

DBMS

LAB – 9

NAME : SANMAT SANJAYAKUMAR PAYAGOUDAR

SRN : PES1UG20CS385

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.

Query :

```
DELIMITER $$
CREATE TRIGGER comp_ins
BEFORE INSERT
ON compartment_385
FOR EACH ROW BEGIN
    DECLARE error_msg VARCHAR(300);
    DECLARE cnt_comp INT;

    SET error_msg = ("Total number of compartments available in the train
shouldn't exceed 4");

    SELECT COUNT(compartment_385.Compartment_No)
    INTO cnt_comp FROM compartment_385
    WHERE NEW.Train_No = compartment_385.Train_No
    AND compartment_385.Availability > 0;

    IF cnt_comp > 3 THEN
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = error_msg;
    END IF;
END $$
```

OUTPUT :

```
MariaDB [railway_reservation]> DELIMITER $$
MariaDB [railway_reservation]> CREATE TRIGGER comp_ins
-> BEFORE INSERT
-> ON compartment_385
-> FOR EACH ROW BEGIN
->     DECLARE error_msg VARCHAR(300);
->     DECLARE cnt_comp INT;
->
->     SET error_msg = ("Total number of compartments available in the train shouldn't exceed 4");
->
->     SELECT COUNT(compartment_385.Compartment_No)
->     INTO cnt_comp FROM compartment_385
->     WHERE NEW.Train_No = compartment_385.Train_No
->     AND compartment_385.Availability > 0;
->
->     IF cnt_comp > 3 THEN
->         SIGNAL SQLSTATE '45000'
->         SET MESSAGE_TEXT = error_msg;
->     END IF;
-> END $$
Query OK, 0 rows affected (0.009 sec)

MariaDB [railway_reservation]> DELIMITER ;
MariaDB [railway_reservation]> INSERT INTO compartment_385 VALUES('S03', 'III Class', 30, 6, 62621);
ERROR 1644 (45000): Total number of compartments available in the train shouldn't exceed 4
MariaDB [railway_reservation]> INSERT INTO compartment_385 VALUES('S03', 'III Class', 30, 6, 25260);
Query OK, 1 row affected (0.024 sec)
```

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

Query :

```
CREATE TABLE payment_info_backup_385(Transaction_ID varchar(20) NOT NULL,  
    Bank varchar(50),  
    Card_Number bigint(20),  
    Price decimal(10,2),  
    PNR varchar(10),  
    PRIMARY KEY (Transaction_ID)  
);  
  
DELIMITER $$  
CREATE TRIGGER backup_pay_info_385  
BEFORE DELETE  
ON TICKET_385  
FOR EACH ROW  
BEGIN  
    INSERT INTO payment_info_backup_385 SELECT * FROM payment_info_385 WHERE  
PNR = old.PNR;  
END $$  
  
DELIMITER ;  
DELETE FROM TICKET_385 WHERE PNR = 'PNR006';  
select * from payment_info_backup_385;
```

OUTPUT :

```
MariaDB [railway_reservation]> CREATE TABLE payment_info_backup_385(Transaction_ID varchar(20) NOT NULL,  
-> Bank varchar(50),  
-> Card_Number bigint(20),  
-> Price decimal(10,2),  
-> PNR varchar(10),  
-> PRIMARY KEY (Transaction_ID)  
-> );  
Query OK, 0 rows affected (0.007 sec)  
  
MariaDB [railway_reservation]> DELIMITER $$  
MariaDB [railway_reservation]> CREATE TRIGGER backup_pay_info_385  
-> BEFORE DELETE  
-> ON TICKET_385  
-> FOR EACH ROW  
-> BEGIN  
-> INSERT INTO payment_info_backup_385 SELECT * FROM payment_info_385 WHERE PNR = old.PNR;  
-> END $$  
Query OK, 0 rows affected (0.005 sec)  
  
MariaDB [railway_reservation]>  
MariaDB [railway_reservation]> DELIMITER ;  
MariaDB [railway_reservation]> DELETE FROM TICKET_385 WHERE PNR = 'PNR006';  
Query OK, 1 row affected (0.006 sec)  
  
MariaDB [railway_reservation]> select * from payment_info_backup_385;  
+-----+-----+-----+-----+-----+  
| Transaction_ID | Bank | Card_Number | Price | PNR |  
+-----+-----+-----+-----+-----+  
| 146445 | HDFC | 1234330000000000 | 806.40 | PNR006 |  
+-----+-----+-----+-----+-----+  
1 row in set (0.000 sec)
```