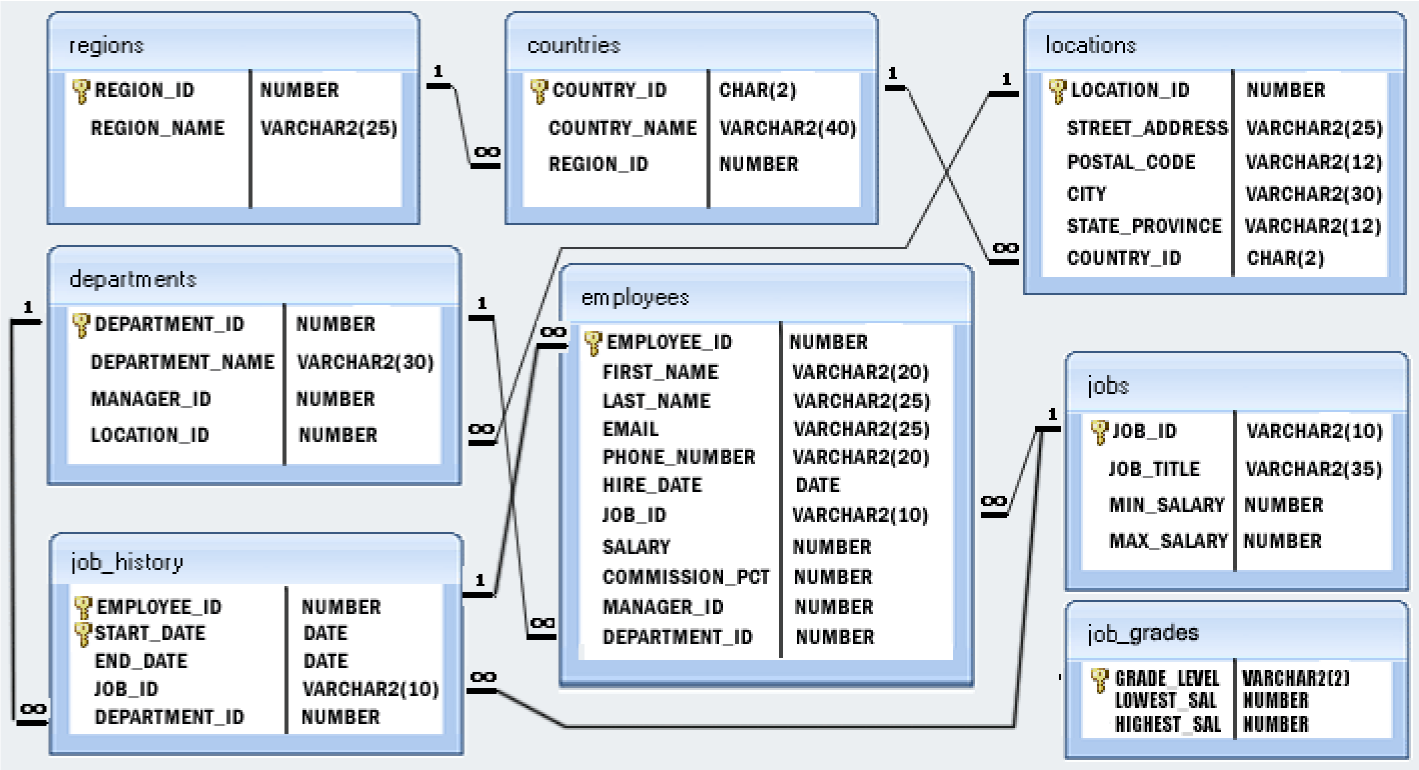
CREATE Statement Examples

The Human Resources (HR) Database

This sample database consists of 8 tables, as depicted in the following Entity-Relationship (ER) Diagram.



regions

REGION_ID	NUMBER
REGION_NAME	VARCHAR2(25)

countries

COUNTRY_ID	CHAR(2)
COUNTRY_NAME	VARCHAR2(40)
REGION_ID	NUMBER

locations

LOCATION_ID	NUMBER
STREET_ADDRESS	VARCHAR2(25)
POSTAL_CODE	VARCHAR2(12)
CITY	VARCHAR2(30)
STATE_PROVINCE	VARCHAR2(12)
COUNTRY_ID	CHAR(2)

departments

DEPARTMENT_ID	NUMBER
DEPARTMENT_NAME	VARCHAR2(30)
MANAGER_ID	NUMBER
LOCATION_ID	NUMBER

employees

EMPLOYEE_ID	NUMBER
FIRST_NAME	VARCHAR2(20)
LAST_NAME	VARCHAR2(25)
EMAIL	VARCHAR2(25)
PHONE_NUMBER	VARCHAR2(20)
HIRE_DATE	DATE
JOB_ID	VARCHAR2(10)
SALARY	NUMBER
COMMISSION_PCT	NUMBER
MANAGER_ID	NUMBER
DEPARTMENT_ID	NUMBER

