1)

Code:

DROP DATABASE IF EXISTS w6q1\_empire;

CREATE DATABASE w6q1\_empire;

USE w6q1\_empire;

CREATE TABLE Department (

dept\_no INT PRIMARY KEY,

dept\_Name VARCHAR(20) NOT NULL,

dept\_location VARCHAR(20) NOT NULL

);

INSERT INTO Department VALUES

(170, 'Technology', 'Massachusetts'),

(291, 'Entertainment', 'California'),

(752, 'Agriculture', 'Texas'),

(817, 'Sales', 'New Jersey'),

(623, 'Medical', 'Missouri'),

(868, 'Tourism', 'Florida');

CREATE TABLE Employee (

emp\_id INT PRIMARY KEY,

emp\_name VARCHAR(25) NOT NULL,

email VARCHAR(25) NOT NULL,

dept\_id INT NOT NULL,

designation VARCHAR(13) NOT NULL,

doj DATE NOT NULL,

phone\_number VARCHAR(12) NOT NULL,

emp\_salary FLOAT(7,2) NOT NULL DEFAULT 0.0

);

INSERT INTO Employee

VALUES

(100, "Steven King" , "SKING" , 752, 'AD\_PRES' , '2003-06-17', '515.123.4567', 24000.00),

(101, "Neena Kochhar" , "NKOCCHAR", 170, 'AD\_VP' , '2005-09-21', '515.123.4568', 17000.00),

(102, "Lex De Haan" , "LDEHAAN" , 752, 'AD\_VP' , '2001-01-13', '515.123.4569', 17000.00),

(103, "Alexander Hunold", "AHUNOLD" , 291, 'IT\_PROG' , '2006-01-03', '590.423.4567', 9000.00),

(104, "Bruce Ernst" , "BERNST" , 291, 'IT\_PROG' , '2007-05-21', '590.423.4568', 6000.00),

(105, "David Austin" , "DAUSTIN" , 170, 'IT\_PROG' , '2005-06-25', '590.423.4569', 4800.00),

(106, "Valli Pataballa" , "VPATABAL", 868, 'IT\_PROG' , '2006-02-05', '590.423.4560', 4800.00),

(107, "Diana Lorentz" , "DLORENTZ", 868, 'IT\_PROG' , '2007-02-07', '590.423.5567', 4800.00),

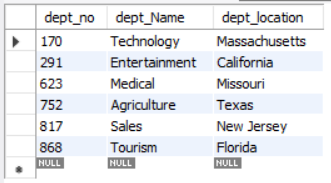
(108, "Nancy Greenberg" , "NGREENBE", 817, 'FI\_MGR' , '2002-08-17', '515.124.4569', 12008.00),

(109, "Daniel Faviet" , "DFAVIET" , 817, 'FI\_ACCOUNT', '2002-08-16', '515.124.4169', 9000.00),

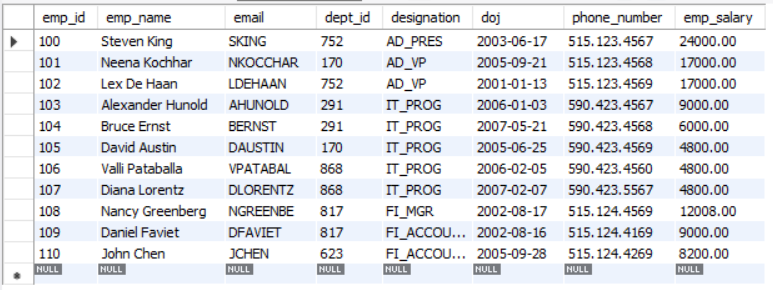
(110, "John Chen" , "JCHEN" , 623, 'FI\_ACCOUNT', '2005-09-28', '515.124.4269', 8200.00);

Inputs:

* Departments



* Employees



Queries & Outputs:

-- 1a) Calculating the number of rows inserted so far.

CREATE TABLE rows\_so\_far (

num\_rows INT

);

INSERT INTO rows\_so\_far VALUES (0);

DELIMITER //

CREATE TRIGGER num\_rows\_inserted\_so\_far\_dept

AFTER INSERT ON Department FOR EACH ROW

BEGIN

UPDATE rows\_so\_far

SET num\_rows = num\_rows + 1;

END; //

DELIMITER ;

DELIMITER //

CREATE TRIGGER num\_rows\_inserted\_so\_far\_emp

AFTER INSERT ON Employee FOR EACH ROW

BEGIN

UPDATE rows\_so\_far

SET num\_rows = num\_rows + 1;

END; //

DELIMITER ;

SELECT \* FROM rows\_so\_far;



/\* \*\/

DROP TABLE IF EXISTS message;

CREATE TABLE message (

msg VARCHAR(35)

);

DELIMITER //

CREATE TRIGGER disp\_msg\_before\_ins\_emp

BEFORE INSERT ON Employee FOR EACH ROW

BEGIN

INSERT INTO message VALUES(CONCAT('Hi ', NEW.emp\_name));

END; //

DELIMITER ;

INSERT INTO Employee VALUES

(1000, 'Jerry', 'JERRY', 623, 'Mouse', '2019-01-01', '10000000000', 10000);

SELECT \* FROM message;

/\*\*/



-- 1c) Ensuring that the Employee Table does not contain duplicates or null value

