

CSLR 51 : DBMS LAB-6

Roll no. : **106119100**

Name : **Rajneesh Pandey**

Section : **CSE-B**

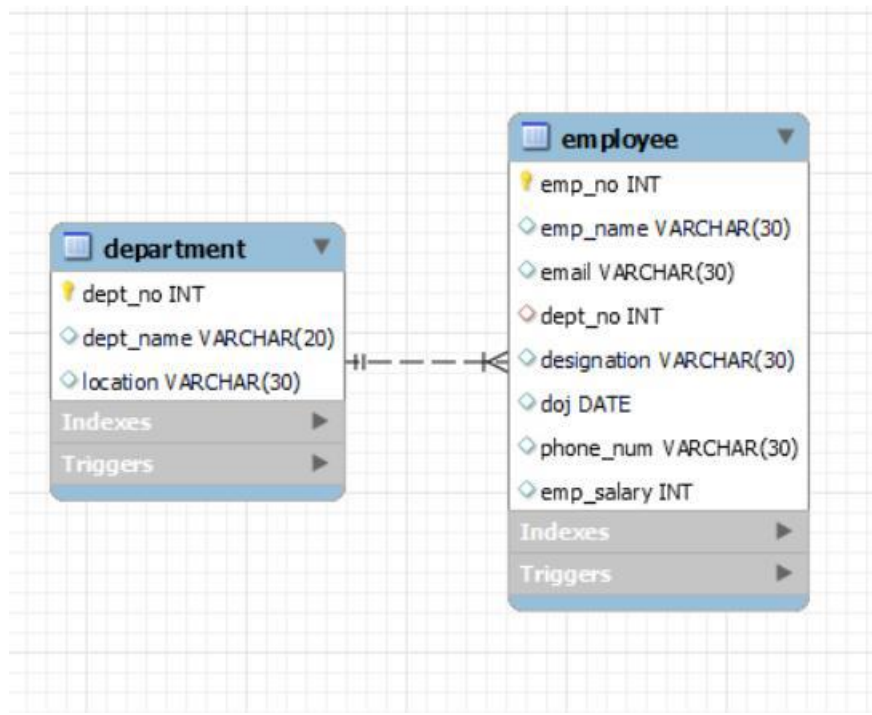
PROBLEM 1

1. Consider the following tables namely “DEPARTMENTS” and “EMPLOYEES”
Their schemas are as follows:

DEPARTMENTS (dept_no , dept_name , dept_location);

**EMPLOYEES (emp_id , emp_name , email, dept, designation, DOJ,
phone_num, emp_salary);**

ER-Diagram



```
CREATE DATABASE CompanyDB;  
USE CompanyDB;
```

```
CREATE TABLE Departments (  
    dept_id VARCHAR (20) NOT NULL PRIMARY KEY,  
    dept_name VARCHAR (20),  
    dept_location VARCHAR (20)  
);
```

```

CREATE TABLE Employees (
    emp_id INT NOT NULL PRIMARY KEY,
    emp_name VARCHAR(100),
    email VARCHAR(100),
    dept_id INT,
    designation VARCHAR(100),
    doj DATE,
    phone_num VARCHAR(45),
    emp_salary DECIMAL(12,3)
);

create table messages
(
    id int(11) not null auto_increment primary key,
    message varchar(255) default null,
    time timestamp null default current_timestamp on update current_t
imestamp
);

/** ----- Queries -----
----**/

-- Create a trigger for:-

-- a) Calculating the number of rows inserted so far.

delimiter //

create trigger trig1A
before insert on departments
for each row
set @dept_rows = @dept_rows+1//

set @dept_rows = 0//

delimiter ;

```

```

insert into departments
values
    ('1', 'sales', 'chennai'),
    ('2', 'marketing', 'bombay'),
    ('3', 'management', 'bengaluru');

```

```

SELECT @dept_rows;