jobs

JOB_ID	VARCHAR2(10)
JOB_TITLE	VARCHAR2(35)
MIN_SALARY	NUMBER
MAX_SALARY	NUMBER

job_history

EMPLOYEE_ID	NUMBER
START_DATE	DATE
END_DATE	DATE
JOB_ID	VARCHAR2(10)
DEPARTMENT_ID	NUMBER

job_grades

GRADE_LEVEL	VARCHAR2(2)
LOWEST_SAL	NUMBER
HIGHEST_SAL	NUMBER

Run the SQL Scripts

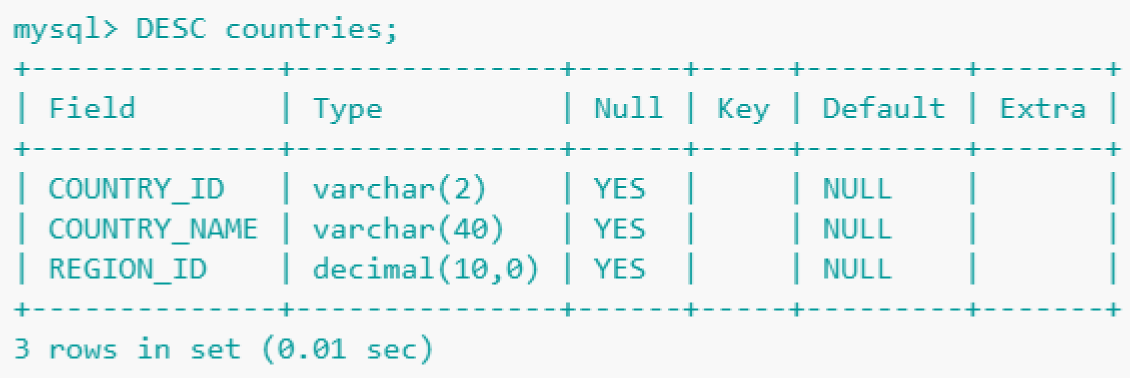
Before we proceed any further, please go ahead and import the SQL Script ([click here (https://bryanuniversity.instructure.com/courses/10152/files/744673/download?wrap=1)](https://bryanuniversity.instructure.com/courses/10152/files/744673/download?wrap=1) [(https://bryanuniversity.instructure.com/courses/10152/files/744673/download?wrap=1)](https://bryanuniversity.instructure.com/courses/10152/files/744673/download?wrap=1) to download the script file) into MySQL Work Bench and/or PostgreSQL applications.

Create Table Statements

- Write an SQL statement to create a simple table countries including columns country_id,country_name and region_id.

```
CREATE TABLE countries(
     COUNTRY_ID varchar(2),
     COUNTRY_NAME varchar(40),
     REGION_ID decimal(10,0)
);
```

Sample Output:

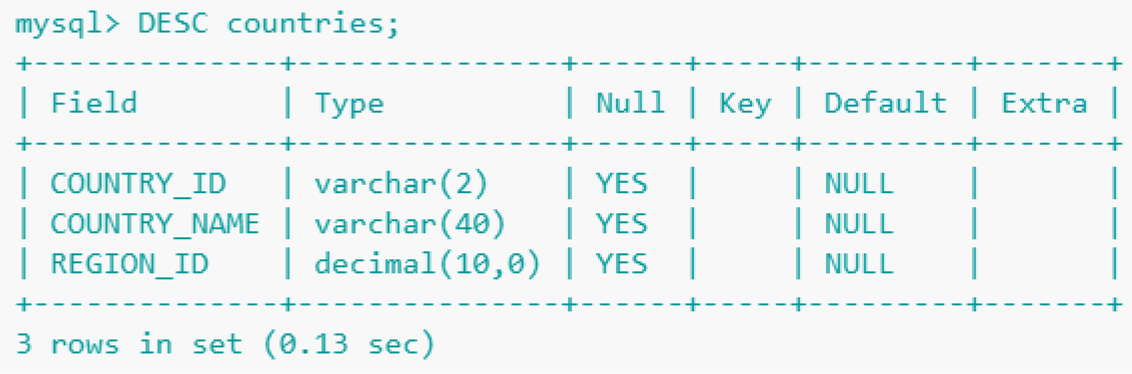


```
mysql> DESC countries;
+--------------+---------------+------+-----+---------+-------+
| Field        | Type          | Null | Key | Default | Extra |
+--------------+---------------+------+-----+---------+-------+
| COUNTRY_ID   | varchar(2)    | YES  |     | NULL    |       |
| COUNTRY_NAME | varchar(40)   | YES  |     | NULL    |       |
| REGION_ID    | decimal(10,0) | YES  |     | NULL    |       |
+--------------+---------------+------+-----+---------+-------+
3 rows in set (0.01 sec)
```

- Write an SQL statement to create a simple table countries including columns country_id,country_name and region_id which is already exists.

```
CREATE TABLE IF NOT EXISTS countries (
     COUNTRY_ID varchar(2),
     COUNTRY_NAME varchar(40),
     REGION_ID decimal(10,0)
);
```

Sample Output:

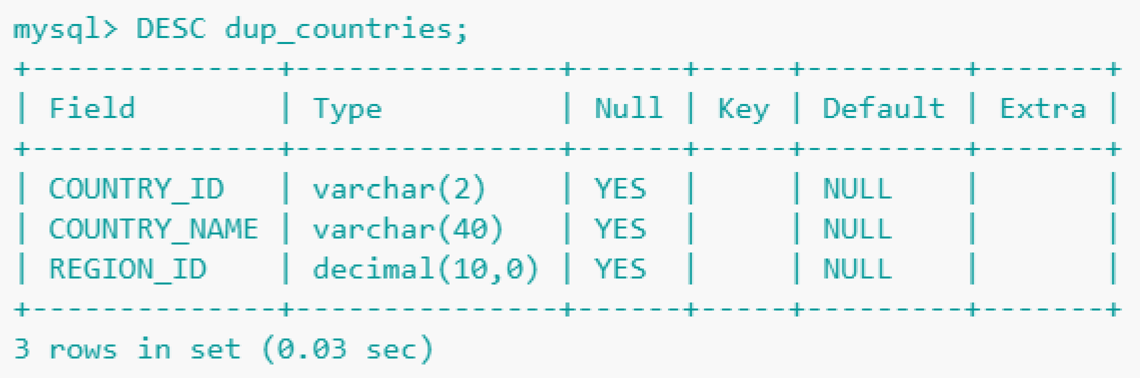


```
mysql> DESC countries;
+--------------+---------------+------+-----+---------+-------+
| Field        | Type          | Null | Key | Default | Extra |
+--------------+---------------+------+-----+---------+-------+
| COUNTRY_ID   | varchar(2)    | YES  |     | NULL    |       |
| COUNTRY_NAME | varchar(40)   | YES  |     | NULL    |       |
| REGION_ID    | decimal(10,0) | YES  |     | NULL    |       |
+--------------+---------------+------+-----+---------+-------+
3 rows in set (0.13 sec)
```

