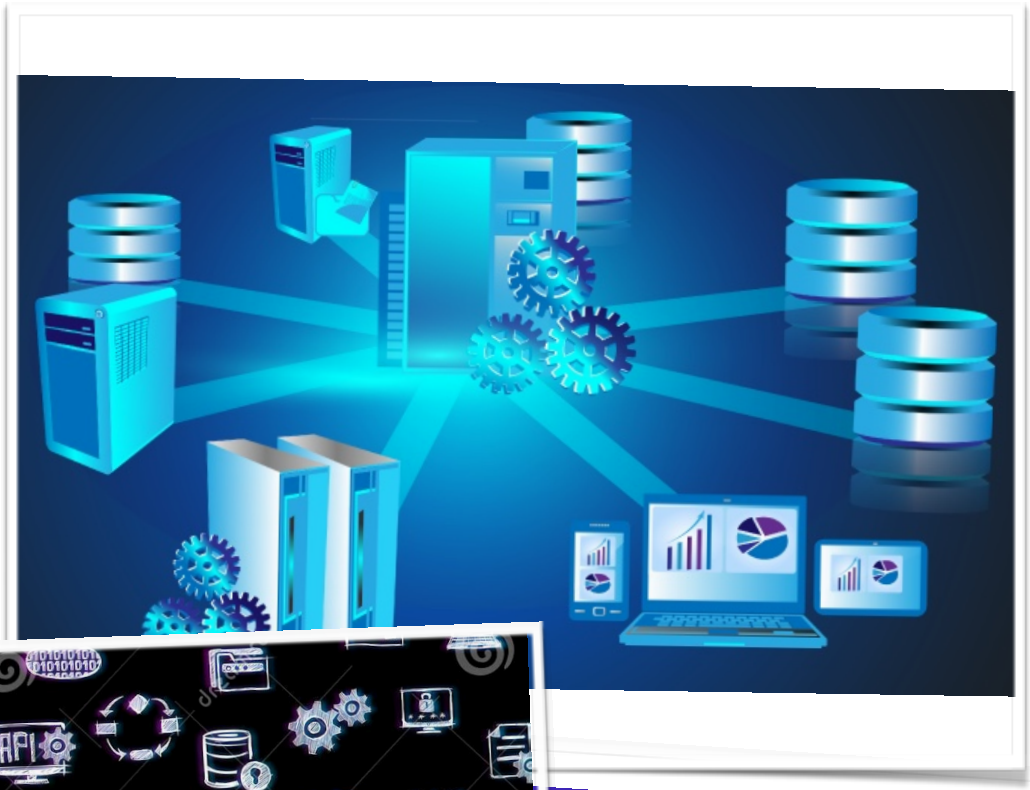
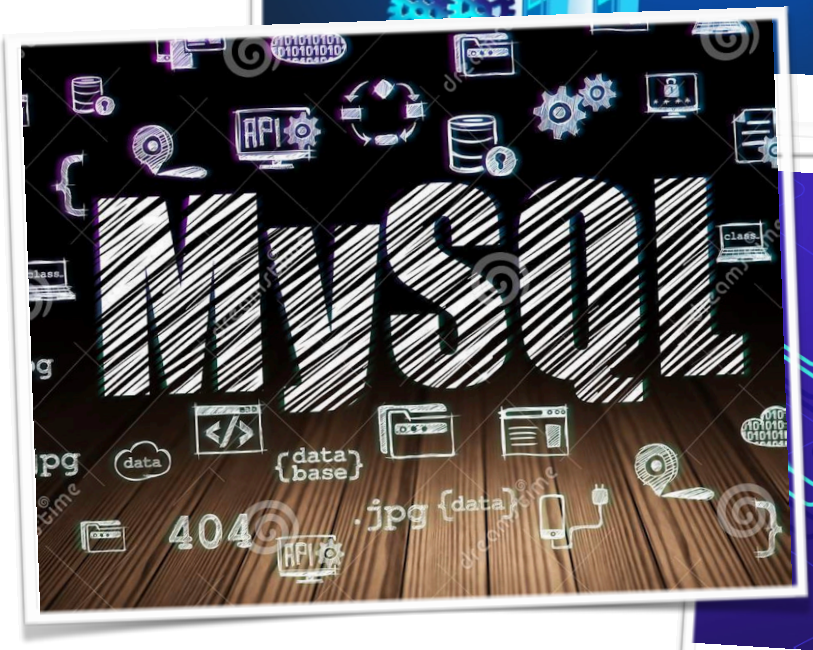
Sandesh Subedi
Roll. Number : 26

INTRODUCTION TO DATABASE





Basics of MySQL Queries

The first step is downloading and to get access to MySQL.
To login to MySQL with user root on Mac, we need to go to Terminal and enter : **mysql -u root -p**
Then, we enter the password and get proper access to the software.