```

```

MySQL localhost:3306 ssl SQL > use companydb;
Default schema set to 'companydb'.
Fetching table and column names from 'companydb' for auto-completion... Press ^C to stop.
MySQL localhost:3306 ssl companydb SQL > delimiter //
MySQL localhost:3306 ssl companydb SQL >
MySQL localhost:3306 ssl companydb SQL > create trigger trig1A
-> before insert on departments
-> for each row
-> set @dept_rows = @dept_rows+1//
Query OK, 0 rows affected (0.0138 sec)
MySQL localhost:3306 ssl companydb SQL >
MySQL localhost:3306 ssl companydb SQL > set @dept_rows = 0//
Query OK, 0 rows affected (0.0002 sec)
MySQL localhost:3306 ssl companydb SQL >
MySQL localhost:3306 ssl companydb SQL > delimiter ;
MySQL localhost:3306 ssl companydb SQL >
MySQL localhost:3306 ssl companydb SQL > insert into departments
-> values
->     ('1', 'sales', 'chennai'),
->     ('2', 'marketing', 'bombay'),
->     ('3', 'management', 'bengaluru');
Query OK, 3 rows affected (0.0086 sec)

Records: 3  Duplicates: 0  Warnings: 0
MySQL localhost:3306 ssl companydb SQL >
MySQL localhost:3306 ssl companydb SQL > SELECT @dept_rows;
+-----+
| @dept_rows |
+-----+
|          3 |
+-----+
1 row in set (0.0008 sec)

```

-- b) Displays message prior to insert operation on Employee Table

```

delimiter //

```

```

create trigger display_message before insert on employees
for each row
begin
insert into messages (message) values('Inserted Row in Employees'
);

```



```
end; //
```

```
delimiter ;
```

```
insert into employees  
values
```

```
    ('100','timber', 'timber@gmail.com','1', 'salseman', '2019-01-  
12', '9842042345', '100000'),  
    ('101','jaanu', 'jaanu@gmail.com', '2', 'senior manager', '2020-  
01-01', '9865161122', '15000'),  
    ('102','jen', 'jen@gmail.com', '3', 'junior manager', '2019-10-  
12', '9042842344', '60000'),  
    ('103','jack', 'jack@gmail.com', '2', 'assistant', '2016-03-  
01', '9498853055', '50000');
```

```
SELECT * FROM messages;
```

```
MySQL localhost:3306 ssl companydb SQL > delimiter //  
MySQL localhost:3306 ssl companydb SQL >  
MySQL localhost:3306 ssl companydb SQL > create trigger display_message before insert on employees  
-> for each row  
-> begin  
-> insert into messages (message) values('Inserted Row in Employees');  
-> end; //  
Query OK, 0 rows affected (0.0134 sec)  
MySQL localhost:3306 ssl companydb SQL >  
MySQL localhost:3306 ssl companydb SQL > delimiter ;  
MySQL localhost:3306 ssl companydb SQL >  
MySQL localhost:3306 ssl companydb SQL > insert into employees  
-> values  
-> ('100','timber', 'timber@gmail.com','1', 'salseman', '2019-01-12', '9842042345', '100000'),  
-> ('101','jaanu', 'jaanu@gmail.com', '2', 'senior manager', '2020-01-01', '9865161122', '15000'),  
-> ('102','jen', 'jen@gmail.com', '3', 'junior manager', '2019-10-12', '9042842344', '60000'),  
-> ('103','jack', 'jack@gmail.com', '2', 'assistant', '2016-03-01', '9498853055', '50000');  
Query OK, 4 rows affected (0.0109 sec)  
Records: 4 Duplicates: 0 Warnings: 0  
MySQL localhost:3306 ssl companydb SQL >  
MySQL localhost:3306 ssl companydb SQL > SELECT * FROM messages;  
+----+-----+-----+  
| id | message | time |  
+----+-----+-----+  
| 1 | Inserted Row in Employees | 2021-09-16 17:29:17 |  
| 2 | Inserted Row in Employees | 2021-09-16 17:29:17 |  
| 3 | Inserted Row in Employees | 2021-09-16 17:29:17 |  
| 4 | Inserted Row in Employees | 2021-09-16 17:29:17 |  
+----+-----+-----+  
4 rows in set (0.0007 sec)
```