- Write a SQL statement to create the structure of a table dup_countries like countries.

```
CREATE TABLE IF NOT EXISTS dup_countries
LIKE countries;
```

Sample Output:

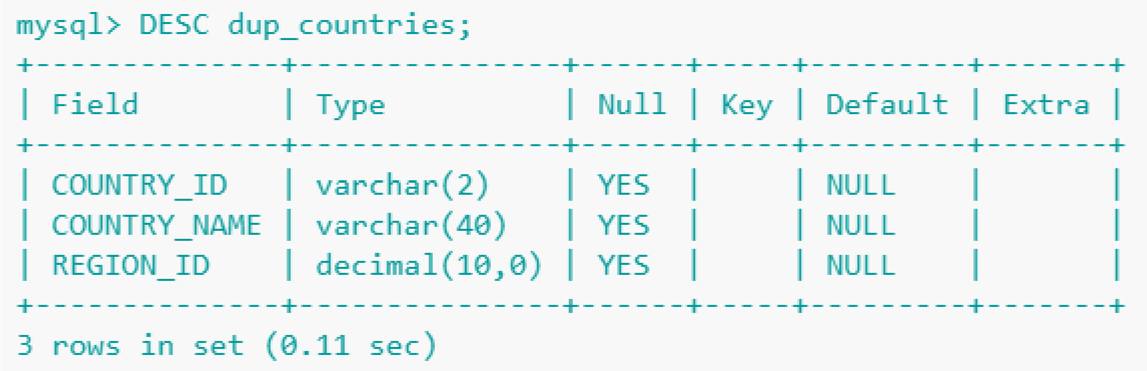


```
mysql> DESC dup_countries;
+--------------+---------------+------+-----+---------+-------+
| Field        | Type          | Null | Key | Default | Extra |
+--------------+---------------+------+-----+---------+-------+
| COUNTRY_ID   | varchar(2)    | YES  |     | NULL    |       |
| COUNTRY_NAME | varchar(40)   | YES  |     | NULL    |       |
| REGION_ID    | decimal(10,0) | YES  |     | NULL    |       |
+--------------+---------------+------+-----+---------+-------+
3 rows in set (0.03 sec)
```

- Write a SQL statement to create a duplicate copy of countries table including structure and data by name dup_countries.

```
CREATE TABLE IF NOT EXISTS dup_countries
AS SELECT * FROM countries;
```

Sample Output:

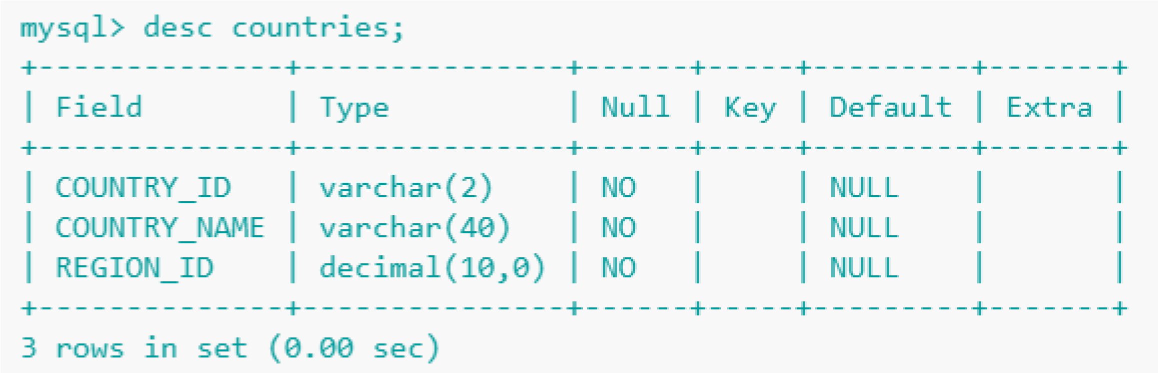


```
mysql> DESC dup_countries;
+--------------+---------------+------+-----+---------+-------+
| Field        | Type          | Null | Key | Default | Extra |
+--------------+---------------+------+-----+---------+-------+
| COUNTRY_ID   | varchar(2)    | YES  |     | NULL    |       |
| COUNTRY_NAME | varchar(40)   | YES  |     | NULL    |       |
| REGION_ID    | decimal(10,0) | YES  |     | NULL    |       |
+--------------+---------------+------+-----+---------+-------+
3 rows in set (0.11 sec)
```

- Write a SQL statement to create a table countries set a constraint NULL.

```
CREATE TABLE IF NOT EXISTS countries (
     COUNTRY_ID varchar(2) NOT NULL,
     COUNTRY_NAME varchar(40) NOT NULL,
     REGION_ID decimal(10,0) NOT NULL
);
```

Sample Output:

