1)

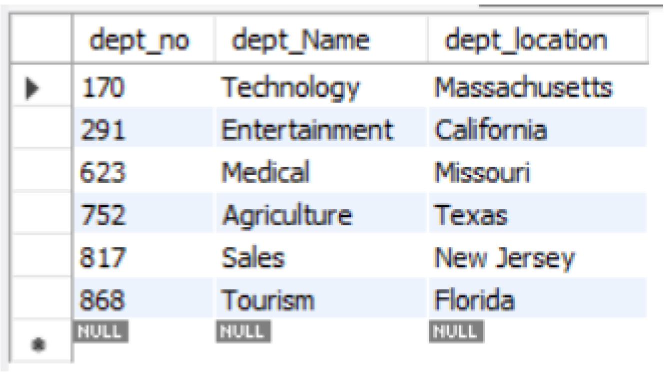
Code:

```
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);
```

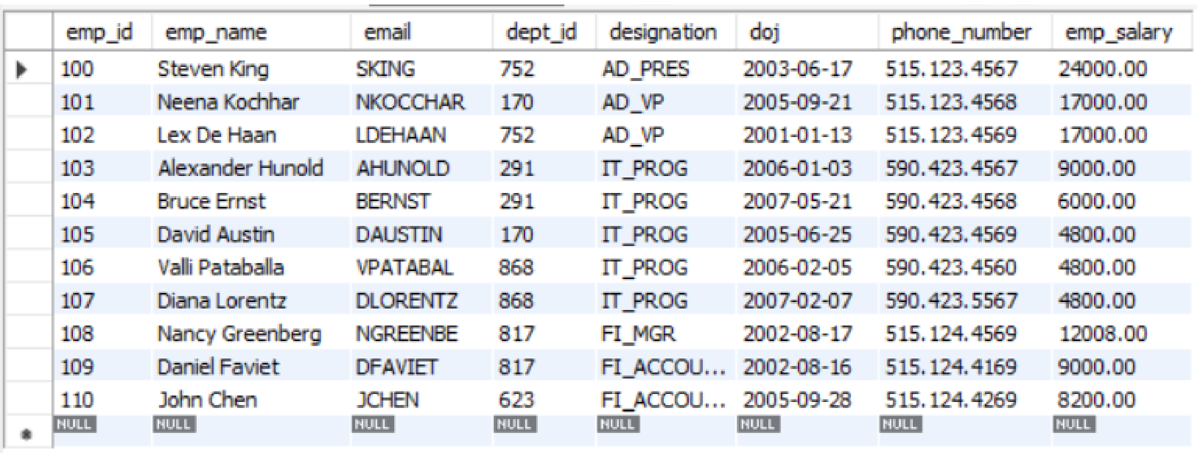
Inputs:

- ➔ Departments



dept_no	dept_Name	dept_location
170	Technology	Massachusetts
291	Entertainment	California
623	Medical	Missouri
752	Agriculture	Texas
817	Sales	New Jersey
868	Tourism	Florida
NULL	NULL	NULL

- ➔ Employees



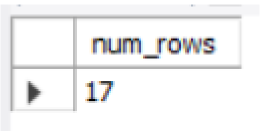
	emp_id	emp_name	email	dept_id	designation	doj	phone_number	emp_salary
▶	100	Steven King	SKING	752	AD_PRES	2003-06-17	515.123.4567	24000.00
	101	Neena Kochhar	NKOCCHAR	170	AD_VP	2005-09-21	515.123.4568	17000.00
	102	Lex De Haan	LDEHAAN	752	AD_VP	2001-01-13	515.123.4569	17000.00
	103	Alexander Hunold	AHUNOLD	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_ACCOU...	2002-08-16	515.124.4169	9000.00
	110	John Chen	JCHEN	623	FI_ACCOU...	2005-09-28	515.124.4269	8200.00
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

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
```

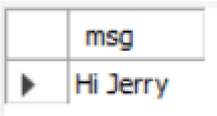
```
        SET num_rows = num_rows + 1;
    END; //
DELIMITER ;

SELECT * FROM rows_so_far;
```



	num_rows
▶	17

➔

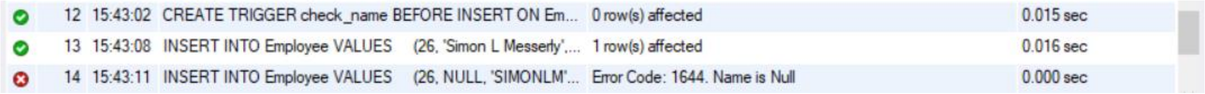
```
/* *\/
DROP TABLE IF EXISTS message;
CREATE TABLE message (
    msg VARCHAR(35)
);

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;
/**/
```



	msg
▶	Hi Jerry

➔

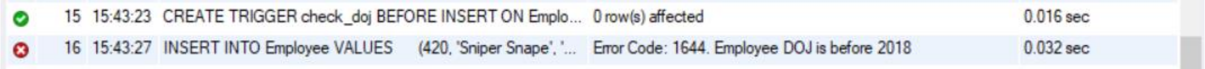
```
-- 1c) Ensuring that the Employee Table does not contain duplicates or null value
-- in the Name column
DELIMITER //
CREATE TRIGGER check_name
BEFORE INSERT ON Employee FOR EACH ROW
  BEGIN
    DECLARE newname_count INT;

