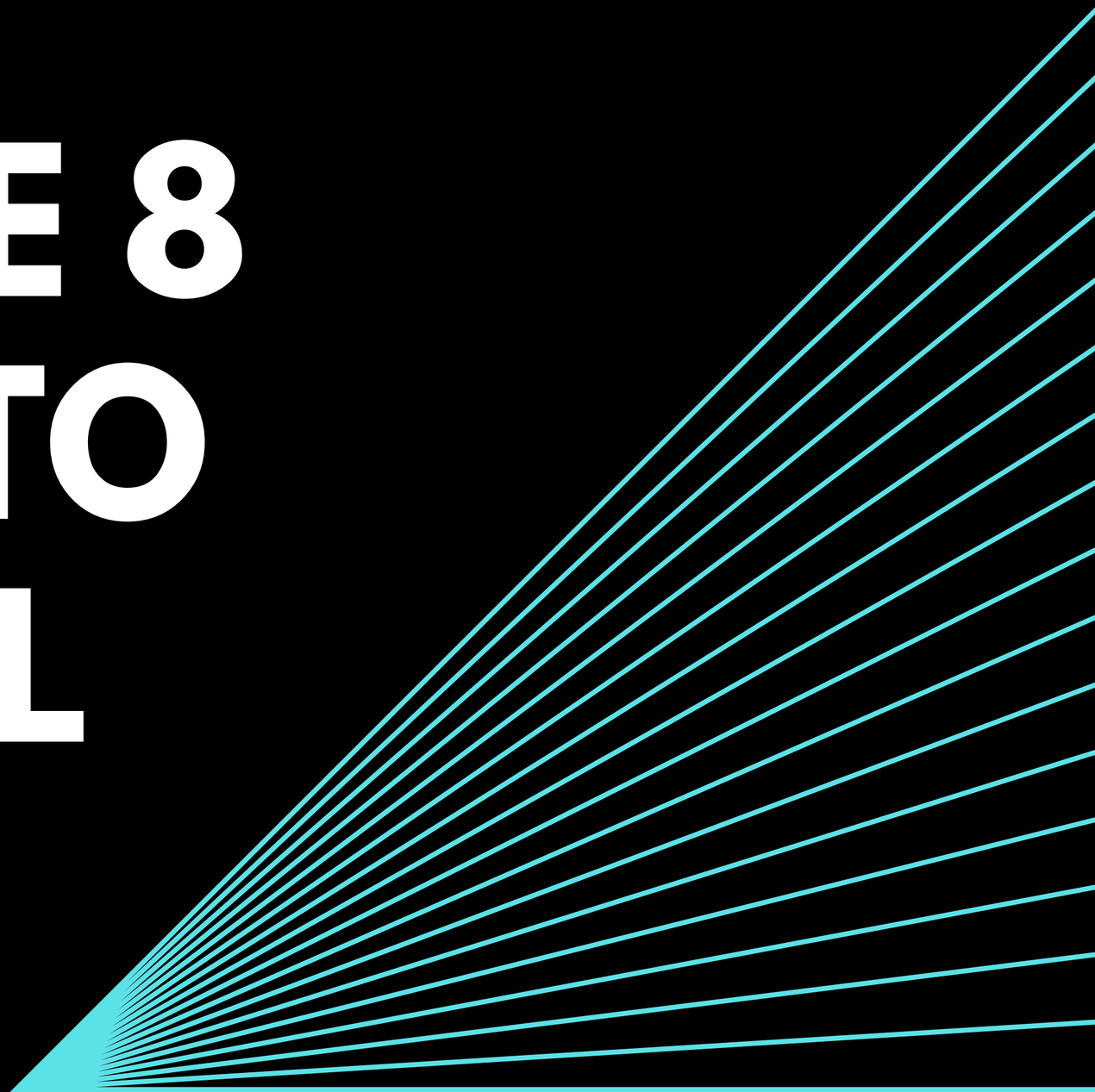
LECTURE 8
INTRO TO MYSQL



SOME POINTERS ON SQL

- Structured Query Language
- Works on FIXED DATA MODEL (Tables)
- Helps a user to interact with the data (direct interaction is not possible and hence MYSQL is required)
- It is a declarative Language (Only matters what needs to be done NOT how to do)

In Procedural Language both matter: What and how.
In Declarative Language only what to do matters.