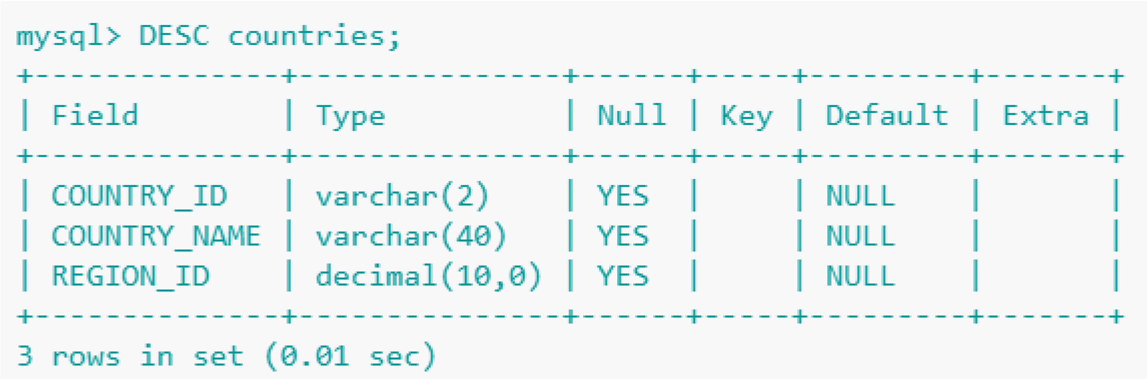


```
mysql> desc countries;
+--------------+---------------+------+-----+---------+-------+
| Field        | Type          | Null | Key | Default | Extra |
+--------------+---------------+------+-----+---------+-------+
| COUNTRY_ID   | varchar(2)    | NO   |     | NULL    |       |
| COUNTRY_NAME | varchar(40)   | NO   |     | NULL    |       |
| REGION_ID    | decimal(10,0) | NO   |     | NULL    |       |
+--------------+---------------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```

- Write a SQL statement to create a table named countries including columns country_id, country_name and region_id and make sure that no countries except Italy, India and China will be entered in the table.

```
CREATE TABLE IF NOT EXISTS countries (
     COUNTRY_ID varchar(2),
     COUNTRY_NAME varchar(40)
     CHECK(COUNTRY_NAME IN('Italy','India','China')) ,
     REGION_ID decimal(10,0)
);
```

Sample Output:

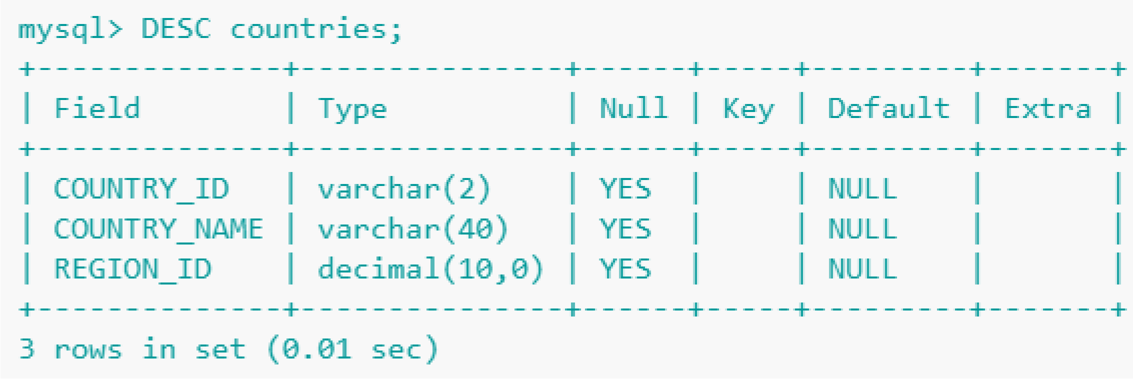


```
mysql> DESC countries;
+--------------+---------------+------+-----+---------+-------+
| Field        | Type          | Null | Key | Default | Extra |
+--------------+---------------+------+-----+---------+-------+
| COUNTRY_ID   | varchar(2)    | YES  |     | NULL    |       |
| COUNTRY_NAME | varchar(40)   | YES  |     | NULL    |       |
| REGION_ID    | decimal(10,0) | YES  |     | NULL    |       |
+--------------+---------------+------+-----+---------+-------+
3 rows in set (0.01 sec)
```

- Write a SQL statement to create a table named countries including columns country_id,country_name and region_id and make sure that no duplicate data against column country_id will be allowed at the time of insertion.

```
CREATE TABLE IF NOT EXISTS countries (
     COUNTRY_ID varchar(2) NOT NULL,
     COUNTRY_NAME varchar(40) NOT NULL,
     REGION_ID decimal(10,0) NOT NULL,
     UNIQUE(COUNTRY_ID)
);
```

Sample Output:

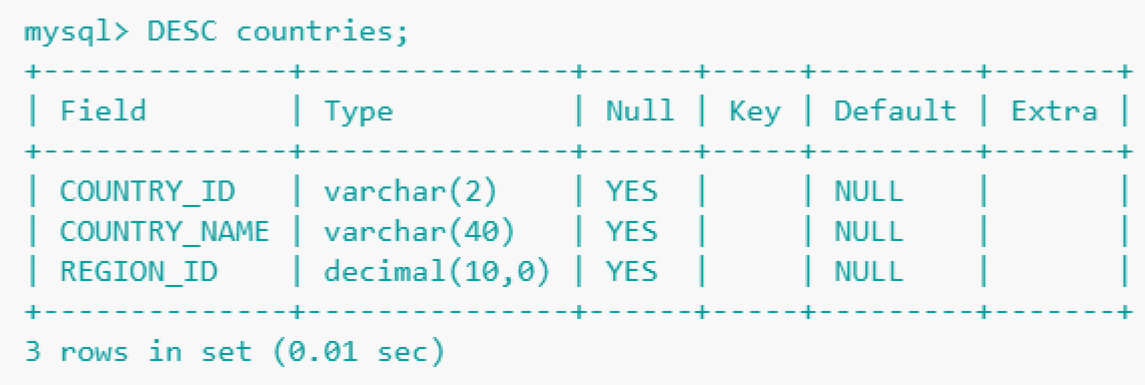