In this task, we are given to create a table course that includes **Course ID, Name, Total Students and Instructor.** Before creating a table we need to complete few more steps. They are:

1. Creating a Database :

To create a table, we need a database in which the table will be stored. So, the syntax for creating a database is,
create database Database_Name;

And, after the creation of a database we can check the databases that we have created. The syntax for this is,

show databases;

```
Sandeshs-MacBook-Air:~ sandeshkey$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 8.0.21 MySQL Community Server - GPL

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| pinkman            |
| sys                |
+--------------------+
5 rows in set (0.00 sec)

mysql> create database SANDESH_NPI000040;
Query OK, 1 row affected (0.00 sec)

mysql> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| pinkman            |
| SANDESH_NPI000040  |
| sys                |
+--------------------+
6 rows in set (0.01 sec)

mysql>
```

Here in my case, I have created a database as SANDESH_NPI000040. Following that, I've displayed the database that we created using **'show databases;'** syntax.

2. Table Creation & its Description :

After the database is created, we can create a table. The table is created using syntax,

create Table [Table_Name] [Parameters];

After the table is created, we can also display the table which we made. It is done using syntax:

show tables;

Moreover, we can also see the description of the table and parameters. This is done using following syntax :

describe [table_name];

```
mysql> SHOW TABLES;
Empty set (0.00 sec)

mysql> CREATE TABLE SANDESH_26(Course_ID VARCHAR(30) not null,
    -> Name VARCHAR(30) not null,
    -> Total_Students INT(10) not null,
    -> Instructor VARCHAR(30) not null
    -> );
Query OK, 0 rows affected, 1 warning (0.01 sec)

mysql> SHOW TABLES;
+----------------------------+
| Tables_in_sandesh_npi000040 |
+----------------------------+
| SANDESH_26                 |
+----------------------------+
1 row in set (0.00 sec)

mysql> DESCRIBE SANDESH_26;
+---------------+-------------+------+-----+---------+-------+
| Field         | Type        | Null | Key | Default | Extra |
+---------------+-------------+------+-----+---------+-------+
| Course_ID     | varchar(30) | NO   |     | NULL    |       |
| Name          | varchar(30) | NO   |     | NULL    |       |
| Total_Students | int        | NO   |     | NULL    |       |
| Instructor    | varchar(30) | NO   |     | NULL    |       |
+---------------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)

mysql>
```

Here, as per the given question, the table name is given as SANDESH_26 using name and roll number. Similarly, the table is shown using 'show table;' syntax and described using '**describe SANDESH_26;** ' syntax.

3. Inserting Values Into The Table :

To insert data or values into the table we created, we can use following syntax :

Insert into [Table_Name] (Parameters…) Values(…);

Furthermore, since a table can consist plenty of rows and columns, we might need to add more than one values. To perform that, we can use same syntax given above but adding values to add more data . i.e,

Insert into [Table_Name] (P1, P2, P3..)
VALUES ('V1','V2','V3'..),
('V4','V5','V6'),
('V7','V8','V9'…);

Where, P = Parameters and, V = Values.

```
mysql> INSERT INTO SANDESH_26 (Course_ID, Name, Total_Students, Instructor)
    -> VALUES ('NP001','Literature','22','Henry');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO SANDESH_26 (Course_ID, Name, Total_Students, Instructor)
    -> VALUES ('NP002','History','12','Gerard'),
    -> ('NP003','Management','33','Romero'),
    -> ('NP004','Arts','7','Picasso'),
    -> ('NP005','News','37','Romano'),
    -> ('NP006','Science','19','Newton'),
    -> ('NP007','Politics','45','Andrea'),
    -> ('NP008','Gardening','54','Pearl'),
    -> ('NP009','Teaching','69','Arnold'),
    -> ('NP010','Parenting','6','John'),
    -> ('NP011','Sports','32','Archer');
Query OK, 10 rows affected (0.00 sec)
Records: 10  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM SANDESH_26;
+-----------+------------+----------------+------------+
| Course_ID | Name       | Total_Students | Instructor |
+-----------+------------+----------------+------------+
| NP001     | Literature |             22 | Henry      |
| NP002     | History    |             12 | Gerard     |
| NP003     | Management |             33 | Romero     |
| NP004     | Arts       |              7 | Picasso    |
| NP005     | News       |             37 | Romano     |
| NP006     | Science    |             19 | Newton     |
| NP007     | Politics   |             45 | Andrea     |
| NP008     | Gardening  |             54 | Pearl      |
| NP009     | Teaching   |             69 | Arnold     |
| NP010     | Parenting  |              6 | John       |
| NP011     | Sports     |             32 | Archer     |
+-----------+------------+----------------+------------+
11 rows in set (0.00 sec)

mysql>
```

