Week 9

| Vanshika Goel | PES1UG20CS484 | Section H | Roll No 40 |

1. Write a trigger on insert to compartment table when a new compartment gets added to a train and make sure that total number of compartments available in the train does not exceed more than 4.

```
[mysql> DELIMITER $$
mysql> CREATE TRIGGER after_insert_compartment
    -> AFTER INSERT
[   -> ON COMPARTMENT_484 FOR EACH ROW
[   -> BEGIN
    ->   DECLARE result VARCHAR(255);
    ->   DECLARE compartment_count INT;
    ->   SET result='Train can have only 4 compartments';
[   -> SELECT COUNT(Train_no) INTO compartment_count FROM COMPARTMENT_484;
[   -> IF compartment_count>4 THEN SIGNAL SQLSTATE '45000'
[   -> SET MESSAGE_TEXT=result;
[   -> END IF;
[   -> END $$
[Query OK, 0 rows affected (0.01 sec)
[
[mysql> DELIMITER ;
[mysql> SELECT * FROM COMPARTMENT_484 WHERE Train_no='58450';
+--------------------+------------------+---------------------+-----------------------+----------+
| Compartment_number | Compartment_type | Compartment_capacity | Compartment_availibility | Train_no |
+--------------------+------------------+---------------------+-----------------------+----------+
| F01                | I Class          |                  16 |                     4 | 58450    |
| G01                | III Class        |                  60 |                     2 | 58450    |
| S01                | II Class         |                  30 |                     5 | 58450    |
| S02                | II Class         |                  30 |                     3 | 58450    |
+--------------------+------------------+---------------------+-----------------------+----------+
4 rows in set (0.01 sec)

[mysql> INSERT INTO COMPARTMENT_484 VALUES('F02', 'I Class', 16, 0, 58450);
ERROR 1644 (45000): Train can have only 4 compartments
mysql>
```

2. Create a trigger to add payment information to the backup table when we try to delete some information from the ticket table.

```
mysql> DELIMITER $$
mysql> CREATE TRIGGER BeforeDelete
    -> BEFORE DELETE
    -> ON TICKET_484 FOR EACH ROW
    -> BEGIN
    ->    INSERT INTO BACKUP_TABLE_PAYMENT_484
    ->    SELECT * FROM PAYMENT_INFO_484 WHERE Pnr=old.Pnr;
    ->    DELETE FROM PAYMENT_INFO_484 WHERE Pnr = old.Pnr;
[   -> END $$
Query OK, 0 rows affected (0.01 sec)

[mysql> DELIMITER ;
mysql>

[mysql> DELETE FROM TICKET_484 WHERE Pnr='PNR001';
Query OK, 1 row affected (0.01 sec)

[mysql> SELECT * FROM BACKUP_TABLE_PAYMENT_484;
+----------------+------+--------------------+-------+--------+
| Transaction_id | Bank | Card_number        | Price | Pnr    |
+----------------+------+--------------------+-------+--------+
|         342344 | IOB  | 5623570000000000   |  NULL | PNR001 |
+----------------+------+--------------------+-------+--------+
1 row in set (0.00 sec)

[mysql> SELECT * FROM TICKET_484 WHERE Pnr='PNR001';
Empty set (0.00 sec)

[mysql> SELECT * FROM PAYMENT_INFO_484 WHERE Pnr='PNR001';
Empty set (0.00 sec)

mysql>
```