```
mysql> DESC countries;
+--------------+---------------+------+-----+---------+-------+
| Field        | Type          | Null | Key | Default | Extra |
+--------------+---------------+------+-----+---------+-------+
| COUNTRY_ID   | varchar(2)    | YES  |     | NULL    |       |
| COUNTRY_NAME | varchar(40)   | YES  |     | NULL    |       |
| REGION_ID    | decimal(10,0) | YES  |     | NULL    |       |
+--------------+---------------+------+-----+---------+-------+
3 rows in set (0.01 sec)
```

- Write a SQL statement to create a table named countries including columns country_id, country_name and region_id and make sure that the country_id column will be a key field which will not contain any duplicate data at the time of insertion.

```
CREATE TABLE IF NOT EXISTS countries (
     COUNTRY_ID varchar(2) NOT NULL UNIQUE PRIMARY KEY,
     COUNTRY_NAME varchar(40) NOT NULL,
     REGION_ID decimal(10,0) NOT NULL
);
```

Sample Output:



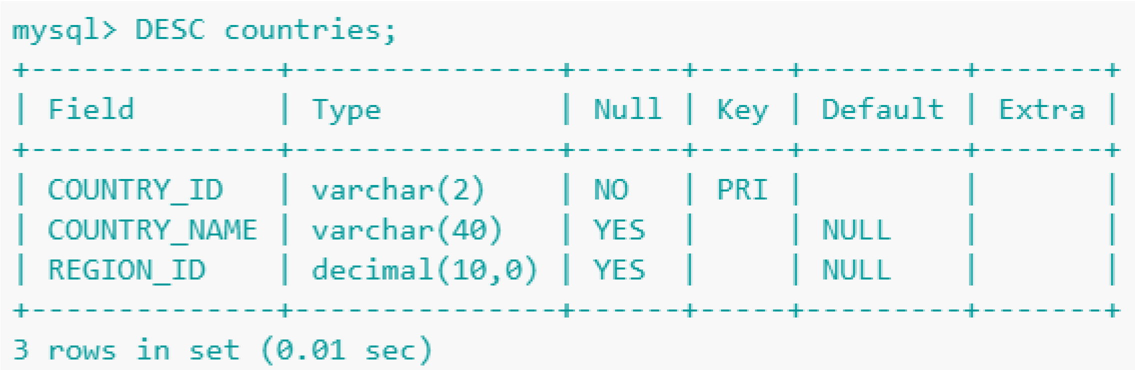
```
mysql> DESC countries;
+--------------+---------------+------+-----+---------+-------+
| Field        | Type          | Null | Key | Default | Extra |
+--------------+---------------+------+-----+---------+-------+
| COUNTRY_ID   | varchar(2)    | YES  |     | NULL    |       |
| COUNTRY_NAME | varchar(40)   | YES  |     | NULL    |       |
| REGION_ID    | decimal(10,0) | YES  |     | NULL    |       |
+--------------+---------------+------+-----+---------+-------+
3 rows in set (0.01 sec)
```

- Write a SQL statement to create a table “countries” including columns country_id, country_name and region_id and make sure that the column country_id will be unique and store an auto incremented value.

```
CREATE TABLE IF NOT EXISTS countries (
     COUNTRY_ID integer NOT NULL UNIQUE AUTO_INCREMENT PRIMARY KEY,
     COUNTRY_NAME varchar(40) NOT NULL,
     REGION_ID decimal(10,0) NOT NULL
);

DESC countries;
```

Sample Output:

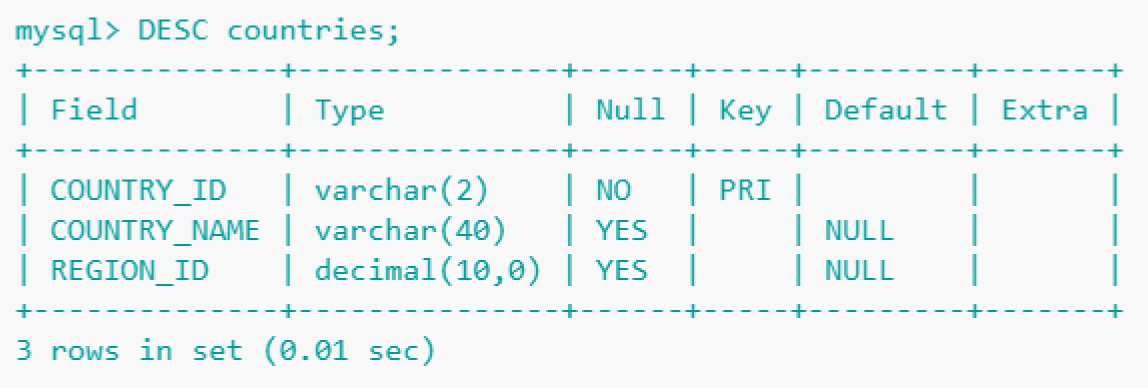


```
mysql> DESC countries;
+--------------+---------------+------+-----+---------+-------+
| Field        | Type          | Null | Key | Default | Extra |
+--------------+---------------+------+-----+---------+-------+
| COUNTRY_ID   | varchar(2)    | NO   | PRI |         |       |
| COUNTRY_NAME | varchar(40)   | YES  |     | NULL    |       |
| REGION_ID    | decimal(10,0) | YES  |     | NULL    |       |
+--------------+---------------+------+-----+---------+-------+
3 rows in set (0.01 sec)
```

- Write a SQL statement to create a table “countries” including columns country_id, country_name and region_id and make sure that the combination of columns country_id and region_id will be unique.