-
- c) Ensuring that the Employee Table does not contain duplicates or null value in the Name column

```
delimiter //
```

```
create trigger NULL_val_trigger before insert on employees  
for each row  
begin
```

```

    if (exists(select 1 from employees where emp_name = new.emp_name
or new.emp_name is null))
    then
        signal sqlstate '45001' set message_text = "INSERT failed due to
duplicate or null emp_name";
    end if;
end//

insert into employees
    values ('104', 'jen', 'xjen@gmail.com', '3', 'junior manager', '201
9-09-10', '7561994181', '62000')//

delimiter ;
select * from employees;

```

```

MySQL localhost:3306 ssl companydb SQL > delimiter //
MySQL localhost:3306 ssl companydb SQL > create trigger NULL_val_trigger before insert on employees
-> for each row
-> begin
->     if (exists(select 1 from employees where emp_name = new.emp_name or new.emp_name is null))
->     then
->         signal sqlstate '45001' set message_text = 'INSERT failed due to duplicate or null emp_name';
->     end if;
-> end//
Query OK, 0 rows affected (0.0116 sec)
MySQL localhost:3306 ssl companydb SQL >
MySQL localhost:3306 ssl companydb SQL > insert into employees
//
->     values ('104', 'jen', 'xjen@gmail.com', '3', 'junior manager', '2019-09-10', '7561994181', '62000')
ERROR: 1644 (45001): INSERT failed due to duplicate or null emp_name
MySQL localhost:3306 ssl companydb SQL >
MySQL localhost:3306 ssl companydb SQL > delimiter ;
MySQL localhost:3306 ssl companydb SQL > select * from employees;
+-----+-----+-----+-----+-----+-----+-----+-----+
| emp_id | emp_name | email          | dept_id | designation    | doj          | phone_num | emp_salary |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 100    | timber   | timber@gmail.com | 1       | salseman       | 2019-01-12   | 9842042345 | 100000.000 |
| 101    | jaanu    | jaanu@gmail.com  | 2       | senior manager | 2020-01-01   | 9865161122 | 15000.000  |
| 102    | jen      | jen@gmail.com    | 3       | junior manager | 2019-10-12   | 9042842344 | 60000.000  |
| 103    | jack     | jack@gmail.com   | 2       | assistant      | 2016-03-01   | 9498853055 | 50000.000  |
+-----+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.0010 sec)

```

-

- d) To insert Employee details into Employee Table only of DOJ > 2018

```

DELIMITER //
CREATE TRIGGER NOT_BELOW_2018
    BEFORE INSERT
    ON EMPLOYEES
    FOR EACH ROW
BEGIN

```

```

    IF(NEW.doj < '2018-01-01') THEN
        SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'EMPLOYEE BE
LOW 2018 DOJ CANNOT BE INSERTED';
    END IF;
END //
insert into employees
values (104, 'raj', 'rajx@gmail.com', '2', 'field officer', '2017-
03-10', '9680988342', '34000')//
DELIMITER ;

```

```

MySQL localhost:3306 ssl companydb SQL > DELIMITER //
MySQL localhost:3306 ssl companydb SQL > CREATE TRIGGER NOT_BELOW_2018
-> BEFORE INSERT
-> ON EMPLOYEES
-> FOR EACH ROW
-> BEGIN
-> IF(NEW.doj < '2018-01-01') THEN
->     SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'EMPLOYEE BELOW 2018 DOJ CANNOT BE INSERTED'
;
-> END IF;
-> END //
Query OK, 0 rows affected (0.0110 sec)
MySQL localhost:3306 ssl companydb SQL > insert into employees
-> values (104, 'raj', 'rajx@gmail.com', '2', 'field officer', '2017-03-10', '9680988342', '34000')//
ERROR: 1644 (45000): EMPLOYEE BELOW 2018 DOJ CANNOT BE INSERTED
MySQL localhost:3306 ssl companydb SQL > DELIMITER ;

