- When we have two tables, and one table takes reference from another table, i.e., the same column is present in both the tables and that column acts as a primary key in one table. That particular column will act as a foreign key in another table.

CHECK

- Whenever a check constraint is applied to the table's column, and the user wants to insert the value in it, then the value will first be checked for certain conditions before inserting the value into that column.
- **For example:** if we have an age column in a table, then the user will insert any value of his choice. The user will also enter even a negative value or any other invalid value. But if the user has applied check constraint on the age column with the condition age greater than 18. Then in such cases, even if a user tries to insert an invalid value such as zero or any other value less than 18, then the age column will not accept that value and will not allow the user to insert it due to the application of check constraint on the age column.

DEFAULT

Whenever a default constraint is applied to the table's column, and the user has not specified the value to be inserted in it, then the default value which was specified while applying the default constraint will be inserted into that particular column.

CREATE INDEX
CREATE INDEX constraint is used to create an index on the table. Indexes are not visible to the user, but they help the user to speed up the searching speed or retrieval of data from the database.

Difference Between Primary Key and Foreign Key

Primary Key	**Foreign Key**
A specific choice of a minimal set of attributes or columns that uniquely specify a tuple or a row in a table.	A field or collection of fields in one table that uniquely identifies a row of another table or the same table.
Related to a single table	Related to two tables.
Value cannot be null	Value can be null.
Cannot have duplicate values	Can have duplicate values
There is only a single primary key in a table.	There can be multiple foreign keys.
Used to identify the records uniquely	Used to link two tables together.

Commands in SQL

- **DATA DEFINITION LANGUAGE (DDL):** It is basically used to create, modify the database structures but not the data. basically, these commands do the operation in database schema.

CREATE – This command is used to create the databases.
ALTER – This is used to alter the structure of database.
DROP - This command is used to delete table from database.
RENAME – Used to rename an object already exists in database.
TRUNCATE -This command is used to delete all the records from a table.

- **DATA MANIPULATION LANGUAGE (DML):** These are the commands used for doing the manipulation with the data.

UPDATE – This command is used to update existing data within a table.
INSERT- It is used to insert data into a table.
DELETE – It is used to delete records from database table.

- **TRANSACTION CONTROL LANGUAGE (TCL):** This commands are used to control the processing of transaction whether it is a success or failure.

COMMIT – This is used to Commit a transaction.
ROLLBACK – Rollback a transaction if any error occurs.
SAVE POINT – For setting a save point.

- **DATA QUERY LANGUAGE (DQL):** It is the Query language which is used for performing queries on the data with schema objects.

SELECT STATEMENT – It is used to fetch out data from database.

- **DATA CONTROL LANGUAGE (DCL):** It deals with the permission, control, authorization of the database systems.

GRANT- This command give user access controls to database.
REVOKE – This command restricts the user access privileges given by the GRANT command.

SELECT STATEMENT

Select statement in SQL query is used to select/fetch out the data from a database.

Syntax:
For fetching out 1 column from the data → **Select col1, col2, col3... from table name**;
For fetching numerous columns at a time → **Select * from table name.** (The * indicates the total number of columns to be fetch out from a database.)

CLAUSES

Clause in SQL is a built-in function that is used to retrieve the data from the records present in the database. Different clauses in SQL are used to fetch or retrieve the records from the database table.

ORDERBY
The ORDER BY keyword is used to sort the result in ascending or descending order. It arranges the result in ascending order by default.

Syntax - Select col1, col2.. from table_name
ORDER BY col1, col2, ... ASC|DESC;
(ASC for ascending order and DSC for descending order.)

GROUP BY
The GROUP BY clause is used to group the rows that have the same values into the resultant row. It is mostly used with Aggregate functions to group the result set by one or more columns.

Syntax - Select column_name from table_name
WHERE condition
GROUP BY column_name
ORDER BY column_name;

WHERE
The WHERE clause is used to filter records. It is used to fetch out only those records that fulfils the specified condition passed in the WHERE clause.

Syntax - Select col1, col2,... from table_name
WHERE condition;

BETWEEN
The BETWEEN clause is used to selects values within a given particular range. The values can be any data type.

Syntax - Select col_name from table_name
WHERE col_name
BETWEEN value1 AND value2;

JOINS

A join is used to merge or combine rows from two or more tables based on the similar column between them. The Joins are of different types:

INNER JOIN

The inner join is used to combine the two tables that have matching values in both.
Syntax – Select column_name from table1 INNER JOIN table2
ON table1.column_name = table2.column_name;

LEFT JOIN

The left join keyword returns all the records from lefty table and the matching records from right table.
Syntax - SELECT column_name FROM table1 LEFT JOIN table2
ON table1.column_name = table2.column_name;

RIGHT JOIN

The right join keyword will fetch out all the records from right table and the matching reocrds from left table.
Syntax - SELECT column_name FROM table1 RIGHT JOIN table2
ON table1.column_name = table2.column_name;

FULL OUTER JOIN

The full outer join keyword is used return all the records when there is a match in either left table or right table.
Syntax - SELECT column name FROM table1 FULL OUTER JOIN table2
ON table1.column name = table2.column name;

SELF JOIN

Self-join is a regular join. It is used to join the table with itself.
Syntax – SELECT column name
FROM table1 T1, table2 T2
WHERE condition;

OPERATORS

Difference between union and union all

UNION operator selects only distinct values by default.
But UNION ALL operator can allow duplicate values.

FUNCTIONS IN SQL

A function is bunch of codes which is used for any operation when it is called. There are 4 types of functions in SQL

1. **Aggregate Functions**
   - SUM() – It will give sum of the data.
   - AVG() - It will result an average
   - MIN() - It will give the minimum
   - MAX() – Gives the maximum value
   - COUNT() – Returns the number of records returned by select query.

2. **Number Functions**
   - Round() – It will round up the number to specific decimal places.
   - Mod() – Returns the remainder of a number divided by another number

3. **Conversion Functions**
   - Convert() – Converts a value into specific datatype or character set.

4. **Character Functions**
   - Upper()- Convert a string to upper case.
   - Lower() – Convert a string to lower case
   - Length() – Returns the length of the string means number of characters in the string.
   - Concat() – Merging or adding two strings together.
   - Replace()- Replace all concurrencies of a substring within a string , with a new substring.