Ex.No. 2

Date:02/05/2023

**DDL Commands with Constraints – PRIMARY, FOREIGN KEY, UNIQUE, CHECK**

**AIM:**

To add the constraints like primary key, foreign key, unique key and check using DDL commands.

**Description:**

**PRIMARY KEY:**

The PRIMARY KEY constraint uniquely identifies each record in a database table.

Primary keys must contain UNIQUE values, and cannot contain NULL values.

A table can have only one primary key, which may consist of single or multiple fields.

**FOREIGN KEY:**

A FOREIGN KEY is a key used to link two tables together.

A FOREIGN KEY is a field (or collection of fields) in one table that refers to the PRIMARY KEY in another table.

The table containing the foreign key is called the child table, and the table containing the candidate key is called the referenced or parent table.

**UNIQUE Constraint:**

The UNIQUE constraint ensures that all values in a column are different.

Both the UNIQUE and PRIMARY KEY constraints provide a guarantee for uniqueness for a column or set of columns.

A PRIMARY KEY constraint automatically has a UNIQUE constraint.

However, you can have many UNIQUE constraints per table, but only one PRIMARY KEY constraint per table.

**CHECK Constraint:**

The CHECK constraint is used to limit the value range that can be placed in a column

If you define a CHECK constraint on a single column it allows only certain values for this column.

If you define a CHECK constraint on a table it can limit the values in certain columns based on values in other columns in the row.

**PRIMARY:**

ALTER TABLE table\_name

ADD PRIMARY KEY(primary\_key\_column);

**FOREIGN KEY:**

ALTER TABLE table\_name

ADD CONSTRAINT constraint\_name

FOREIGN KEY foreign\_key\_name (columns)

REFERENCES parent\_table(columns)

ON DELETE action

ON UPDATE action

**UNIQUE:**

CREATE TABLE table\_1(

...

column\_name\_1 data\_type,

...

UNIQUE(column\_name\_1)

);

**CHECK**

CREATE TABLE IF NOT EXISTS parts (

part\_no VARCHAR(18) PRIMARY KEY,

description VARCHAR(40),

cost DECIMAL(10 , 2 ) NOT NULL CHECK(cost > 0), price DECIMAL (10,2) NOT NULL

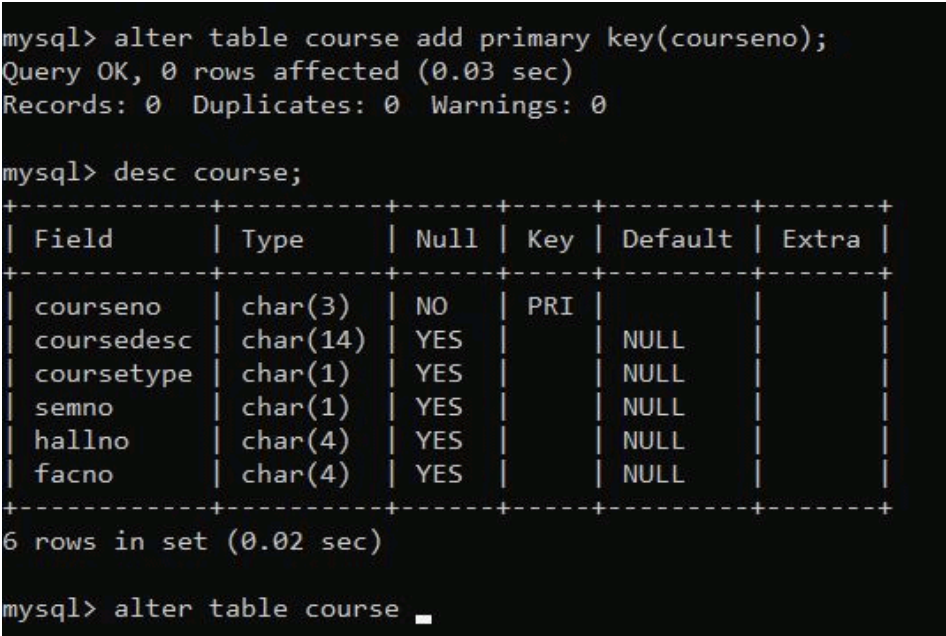
);

**Questions:**

1) Alter the table STUDENT with following structure.

|  |  |  |
| --- | --- | --- |
|  | Column | Constraints |
| # | Name |  |
|  |  | PRIMARY |
| 1 | RegNo | KEY |
| 2 | MobileNo | NOT NULL |

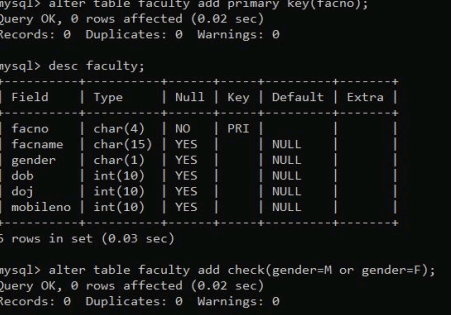
**OUTPUT:**



2)Alter the table name FACULTY with following structure. The DeptNo in this table refers the DeptNo in the DEPARTMENT table

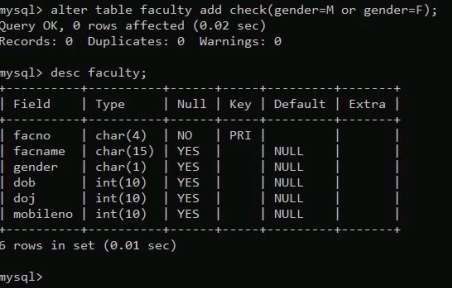
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | Column | | Constraints |  |  |  |  |
|  | # |  | Name | |  |  |  |  |  |
|  |  |  | FacNo | | PRIMARY |  |  |  |  |
|  | 1 |  | KEY |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  | Gender | | CHECK |  |  |  |  |
|  | 2 |  | ‘M’ or ‘F’ |  |  |  |  |
|  |  |  |  |  |  |  |  |

**OUTPUT:**



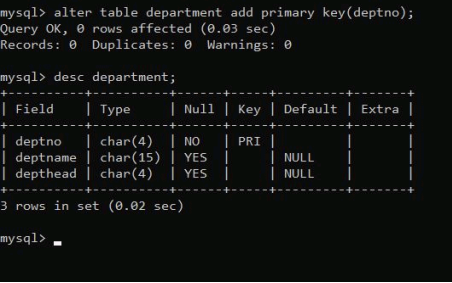
|  |  |
| --- | --- |
| 3) | After the FACULTY table is successfully created, test if you can add a constraint |
| FOREIGN KEY to the DeptNo of this table. | |

**OUTPUT:**



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 4) | Alter the table name DEPARTMENT with following structure. | | | | | | |
|  |  |  |  |  | |  |  |
|  |  | Column |  | Constraint | |  |  |
| # |  | Name |  |  |  |  |  |
|  |  | DeptNo |  | PRIMARY | |  |  |
| 1 |  |  | KEY | |  |  |
|  |  |  |  |  |

**OUTPUT:**



|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 5) | Alter the table name COURSE with following structure. | | | | | | |  |
|  |  |  |  | |  | | |  |  |
|  |  |  | Column |  | Constraint | | |  |  |
|  | # |  | Name |  |  |  |  |  |  |
|  |  |  | CourseNo |  | PRIMARY | | |  |  |
|  | 1 |  |  | KEY | | |  |  |
|  |  |  |  |  |  |
|  | 2 |  | SemNo |  | 1 to 6 | | |  |  |

**OUTPUT:**