    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);
```



```
✓  12  15:43:02  CREATE TRIGGER check_name BEFORE INSERT ON Em...  0 row(s) affected                 0.015 sec
✓  13  15:43:08  INSERT INTO Employee VALUES    (26, 'Simon L Messerly',...  1 row(s) affected      0.016 sec
✗  14  15:43:11  INSERT INTO Employee VALUES    (26, NULL, 'SIMONLM'...  Error Code: 1644. Name is Null   0.000 sec
```

➔

```
-- 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);
```

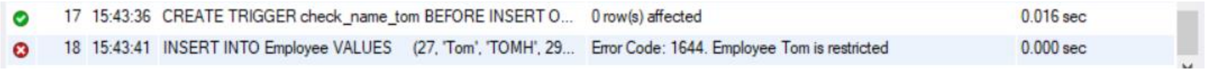


```
✓  15  15:43:23  CREATE TRIGGER check_doj BEFORE INSERT ON Emplo...  0 row(s) affected                              0.016 sec
✗  16  15:43:27  INSERT INTO Employee VALUES    (420, 'Sniper Snape', '...  Error Code: 1644. Employee DOJ is before 2018   0.032 sec
```

➔

```
-- 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);
```



```
17  15:43:36  CREATE TRIGGER check_name_tom BEFORE INSERT O...   0 row(s) affected                               0.016 sec
18  15:43:41  INSERT INTO Employee VALUES    (27, 'Tom', 'TOMH', 29...   Error Code: 1644. Employee Tom is restricted   0.000 sec
```

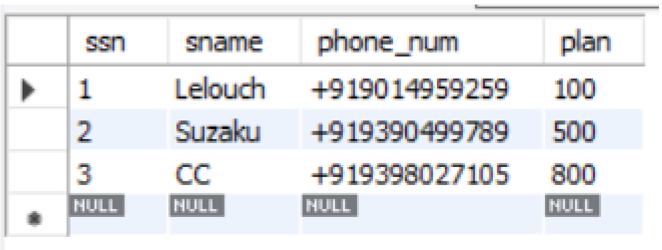
2)

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);
```

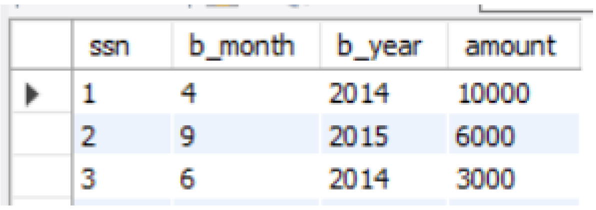
Inputs:

➔ Customer (trigger in action for phone_num)



	ssn	sname	phone_num	plan
▶	1	Lelouch	+919014959259	100
	2	Suzaku	+919390499789	500
	3	CC	+919398027105	800
*	NULL	NULL	NULL	NULL

➔ Bill

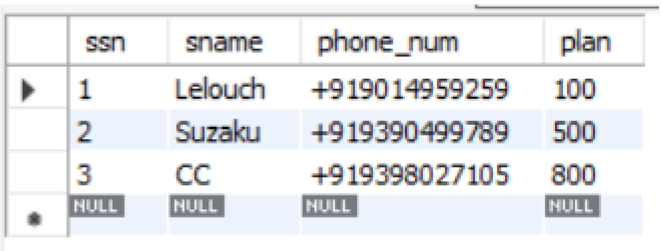


	ssn	b_month	b_year	amount
▶	1	4	2014	10000
	2	9	2015	6000
	3	6	2014	3000

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;
```



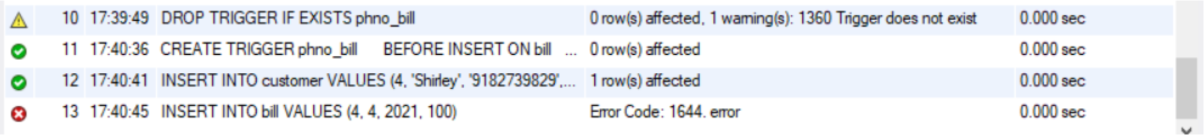
ssn	sname	phone_num	plan
1	Lelouch	+919014959259	100
2	Suzaku	+919390499789	500
3	CC	+919398027105	800
NULL	NULL	NULL	NULL

➔

```
-- 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);



```
⚠  10  17:39:49  DROP TRIGGER IF EXISTS phno_bill                 0 row(s) affected, 1 warning(s): 1360 Trigger does not exist   0.000 sec
✓  11  17:40:36  CREATE TRIGGER phno_bill      BEFORE INSERT ON bill  ...   0 row(s) affected                                    0.000 sec
✓  12  17:40:41  INSERT INTO customer VALUES (4, 'Shirley', '9182739829',...   1 row(s) affected                                0.000 sec
✗  13  17:40:45  INSERT INTO bill VALUES (4, 4, 2021, 100)         Error Code: 1644. error                                       0.000 sec
```

/**/