Data-Base Management System

LAB - 04

NAME – KEERTHAN P V

SRN – PES2UG23CS272

TASK1

1. **Insert a new event called "AI Hackathon" conducted by team 'T4' in 'Seminar Hall', floor '2', room 205, priced at 900.00.**

Sol:

Before:

```
mysql> select * from event;
+----------+------------------------+---------------+-------+---------+--------+---------+
| event_id | event_name             | building      | floor | room_no | price  | team_id |
+----------+------------------------+---------------+-------+---------+--------+---------+
| E1       | Adventure Trek         | Outdoors      | 1     |     101 | 200.00 | T3      |
| E10      | Music Concert          | Amphi%he@ter  | 2     |     202 | 120.00 | T6      |
| E11      | Art Workshop           | M@in Building | 1     |     101 | 150.00 | T6      |
| E12      | Film Screening         | Theater       | 2     |     203 | 500.00 | T6      |
| E13      | Traditional Dance wars | @udi%orium    | 1     |     102 | 850.00 | T8      |
| E14      | Singing Contest        | M@in Building | 2     |     203 | 500.00 | T8      |
| E15      | Fashion Exhibition     | Sports Comp%ex | 1     |     103 | 120.00 | T8      |
| E16      | Poetry Slam            | Amphi%he@ter  | 2     |     201 | 100.00 | T8      |
| E17      | Quiz Competition       | @udi%orium    | 1     |     102 | 100.00 | T9      |
| E18      | Tech Talk              | Seminar Hall  | 1     |     101 | 200.00 | T9      |
| E19      | Web Design Contest     | @Computer%Lab% | 2     |     202 | 890.00 | T9      |
| E2       | Paintball Tournament   | Sports Comp%ex | 2     |     201 | 150.00 | T3      |
| E20      | Hackathon              | M@in Building | 1     |     103 | 650.00 | T9      |
| E21      | Stage Setup            | M@in Building | 2     |     203 | 900.00 | T12     |
| E22      | Sound and Lighting     | @udi%orium    | 1     |     102 | 250.00 | T12     |
| E23      | Event Coordination     | Sports Comp%ex | 1     |     103 | 500.00 | T12     |
| E24      | Volunteer Management   | M@in Building | 3     |     303 | 600.00 | T12     |
| E25      | Baking Competition     | Kitchen       | 1     |     101 | 850.00 | T14     |
| E26      | Food Tasting           | Cafeteria     | 2     |     202 | 500.00 | T14     |
| E27      | Cooking Workshop       | M@in Building | 1     |     102 | 120.00 | T14     |
| E29      | Short Film Screening   | Theater       | 1     |     103 | 400.00 | T15     |
| E3       | Escape Room Challenge  | M@in Building | 1     |     102 | 120.00 | T3      |
| E30      | Literary Debate        | Seminar Hall  | 1     |     101 | 550.00 | T15     |
| E31      | Writing Competition    | Library       | 2     |     202 | 950.00 | T15     |
| E32      | Poetry Recitation      | Amphi%he@ter  | 1     |     102 | 100.00 | T15     |
| E4       | Photography Contest    | @udi%orium    | 2     |     202 | 850.00 | T4      |
| E5       | Code Jam               | @Computer%Lab% | 1     |     101 | 100.00 | T4      |
| E6       | Robot Wars             | Robotics Arena | 1     |     201 | 100.00 | T4      |
| E7       | Tech Expo              | M@in Building | 2     |     203 | 550.00 | T4      |
| E8       | Gaming Tournament      | Sports Comp%ex | 1     |     103 | 800.00 | T4      |
| E9       | Drama Play             | @udi%orium    | 1     |     102 | 100.00 | T6      |
+----------+------------------------+---------------+-------+---------+--------+---------+
31 rows in set (0.00 sec)
```

Command:

```
mysql> INSERT INTO event (event_id,event_name, building, floor, room_no,price,team_id) VALUES('E33', 'AI HACATHON', 'Seminar Hall',2,205,900,'T4');
Query OK, 1 row affected (0.01 sec)
```