```

-
- e) To Prevent any Employee Named Tom to be inserted into the table

```

DELIMITER //
create trigger prevent_name before insert on employees
for each row
begin
    if new.emp_name = "tom" then signal sqlstate '45000' set message_
text= "tom is prohibited from insertion !!!";
    end if;
end//
DELIMITER ;

insert into employees values
('107', 'tom', 'tommy@gmail.com', '6', 'assistant doctor', '2020-
02-22', '7373558197', '25000');

select * from employees;

```

```

MySQL localhost:3306 ssl companydb SQL > DELIMITER //
MySQL localhost:3306 ssl companydb SQL > create trigger prevent_name before insert on employees
--> for each row
--> begin
--> if new.emp_name = "tom" then signal sqlstate '45000' set message_text= "tom is prohibited from in
section !!!";
--> end if;
--> end//
Query OK, 0 rows affected (0.0111 sec)
MySQL localhost:3306 ssl companydb SQL > DELIMITER ;
MySQL localhost:3306 ssl companydb SQL >
MySQL localhost:3306 ssl companydb SQL > insert into employees values
--> ('107', 'tom', 'tommy@gmail.com', '6', 'assistant doctor', '2020-02-22', '7373558197', '25000');
ERROR: 1644 (45000): tom is prohibited from insertion !!!
MySQL localhost:3306 ssl companydb SQL >
MySQL localhost:3306 ssl companydb SQL > select * from employees;
+-----+-----+-----+-----+-----+-----+-----+-----+
| emp_id | emp_name | email | dept_id | designation | doj | phone_num | emp_salary |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 100 | timber | timber@gmail.com | 1 | salesman | 2019-01-12 | 9842042345 | 100000.000 |
| 101 | jaanu | jaanu@gmail.com | 2 | senior manager | 2020-01-01 | 9865161122 | 15000.000 |
| 102 | jen | jen@gmail.com | 3 | junior manager | 2019-10-12 | 9042842344 | 60000.000 |
| 103 | jack | jack@gmail.com | 2 | assistant | 2016-03-01 | 9498853055 | 50000.000 |
+-----+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.0004 sec)

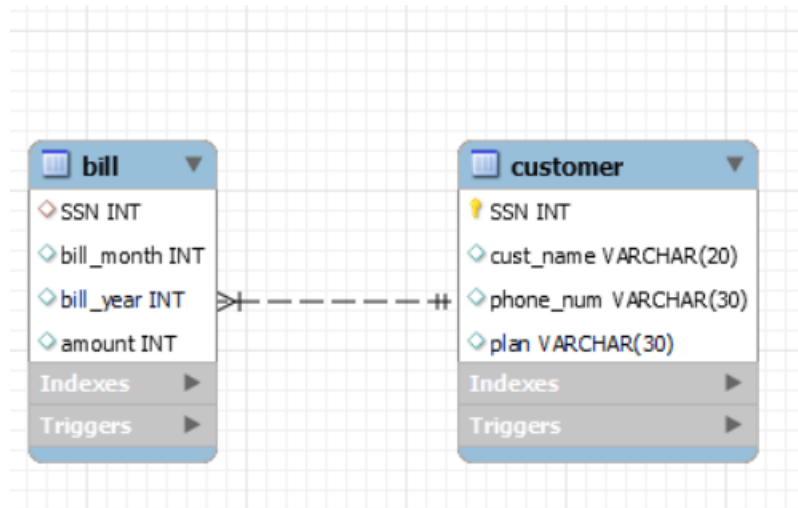
```


PROBLEM 2

2. Consider the following relational schema that manages the telephone bills of a mobile phone company.

CUSTOMER (SSN, Name, PhoneNum, Plan)