-- in the Name column

DELIMITER //

CREATE TRIGGER check\_name

BEFORE INSERT ON Employee FOR EACH ROW

BEGIN

DECLARE newname\_count INT;

IF NEW.emp\_name IS NULL THEN

SIGNAL SQLSTATE '50002' SET MESSAGE\_TEXT = 'Name is Null';

END IF;

SELECT COUNT(emp\_id) INTO newname\_count FROM Employee WHERE emp\_name = NEW.emp\_name;

IF newname\_count > 0 THEN

SIGNAL SQLSTATE '50003' SET MESSAGE\_TEXT = 'Name already exists';

END IF;

END; //

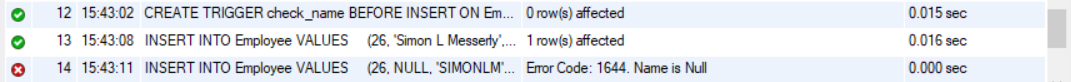
DELIMITER ;

INSERT INTO Employee VALUES

(26, 'Simon L Messerly', 'SIMONLM', 291, 'Producer', '2015-11-09', '8475295864', 41000);

INSERT INTO Employee VALUES

(26, NULL, 'SIMONLM', 291, 'Producer', '2015-11-09', '8475295864', 41000);



-- 1d) To insert Employee details into Employee Table only of DOJ > 2018

DELIMITER //

CREATE TRIGGER check\_doj

BEFORE INSERT ON Employee FOR EACH ROW

BEGIN

IF NEW.DOJ <= '2018-01-01' THEN

SIGNAL SQLSTATE '50004' SET MESSAGE\_TEXT = 'Employee DOJ is before 2018';

END IF;

END; //

DELIMITER ;

INSERT INTO Employee VALUES

(420, 'Sniper Snape', 'SWIPER', 623, 'Pro Thief', '2017-12-31', '1674531568', 500);



-- 1e) To Prevent any Employee Named Tom to be inserted into the table

DELIMITER //

CREATE TRIGGER check\_name\_tom

BEFORE INSERT ON Employee FOR EACH ROW

BEGIN

IF NEW.emp\_name = 'Tom' THEN

SIGNAL SQLSTATE '50005' SET MESSAGE\_TEXT = 'Employee Tom is restricted';

END IF;

END; //

DELIMITER ;

INSERT INTO Employee VALUES

(27, 'Tom', 'TOMH', 291, 'Lead Actor', '2020-03-15', '7535793458', 1500000);



2)

Code:

DROP DATABASE IF EXISTS w6q2\_bills;

CREATE DATABASE w6q2\_bills;

USE w6q2\_bills;

CREATE TABLE customer (

ssn INT PRIMARY KEY,

sname VARCHAR(20) NOT NULL,

phone\_num VARCHAR(20) NOT NULL,

plan INT NOT NULL

);

INSERT INTO customer VALUES

(1, 'Lelouch', '9014959259', 100),

(2, 'Suzaku' , '9390499789', 500),

(3, 'CC' , '9398027105', 800);

CREATE TABLE bill (

ssn INT,

b\_month INT,

b\_year INT,

amount FLOAT,

FOREIGN KEY (ssn) REFERENCES customer(ssn)

);

INSERT INTO bill VALUES

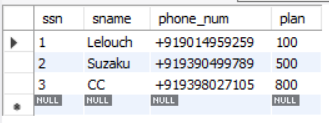
(1, 4, 2014, 10000),

(2, 9, 2015, 6000),

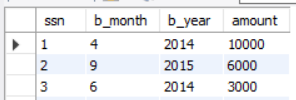
(3, 6, 2014, 3000);

Inputs:

* Customer (trigger in action for phone\_num)



* Bill



Outputs & Queries:

-- 2a) Adds +91 to all phone numbers as a prefix in the Customer Table

DELIMITER $$

CREATE TRIGGER trig

BEFORE INSERT ON customer

FOR EACH ROW

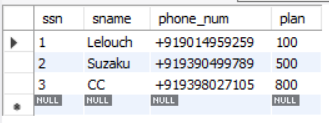
BEGIN

SET NEW.phone\_num = CONCAT("+91", NEW.phone\_num);

END $$

DELIMITER ;

SELECT \* FROM customer;



-- 2c) not insert bill older than 2020 (year > 2020)

DROP TRIGGER IF EXISTS phno\_bill;

DELIMITER $$

CREATE TRIGGER phno\_bill

BEFORE INSERT ON bill

FOR EACH ROW

BEGIN

IF NEW.b\_year > 2020 THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = "error";

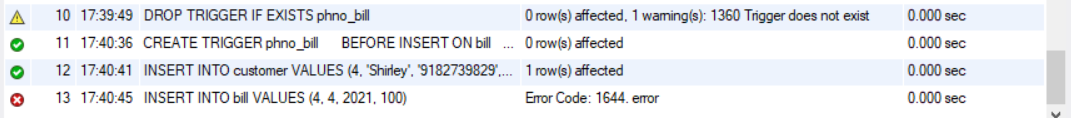
END IF;

END $$

DELIMITER ;

INSERT INTO customer VALUES (4, 'Shirley', '9182739829', 150);

INSERT INTO bill VALUES (4, 4, 2021, 100);



/\*\*/