```
CREATE TABLE IF NOT EXISTS countries (
     COUNTRY_ID varchar(2) NOT NULL UNIQUE DEFAULT '',
     COUNTRY_NAME varchar(40) DEFAULT NULL,
     REGION_ID decimal(10,0) NOT NULL,
PRIMARY KEY (COUNTRY_ID,REGION_ID));
```

Sample Output:

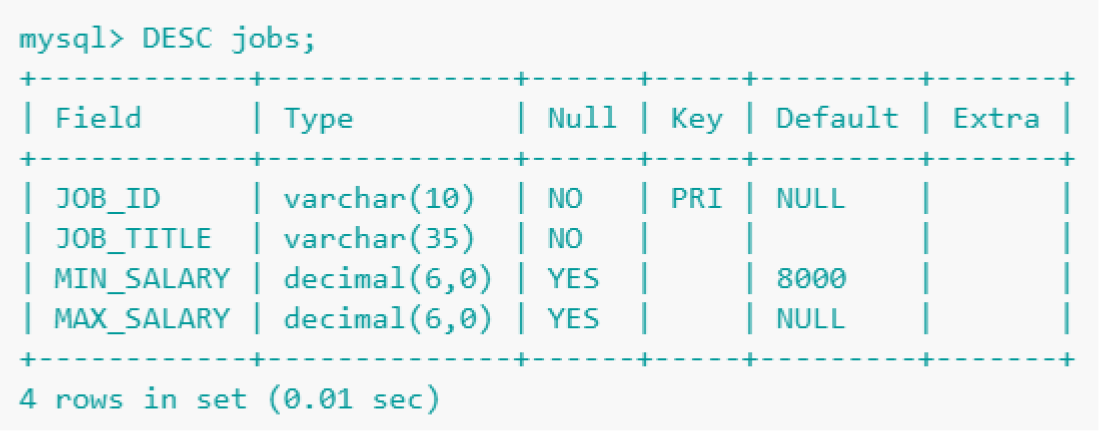


```
mysql> DESC countries;
+--------------+---------------+------+-----+---------+-------+
| Field        | Type          | Null | Key | Default | Extra |
+--------------+---------------+------+-----+---------+-------+
| COUNTRY_ID   | varchar(2)    | NO   | PRI |         |       |
| COUNTRY_NAME | varchar(40)   | YES  |     | NULL    |       |
| REGION_ID    | decimal(10,0) | YES  |     | NULL    |       |
+--------------+---------------+------+-----+---------+-------+
3 rows in set (0.01 sec)
```

- Write a SQL statement to create a table named jobs including columns job_id, job_title, min_salary and max_salary, and make sure that, the default value for job_title is blank and min_salary is 8000 and max_salary is NULL will be entered automatically at the time of insertion if no value assigned for the specified columns.

```
CREATE TABLE IF NOT EXISTS jobs (
     JOB_ID varchar(10) NOT NULL UNIQUE,
     JOB_TITLE varchar(35) NOT NULL DEFAULT ' ',
     MIN_SALARY decimal(6,0) DEFAULT 8000,
     MAX_SALARY decimal(6,0) DEFAULT NULL
);
```

Sample Output:

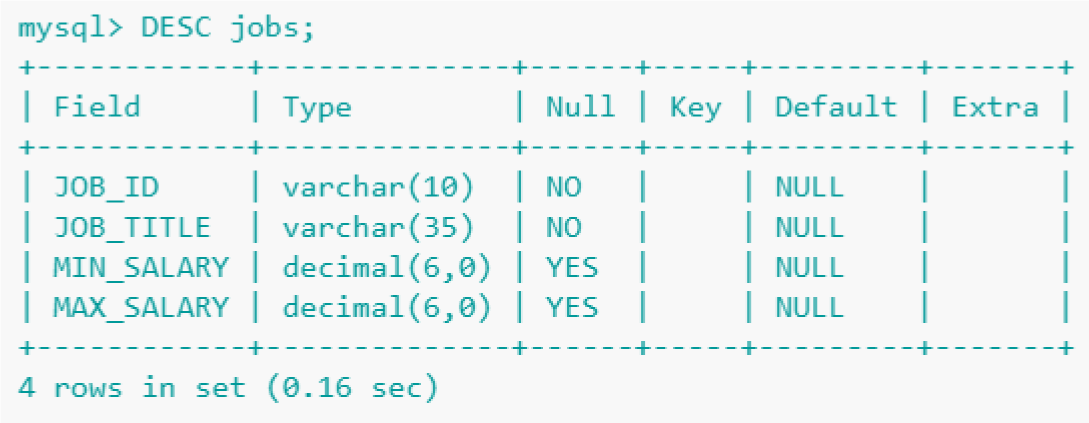


```
mysql> DESC jobs;
+------------+--------------+------+-----+---------+-------+
| Field      | Type         | Null | Key | Default | Extra |
+------------+--------------+------+-----+---------+-------+
| JOB_ID     | varchar(10)  | NO   | PRI | NULL    |       |
| JOB_TITLE  | varchar(35)  | NO   |     |         |       |
| MIN_SALARY | decimal(6,0) | YES  |     | 8000    |       |
| MAX_SALARY | decimal(6,0) | YES  |     | NULL    |       |
+------------+--------------+------+-----+---------+-------+
4 rows in set (0.01 sec)
```

- Write a SQL statement to create a table named jobs including columns job_id, job_title, min_salary, max_salary and check whether the max_salary amount exceeding the upper limit 25000.