DDL - DATA DEFINITION LANGUAGE

These commands handle the structure of the table just like deciding the box structure when giving a gift to someone.

The commands under DDL are as follows:

1. CREATE : Creating the Table
2. ALTER : Changing the table structural values such as data type of columns or properties of columns
3. DROP : Dropping tables or certain columns
4. RENAME : Renaming the table (used along with alter)

DML - DATA MANIPULATION LANGUAGE

These commands are responsible for manipulating data after the whole table is created.

The commands under DML are as follows:

1. SELECT
2. INSERT
3. UPDATE
4. DELETE

DCL - DATA CONTROL LANGUAGE

These commands are responsible for setting permission of access to a database for various users.

The commands under DCL are as follows:

1. REVOKE
2. GRANT

TCL - TRANSACTION CONTROL LANGUAGE

These commands are responsible for handling invalid transactions such as ticket not booked but money is taken from account (transaction is rollbacked).

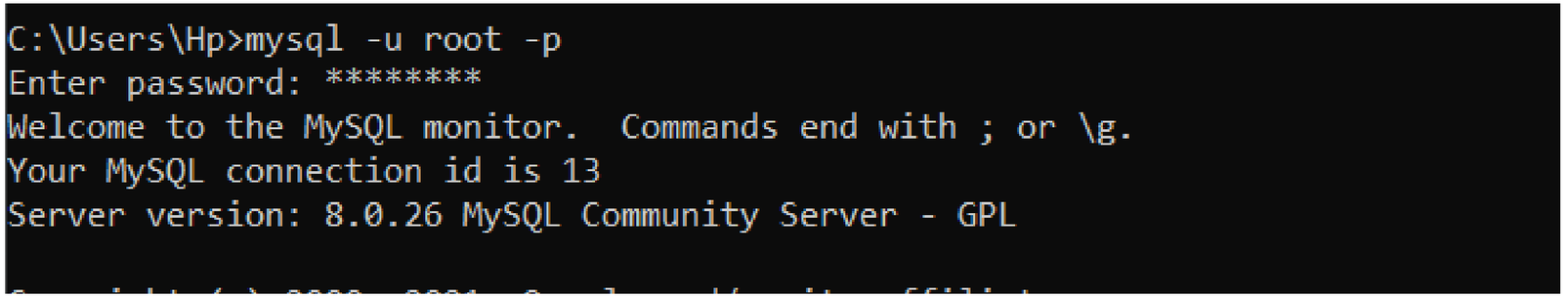
The commands under TCL are as follows:

1. COMMIT
2. SAVEPOINT
3. ROLLBACK

WORKING WITH SQL

HOW TO ACCESS MYSQL?

- Go to Command Prompt/ Terminal
- **mysql -u root - p**
- Enter Password
- You are all set!



CHECK THE ALREADY PRESENT DATABASES

show databases;

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
4 rows in set (0.01 sec)
```

HOW TO GO INSIDE A DATABASE

use < name of database>;

```
mysql> use mysql
Database changed
mysql>
```

LOOK AT THE TABLES INSIDE A DATABASE

show tables;

```
mysql> use mysql
Database changed
mysql> show tables;
+-------------------------------------------
| Tables_in_mysql
+-------------------------------------------
| columns_priv
| component
```

DROPPING A DATABASE

drop <name of database>;

Make sure you don't lose anything important!

CREATING A DATABASE

create database <name of database>;

```
mysql> create database emp_db;
Query OK, 1 row affected (0.01 sec)

mysql> use emp_db;
Database changed
mysql>
```

CREATING A TABLE IN DATABASE

create table <table name> (<column1> <datatype>(<size>) , <column2> <datatype>(<size>)..... and so on)

```
mysql> create table employee(E_ID int,First_Name varchar(200), Last_Name varchar(200),salary int);
Query OK, 0 rows affected (0.04 sec)

mysql> desc employee;
+------------+--------------+------+-----+---------+-------+
| Field      | Type         | Null | Key | Default | Extra |
+------------+--------------+------+-----+---------+-------+
| E_ID       | int          | YES  |     | NULL    |       |
| First_Name | varchar(200) | YES  |     | NULL    |       |
| Last_Name  | varchar(200) | YES  |     | NULL    |       |
| salary     | int          | YES  |     | NULL    |       |
+------------+--------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