BILL (SSN, Month, Year, amount)



```
CREATE DATABASE customerDB;  
USE customerDB;
```

```
create table customer  
(  
    ssn int primary key,  
    name varchar(30),  
    phone_no varchar(30),  
    plan varchar(30)  
);
```

```
create table bill  
(  
    ssn int not null,  
    foreign key (ssn) references customer(ssn),  
    month int not null,  
    year int not null,  
    amount int not null  
);
```

```

create table calls
(
    from_ssn int,
    to_ssn int,
    month int,
    year int,
    amount int,
    foreign key(from_ssn) references customer(ssn),
    foreign key(to_ssn) references customer(ssn)
);

```

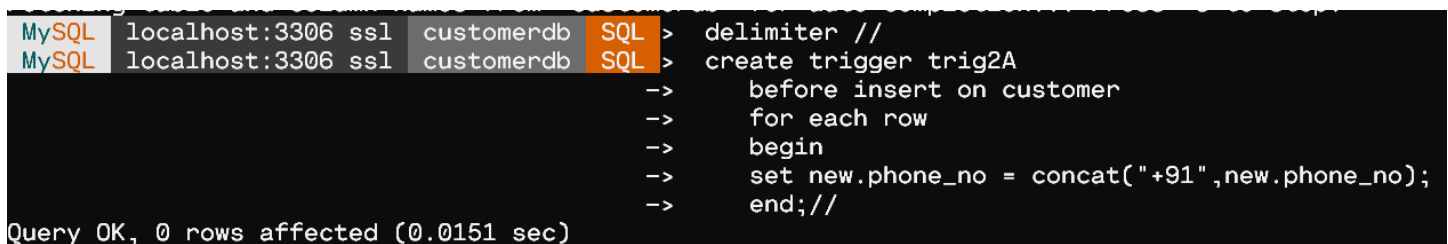
-

- a. That Adds +91 to all phone numbers as a prefix in the Customer Table.

```

delimiter //
create trigger trig2A
    before insert on customer
    for each row
    begin
    set new.phone_no = concat("+91",new.phone_no);
    end; //

```



```

MySQL localhost:3306 ssl customerdb SQL > delimiter //
MySQL localhost:3306 ssl customerdb SQL > create trigger trig2A
-> before insert on customer
-> for each row
-> begin
-> set new.phone_no = concat("+91",new.phone_no);
-> end; //
Query OK, 0 rows affected (0.0151 sec)

```

-

- b. Write a trigger that after each phone call updates the customer's bill.

```

delimiter //
create trigger trig2B after insert on calls
    for each row
    update bill
    set bill.amount = bill.amount + new.amount

```

```

where bill.year = new.year
and bill.month = new.month
and bill.ssn = new.from_ssn;

```

```

MySQL localhost:3306 ssl customerdb SQL > delimiter //
MySQL localhost:3306 ssl customerdb SQL > create trigger trig2B after insert on calls
->     for each row
->     update bill
->     set bill.amount = bill.amount + new.amount
->     where bill.year = new.year
->     and bill.month = new.month
->     and bill.ssn = new.from_ssn;
Query OK, 0 rows affected (0.0150 sec)

```

-

- c. Write a trigger to not insert bill older than 2020 (year > 2020)

.

```

delimiter //
create trigger trig2C before insert on bill
for each row
begin
if new.year < 2020 then signal sqlstate '45003' set message_text
= "Bill year must be >= 2020 !!!";
end if;
end;

```

```

MySQL localhost:3306 ssl customerdb SQL > delimiter //
MySQL localhost:3306 ssl customerdb SQL > create trigger trig2C before insert on bill
->     for each row
->     begin
->     if new.year < 2020 then signal sqlstate '45003' set message_text = "Bill year must be >= 2020 !!
!";
->     end if;
->     end;
Query OK, 0 rows affected (0.0113 sec)

```