```
CREATE TABLE IF NOT EXISTS jobs (
     JOB_ID varchar(10) NOT NULL ,
     JOB_TITLE varchar(35) NOT NULL,
     MIN_SALARY decimal(6,0),
     MAX_SALARY decimal(6,0)
     CHECK(MAX_SALARY<=25000)
);
```

Sample Output:

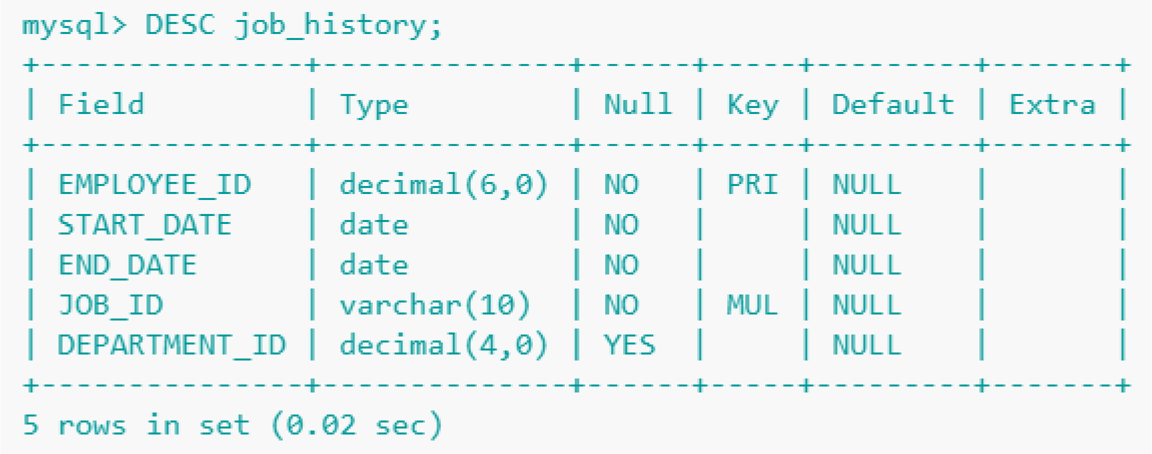


```
mysql> DESC jobs;
+------------+--------------+------+-----+---------+-------+
| Field      | Type         | Null | Key | Default | Extra |
+------------+--------------+------+-----+---------+-------+
| JOB_ID     | varchar(10)  | NO   |     | NULL    |       |
| JOB_TITLE  | varchar(35)  | NO   |     | NULL    |       |
| MIN_SALARY | decimal(6,0) | YES  |     | NULL    |       |
| MAX_SALARY | decimal(6,0) | YES  |     | NULL    |       |
+------------+--------------+------+-----+---------+-------+
4 rows in set (0.16 sec)
```

- Write a SQL statement to create a table job_history including columns employee_id, start_date, end_date, job_id and department_id and make sure that, the employee_id column does not contain any duplicate value at the time of insertion and the foreign key column job_id contain only those values which are exists in the jobs table. Make sure that the value against column end_date will be entered at the time of insertion to the format like '--/--/----'.

```
CREATE TABLE job_history (
     EMPLOYEE_ID decimal(6,0) NOT NULL PRIMARY KEY,
     START_DATE date NOT NULL,
     END_DATE date NOT NULL
     CHECK (END_DATE LIKE '--/--/----'),
     JOB_ID varchar(10) NOT NULL,
     DEPARTMENT_ID decimal(4,0) DEFAULT NULL,
     FOREIGN KEY (job_id) REFERENCES jobs(job_id)
);
```

Sample Output:

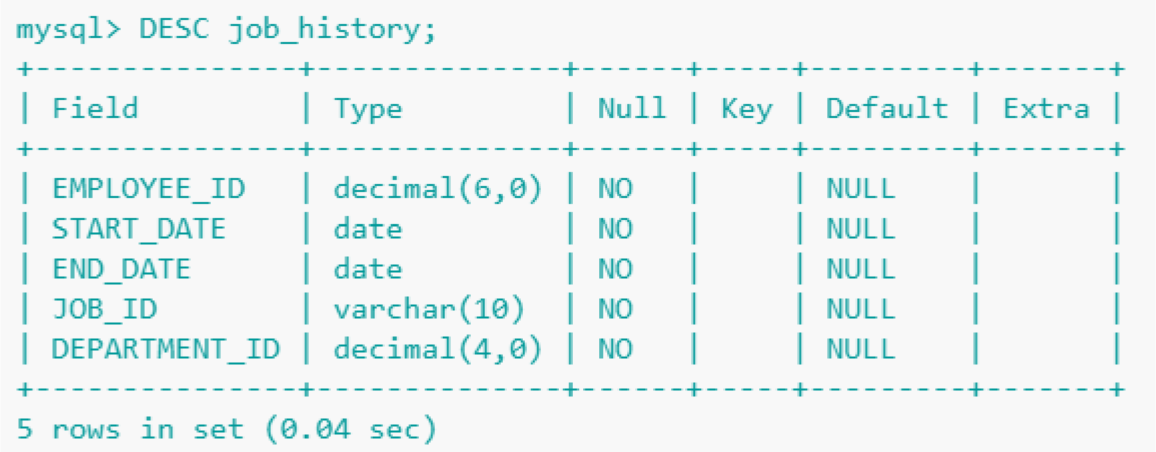


```
mysql> DESC job_history;
+---------------+--------------+------+-----+---------+-------+
| Field         | Type         | Null | Key | Default | Extra |
+---------------+--------------+------+-----+---------+-------+
| EMPLOYEE_ID   | decimal(6,0) | NO   | PRI | NULL    |       |
| START_DATE    | date         | NO   |     | NULL    |       |
| END_DATE      | date         | NO   |     | NULL    |       |
| JOB_ID        | varchar(10)  | NO   | MUL | NULL    |       |
| DEPARTMENT_ID | decimal(4,0) | YES  |     | NULL    |       |
+---------------+--------------+------+-----+---------+-------+
5 rows in set (0.02 sec)
```

- Write a SQL statement to create a table named job_history including columns employee_id, start_date, end_date, job_id and department_id and make sure that the value against column end_date will be entered at the time of insertion to the format like '--/--/----'.