Here in given figure above,

Insert into SANDESH_26 (Course_ID, Name, Total_Students, Instructor)
VALUES ('NP001','Literature','22','Henry'),
('NP002','History','12','Gerard'),
('NP003','Management','33','Romero');

is taken as an example to demonstrate how to insert values in multiple rows. In the given table, there are 11 rows with different values inserted.

4. Selecting Department As Per Instructions :

We can select only particular section from the table as per the instructions we are provided. This allows user to avoid large amount of data and search required data easily and quickly. There are multiple syntax used in these context. However, some of them are:

- **Selecting courses with minimum 20 students**
- **Selecting courses whose instructors name starts with a letter “A”.**

```
mysql> SELECT * FROM SANDESH_26 WHERE Total_Students >= '20';
+-----------+------------+----------------+------------+
| Course_ID | Name       | Total_Students | Instructor |
+-----------+------------+----------------+------------+
| NP001     | Literature |             22 | Henry      |
| NP003     | Management |             33 | Romero     |
| NP005     | News       |             37 | Romano     |
| NP007     | Politics   |             45 | Andrea     |
| NP008     | Gardening  |             54 | Pearl      |
| NP009     | Teaching   |             69 | Arnold     |
| NP011     | Sports     |             32 | Archer     |
+-----------+------------+----------------+------------+
7 rows in set (0.00 sec)

mysql> SELECT * FROM SANDESH_26 WHERE Instructor LIKE 'A%';
+-----------+----------+----------------+------------+
| Course_ID | Name     | Total_Students | Instructor |
+-----------+----------+----------------+------------+
| NP007     | Politics |             45 | Andrea     |
| NP009     | Teaching |             69 | Arnold     |
| NP011     | Sports   |             32 | Archer     |
+-----------+----------+----------------+------------+
3 rows in set (0.00 sec)
```

From the table displayed above, to find courses with 20 or more students, the following syntax is used :

Select * from [Table_Name] where [Parameter]>= '20';

i.e,

Select * from [SANDESH_26] where [Total_Students]>= '20';

Likewise, to select courses whose instructors name starts with a letter "A", the following syntax is used:

Select * from [Table_Name] where [Parameter] LIKE 'A%';

i.e,

Select * from [SANDESH_26] where [Instructor] LIKE 'A%';

In these cases, parameters are chosen according to the instruction.

- **Showing courses having 'even' number of students**

To display courses that have total students of even numbers, we can use following syntax:

Select * from [Table_Name] where (Parameter % 2) = 0;

```
mysql> SELECT * FROM SANDESH_26 WHERE (Total_Students % 2)= 0;
+-----------+------------+----------------+------------+
| Course_ID | Name       | Total_Students | Instructor |
+-----------+------------+----------------+------------+
| NP001     | Literature |             22 | Henry      |
| NP002     | History    |             12 | Gerard     |
| NP008     | Gardening  |             54 | Pearl      |
| NP010     | Parenting  |              6 | John       |
| NP011     | Sports     |             32 | Archer     |
+-----------+------------+----------------+------------+
5 rows in set (0.00 sec)

mysql>
```

In the table above, following syntax of

Select * from [SANDESH_26] where (Total_students % 2) = 0;

is used to check the courses that have even number of students.

CONCLUSION:

With these tasks we ended up, creating a database, displaying the created database, creating a table, adding multiple rows of values in the table, choosing particular section using different sections and many more using different syntax.

■ ■

NAME : SANDESH SUBEDI

ROLL NO. 26

STUDENT ID : NPI000040