ASSIGNMENT MYSQL OPERATIONS AND KEYWORDS

Nimra Hshraf

110829

Submitted to:

Mam Sehrish Khan

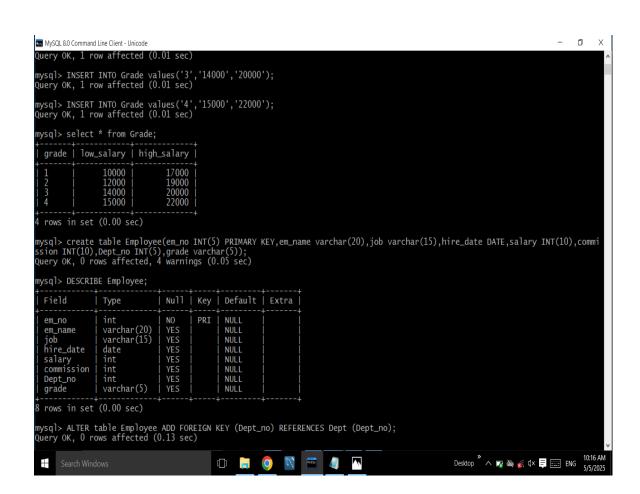
05/05/2025

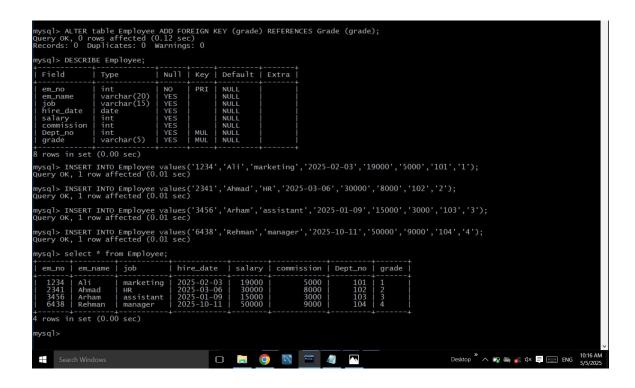
CREATE DATABASE, TABLES & INSERTING VALUES

Here are the syntaxes of mostly keys used here and before:

- ⇒ create database database_name;
- ⇒ use database_name;
- ⇒ create table table_name(attribute_name
 datatype(size),....);
- ⇒ INSERT INTO table_name values('values...')
- ⇒ ALTER table table_name ADD FOREIGN KEY (name) REFERENCES table_name (P.K);

```
mysql> INSERT INTO Dept values('104','employee_union');
Query OK, 1 row affected (0.01 sec)
 nysql> select * from Dept;
  Dept_no | Dept_name
              marketing
              finance
       104
             employee_union
  rows in set (0.00 sec)
mysql> create table Grade(grade varchar(5) PRIMARY KEY,low_salary INT(10),high_salary INT(10));
Query OK, O rows affected, 2 warnings (0.05 sec)
 nysql> DESCRIBE Grade;
  Field
                  Type
                                 Null | Key | Default | Extra
                   varchar(5)
                                  NO
                                          PRI
                                                  NULL
  grade
   low_salary
                   int
  high_salary
                   int
  rows in set (0.00 sec)
mysql> INSERT INTO Grade values('1','10000','17000');
Query OK, 1 row affected (0.01 sec)
 nysql> INSERT INTO Grade values('2','12000','19000');
                                               Desktop <sup>≫</sup> ∧ 🕝 🌭 🌾 ជx 📮 🚃 ENG
```

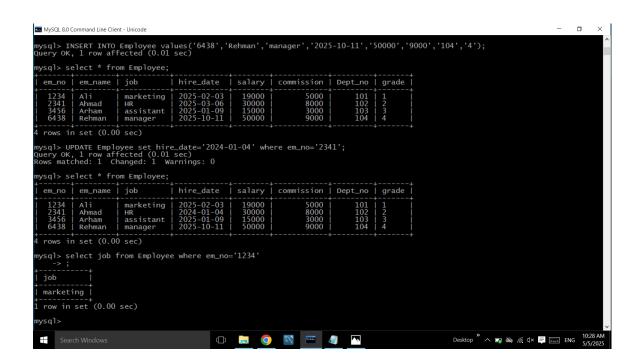




SELECTING ONE COLUMN FROM TABLE

⇒ To select only one column from a table use the syntax follows:

Select attribute1_name from table_name where attribute2_name='---';



PERFORM ARITHMETIC & RELATIONAL OPERATIONS

To perform the arithmetic operations (+,-,*,/) and relational operations (>,<,=,!=,<=,>=) use the following syntaxes:

⇒ select attribute_name + value AS annual from table_name;

USE OF AS

Here **AS** annual keyword is used for performing operations on all values of attribute.

USE OF WHERE CLAUSE

We can use this syntax for each operator. The **where clause** is used to point that attribute or table on which we are working.

```
mysql> select salary*10 from Employee;

| salary*10 |
| 190000 |
| 190000 |
| 150000 |
| 500000 |
| 4 rows in set (0.00 sec)
| mysql> select em_name from Employee where salary>19000;
| em_name |
| Ahmad | Rehman |
| 7 rows in set (0.00 sec)
| mysql> select em_name from Employee where salary<19000;
| em_name |
| Arham |
| Arham |
| Arham |
| I row in set (0.00 sec)
| mysql> select em_name from Employee where salary=19000;
| em_name |
| Arham |
| Arham |
| I row in set (0.00 sec)
| mysql> select em_name from Employee where salary=19000;
| em_name |
| Ali |
| I row in set (0.00 sec)
| mysql> select em_name from Employee where salary=19000;
| em_name |
| Ali |
| I row in set (0.00 sec)
| mysql> select em_name from Employee where salary=19000;
| em_name |
| Ali |
| I row in set (0.00 sec)
| mysql> select em_name from Employee where salary=19000;
| em_name |
| Ali |
| I row in set (0.00 sec)
```

USE OF ORDER BY FOR SORTING

ORDER BY keyword is used to sort the values of tables in ascending or descending order. By default ascending order is set. Otherwise for sorting the records in descending order DESC order is used. For ascending **ASC** and descending **DESC** is used.

Syntaxes of above two orders are:

- ⇒ select * from table_name ORDER BY attribute_name DESC;
- ⇒ select * from table_name ORDER BY attribute_name DESC;



USE OF DINTINCT KEYWORD

Distinct keyword is used to retrieve unique values from a specified column or set of columns. It eliminates duplicate records.

```
mysql> select DISTINCT commission from Employee;
| commission |
| 5000 |
| 8000 |
| 3000 |
| 9000 |
4 rows in set (0.01 sec)
```