```
CREATE TABLE IF NOT EXISTS job_history (
     EMPLOYEE_ID decimal(6,0) NOT NULL,
     START_DATE date NOT NULL,
     END_DATE date NOT NULL
     CHECK (END_DATE LIKE '--/--/----'),
     JOB_ID varchar(10) NOT NULL,
     DEPARTMENT_ID decimal(4,0) NOT NULL
);
```

Sample Output:

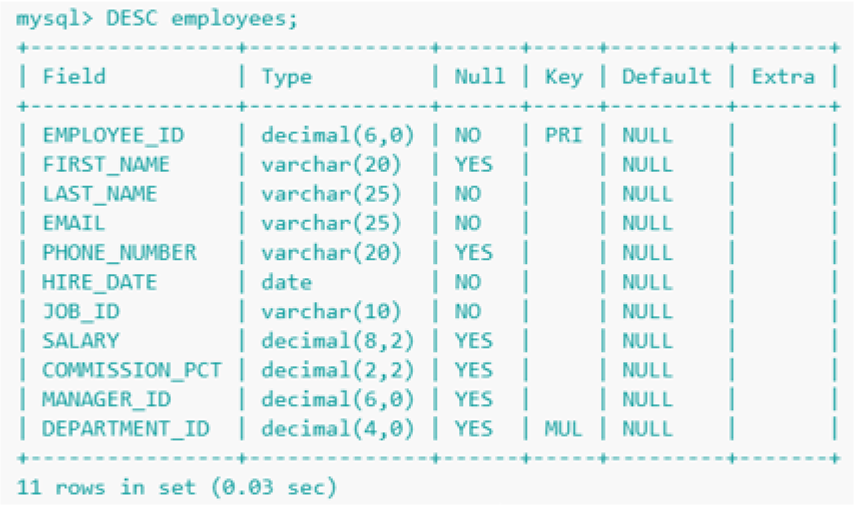


```
mysql> DESC job_history;
+---------------+--------------+------+-----+---------+-------+
| Field         | Type         | Null | Key | Default | Extra |
+---------------+--------------+------+-----+---------+-------+
| EMPLOYEE_ID   | decimal(6,0) | NO   |     | NULL    |       |
| START_DATE    | date         | NO   |     | NULL    |       |
| END_DATE      | date         | NO   |     | NULL    |       |
| JOB_ID        | varchar(10)  | NO   |     | NULL    |       |
| DEPARTMENT_ID | decimal(4,0) | NO   |     | NULL    |       |
+---------------+--------------+------+-----+---------+-------+
5 rows in set (0.04 sec)
```

- Write a SQL statement to create a table employees including columns employee_id, first_name, last_name, email, phone_number hire_date, job_id, salary, commission, manager_id and department_id and make sure that, the employee_id column does not contain any duplicate value at the time of insertion and the foreign key columns combined by department_id and manager_id columns contain only those unique combination values, which combinations are exists in the departments table.

```
CREATE TABLE IF NOT EXISTS employees (
     EMPLOYEE_ID decimal(6,0) NOT NULL PRIMARY KEY,
     FIRST_NAME varchar(20) DEFAULT NULL,
     LAST_NAME varchar(25) NOT NULL,
     EMAIL varchar(25) NOT NULL,
     PHONE_NUMBER varchar(20) DEFAULT NULL,
     HIRE_DATE date NOT NULL,
     JOB_ID varchar(10) NOT NULL,
     SALARY decimal(8,2) DEFAULT NULL,
     COMMISSION_PCT decimal(2,2) DEFAULT NULL,
     MANAGER_ID decimal(6,0) DEFAULT NULL,
     DEPARTMENT_ID decimal(4,0) DEFAULT NULL,
     FOREIGN KEY(DEPARTMENT_ID, MANAGER_ID)
     REFERENCES departments(DEPARTMENT_ID,MANAGER_ID)
);
```

Sample Output:



```
mysql> DESC employees;
+----------------+--------------+------+-----+---------+-------+
| Field          | Type         | Null | Key | Default | Extra |
+----------------+--------------+------+-----+---------+-------+
| EMPLOYEE_ID    | decimal(6,0) | NO   | PRI | NULL    |       |
| FIRST_NAME     | varchar(20)  | YES  |     | NULL    |       |
| LAST_NAME      | varchar(25)  | NO   |     | NULL    |       |
| EMAIL          | varchar(25)  | NO   |     | NULL    |       |
| PHONE_NUMBER   | varchar(20)  | YES  |     | NULL    |       |
| HIRE_DATE      | date         | NO   |     | NULL    |       |
| JOB_ID         | varchar(10)  | NO   |     | NULL    |       |
| SALARY         | decimal(8,2) | YES  |     | NULL    |       |
| COMMISSION_PCT | decimal(2,2) | YES  |     | NULL    |       |
| MANAGER_ID     | decimal(6,0) | YES  |     | NULL    |       |
| DEPARTMENT_ID  | decimal(4,0) | YES  | MUL | NULL    |       |
+----------------+--------------+------+-----+---------+-------+
11 rows in set (0.03 sec)
```