INSERT VALUES TO THE TABLE

insert into <table name> values (<column1 value>,<column2 value>,.... and so on)

Make sure the order of values match with the column names!

```
mysql> insert into employee values (001,"Riya", "Sharma", "35000");
Query OK, 1 row affected (0.01 sec)

mysql> insert into employee values (002,"Riyansh", "Shokeen", "55000");
Query OK, 1 row affected (0.00 sec)

mysql> insert into employee values (032,"Arnav", "Kapoor", "45000");
Query OK, 1 row affected (0.00 sec)
```

VIEW TABLE THAT WAS JUST CREATED

select * from <table name>;

```
mysql> select * from employee;
+------+------------+-----------+--------+
| E_ID | First_Name | Last_Name | salary |
+------+------------+-----------+--------+
|    1 | Riya       | Sharma    |  35000 |
|    2 | Riyansh    | Shokeen   |  55000 |
|   32 | Arnav      | Kapoor    |  45000 |
|    8 | Rajesh     | Roy       |  40000 |
|    4 | Kavya      | Sharma    |  55000 |
|   44 | Ritu       | Khan      |  35000 |
|    5 | Karan      | Bansal    |  65000 |
|   15 | Kiran      | Khanna    |  45000 |
|   25 | Zorro      | Jain      |  40000 |
+------+------------+-----------+--------+
9 rows in set (0.00 sec)
```

select <column name> from <table name>;

```
mysql> select First_name from employee;
+------------+
| First_name |
+------------+
| Riya       |
| Riyansh    |
| Arnav      |
| Rajesh     |
| Kavya      |
| Ritu       |
| Karan      |
| Kiran      |
| Zorro      |
+------------+
9 rows in set (0.00 sec)
```

```
mysql> select salary from employee;
+--------+
| salary |
+--------+
|  35000 |
|  55000 |
|  45000 |
|  40000 |
|  55000 |
|  35000 |
|  65000 |
|  45000 |
|  40000 |
+--------+
9 rows in set (0.00 sec)
```

ADDING AN EXTRA COLUMN

alter table <table name> add column <name of new column> <datatype>(<size>);

```
mysql> alter table employee add column Curr_Loc varchar(20);
Query OK, 0 rows affected (0.02 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> select * from employee;
+------+------------+-----------+--------+----------+
| E_ID | First_Name | Last_Name | salary | Curr_Loc |
+------+------------+-----------+--------+----------+
|    1 | Riya       | Sharma    |  35000 | NULL     |
|    2 | Riyansh    | Shokeen   |  55000 | NULL     |
|   32 | Arnav      | Kapoor    |  45000 | NULL     |
|    8 | Rajesh     | Roy       |  40000 | NULL     |
```

DROPPING A COLUMN

alter table <table name> drop <column name>;

```
mysql> alter table employee drop Curr_Loc;
Query OK, 0 rows affected (0.09 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> select * from employee;
+------+------------+-----------+--------+
| E_ID | First_Name | Last_Name | salary |
+------+------------+-----------+--------+
|    1 | Riya       | Sharma    |  35000 |
|    2 | Riyansh    | Shokeen   |  55000 |
|   32 | Arnav      | Kapoor    |  45000 |
|    8 | Rajesh     | Roy       |  40000 |
|    4 | Kavya      | Sharma    |  55000 |
```

ADDING A PRIMARY KEY

alter table <table name> add primary key (<name of the column>);

```
mysql> alter table employee add primary key (E_ID);
Query OK, 0 rows affected (0.04 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

MODIFY DATATYPE OF A COLUMN

alter table <table name> modify <column name> <new data type> (<size>);

```
mysql> alter table employee modify E_ID varchar(200);
Query OK, 9 rows affected (0.04 sec)
Records: 9  Duplicates: 0  Warnings: 0
```

DROP PRIMARY KEY

alter table <table name> drop primary key;

COMBINATION OF PRIMARY KEY

alter table <table name> add primary key (<col1>, <col2>);

END OF LECTURE 8