

DBMS Lab-9

Name: Vrushank G

SRN: PES1UG20CS516

Section: I

Lab 8 – Creating Triggers

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.

```
DELIMITER $$
CREATE TRIGGER compartment_no_check BEFORE INSERT on compartment for each row
BEGIN
IF (SELECT count(compartment_no) FROM compartment WHERE
train_number=NEW.train_number)>=4 THEN
SIGNAL SQLSTATE '45000' SET message_text="Cannot add more than 4 compartments
to a train";
END IF;
END $$
DELIMITER ;
```

```
MariaDB [peslug20cs516]> show triggers ;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Trigger | Event | Table | Statement | sql_mode | Definer | character_set_client | collation_connection | Database Collat1 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| compartment_no_check | INSERT | compartment | BEGIN
IF (SELECT count(compartment_no) FROM compartment WHERE train_number=NEW.train_number)>=4 THEN
SIGNAL SQLSTATE '45000' SET message_text="Cannot add more than 4 compartments to a train";
END IF;
END | BEFORE | 2022-11-04 11:38:34.79 | NO_ZERO_IN_DATE,NO_ZERO_DATE,NO_ENGINE_SUBSTITUTION | root@localhost | utf8mb4 | utf8mb4_general_ci | utf8mb4_general_ci |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.008 sec)
```

```
MariaDB [peslug20cs516]> select count(compartment_no),train_number from compartment group by train_number;
+-----+-----+
| count(compartment_no) | train_number |
+-----+-----+
| 3 | 25260 |
| 4 | 25261 |
| 4 | 58450 |
| 2 | 58451 |
| 2 | 62620 |
| 5 | 62621 |
+-----+-----+
6 rows in set (0.001 sec)
```

```
MariaDB [peslug20cs516]> insert into compartment values("F02","I Class",16,6,25261);
ERROR 1644 (45000): Cannot add more than 4 compartments to a train
MariaDB [peslug20cs516]>
```

DBMS Lab-9

Name: Vrushank G

SRN: PES1UG20CS516

Section: I

```
MariaDB [peslug20cs516]> select count(compartment_no),train_number from compartment group by train_number;
+-----+-----+
| count(compartment_no) | train_number |
+-----+-----+
| 3                     | 25260        |
| 4                     | 25261        |
| 4                     | 58450        |
| 2                     | 58451        |
| 2                     | 62620        |
| 5                     | 62621        |
+-----+-----+
6 rows in set (0.000 sec)

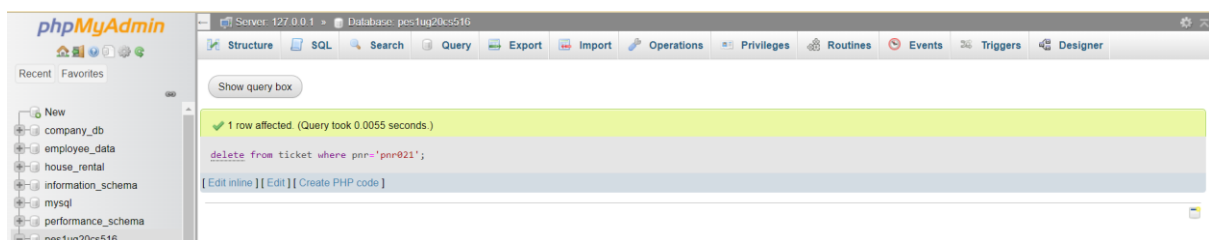
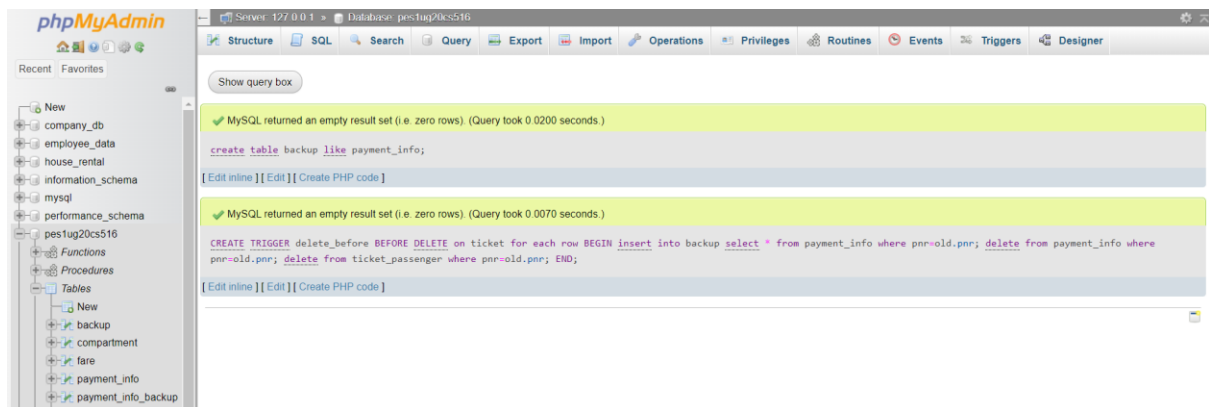
MariaDB [peslug20cs516]> insert into compartment values("F02","II Class",20,5,25260);
Query OK, 1 row affected (0.012 sec)

MariaDB [peslug20cs516]> insert into compartment values("F02","III Class",20,5,25260);
ERROR 1644 (45000): Cannot add more than 4 compartments to a train
```

2. Create a new table called 'backup' in the railway reservation database with all the attributes of ticket, ticket passenger and paymentinfo.

You are supposed to initiate a 'deletetrigger' whenever a ticket record is deleted from ticket table.

This deletetrigger has to Fetch the data to be deleted from all the three tables(ticket, ticket passenger and paymentinfo) with the given ticket info and add the same into backup table. After that, deletion of ticket record should be allowed.



DBMS Lab-9

Name: Vrushank G

SRN: PES1UG20CS516

Section: I

The screenshot displays the phpMyAdmin web interface. On the left, a sidebar shows a database structure with various tables like 'company_db', 'employee_data', 'house_rental', 'information_schema', 'mysql', 'performance_schema', 'pes1ug20cs516', 'Functions', 'Procedures', and 'Tables'. The 'Tables' section is expanded, showing a list of tables including 'backup', 'compartment', 'fare', 'payment_info', 'payment_info_backup', 'route_info', 'ticket', and 'ticket_passenger'. The main panel shows the 'backup' table selected. At the top, a status bar indicates 'Showing rows 0 - 0 (1 total, Query took 0.0005 seconds)'. Below this, the SQL query 'SELECT * FROM `backup`' is displayed. A toolbar offers options like 'Profiling', 'Edit inline', 'Edit', 'Explain SQL', 'Create PHP code', and 'Refresh'. A 'Show all' button and a 'Number of rows' dropdown (set to 25) are present. The table data is shown with columns: 'transaction_no', 'bank', 'card_no', 'price', and 'pnr'. A single row of data is visible: '146449', 'SBI', '123456789', '900.00', and 'pnr021'. Below the table, there are buttons for 'Edit', 'Copy', and 'Delete'. A 'Query results operations' section at the bottom includes links for 'Print', 'Copy to clipboard', 'Export', 'Display chart', and 'Create view'.

transaction_no	bank	card_no	price	pnr
146449	SBI	123456789	900.00	pnr021