after:

```
mysql> select * from event;
+----------+------------------------+----------------+-------+---------+--------+---------+
| event_id | event_name             | building       | floor | room_no | price  | team_id |
+----------+------------------------+----------------+-------+---------+--------+---------+
| E1       | Adventure Trek         | Outdoors       | 1     |     101 | 200.00 | T3      |
| E10      | Music Concert          | Amphi%he@ter   | 2     |     202 | 120.00 | T6      |
| E11      | Art Workshop           | M@in Building  | 1     |     101 | 150.00 | T6      |
| E12      | Film Screening         | Theater        | 2     |     203 | 500.00 | T6      |
| E13      | Traditional Dance wars | @udi%orium     | 1     |     102 | 850.00 | T8      |
| E14      | Singing Contest        | M@in Building  | 2     |     203 | 500.00 | T8      |
| E15      | Fashion Exhibition     | Sports Comp%ex | 1     |     103 | 120.00 | T8      |
| E16      | Poetry Slam            | Amphi%he@ter   | 2     |     201 | 100.00 | T8      |
| E17      | Quiz Competition       | @udi%orium     | 1     |     102 | 100.00 | T9      |
| E18      | Tech Talk              | Seminar Hall   | 1     |     101 | 200.00 | T9      |
| E19      | Web Design Contest     | @Computer%Lab% | 2     |     202 | 890.00 | T9      |
| E2       | Paintball Tournament   | Sports Comp%ex | 2     |     201 | 150.00 | T3      |
| E20      | Hackathon              | M@in Building  | 1     |     103 | 650.00 | T9      |
| E21      | Stage Setup            | M@in Building  | 2     |     203 | 900.00 | T12     |
| E22      | Sound and Lighting     | @udi%orium     | 1     |     102 | 250.00 | T12     |
| E23      | Event Coordination     | Sports Comp%ex | 1     |     103 | 500.00 | T12     |
| E24      | Volunteer Management   | M@in Building  | 3     |     303 | 600.00 | T12     |
| E25      | Baking Competition     | Kitchen        | 1     |     101 | 850.00 | T14     |
| E26      | Food Tasting           | Cafeteria      | 2     |     202 | 500.00 | T14     |
| E27      | Cooking Workshop       | M@in Building  | 1     |     102 | 120.00 | T14     |
| E29      | Short Film Screening   | Theater        | 1     |     103 | 400.00 | T15     |
| E3       | Escape Room Challenge  | M@in Building  | 1     |     102 | 120.00 | T3      |
| E30      | Literary Debate        | Seminar Hall   | 1     |     101 | 550.00 | T15     |
| E31      | Writing Competition    | Library        | 2     |     202 | 950.00 | T15     |
| E32      | Poetry Recitation      | Amphi%he@ter   | 1     |     102 | 100.00 | T15     |
| E33      | AI HACATHON            | Seminar Hall   | 2     |     205 | 900.00 | T4      |
| E4       | Photography Contest    | @udi%orium     | 2     |     202 | 850.00 | T4      |
| E5       | Code Jam               | @Computer%Lab% | 1     |     101 | 100.00 | T4      |
| E6       | Robot Wars             | Robotics Arena | 1     |     201 | 100.00 | T4      |
| E7       | Tech Expo              | M@in Building  | 2     |     203 | 550.00 | T4      |
| E8       | Gaming Tournament      | Sports Comp%ex | 1     |     103 | 800.00 | T4      |
| E9       | Drama Play             | @udi%orium     | 1     |     102 | 100.00 | T6      |
+----------+------------------------+----------------+-------+---------+--------+---------+
32 rows in set (0.00 sec)
```

2. Update the quantity of 'Mushroom Risotto' in stall 'S1' to 25.

Sol:

Before:

```
mysql> select * from stall_items;
+----------+--------------------------+----------------+----------------+
| stall_id | item_name                | price_per_unit | total_quantity |
+----------+--------------------------+----------------+----------------+
| S1       | Caprese Salad            |         249.00 |             35 |
| S1       | Classic Caesar Salad     |         349.50 |             45 |
| S1       | Margherita Pizza         |         350.00 |             25 |
| S1       | Mushroom Risotto         |         359.00 |             30 |
| S1       | Spinach and Feta Omelette |         259.00 |             40 |
| S1       | Vegetable Pad Thai       |         329.00 |             25 |
| S1       | Vegetable Stir-Fry       |         299.00 |             30 |
| S1       | Veggie Wrap              |         399.00 |             50 |
| S2       | Chicken Noodle Soup      |         529.00 |             30 |
| S2       | Fish Tacos               |         449.00 |             25 |
| S2       | Mushroom Risotto         |         387.00 |             28 |
| S2       | Vegetable Stir-Fry       |         322.00 |             28 |
| S2       | Veggie Wrap              |         429.00 |             40 |
| S3       | Chicken Noodle Soup      |         564.50 |             25 |
| S3       | Fish Tacos               |         481.50 |             23 |
| S3       | Mushroom Risotto         |         349.00 |             35 |
| S3       | Spinach and Feta Omelette |         282.50 |             38 |
| S3       | Vegetable Stir-Fry       |         293.00 |             35 |
| S3       | Veggie Wrap              |         379.00 |             55 |
| S4       | BBQ Chicken Sandwich     |         489.00 |             20 |
| S4       | Chicken Noodle Soup      |         519.75 |             35 |
| S4       | Fish Tacos               |         470.00 |             28 |
| S4       | Grilled Steak            |         573.75 |             17 |
| S4       | Shrimp Scampi            |         419.00 |             20 |
| S5       | Classic Caesar Salad     |         349.25 |             50 |
| S5       | Margherita Pizza         |         349.00 |             25 |
| S5       | Mushroom Risotto         |         360.00 |             38 |
| S5       | Vegetable Pad Thai       |         324.25 |             28 |
| S5       | Veggie Wrap              |         400.00 |             60 |
```

Command:

```
mysql> UPDATE stall_items SET total_quantity=25 WHERE stall_id ='S1' AND item_name = 'Mushroom Risotto';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

After:

```
mysql> select * from stall_items;
+----------+--------------------------+----------------+----------------+
| stall_id | item_name                | price_per_unit | total_quantity |
+----------+--------------------------+----------------+----------------+
| S1       | Caprese Salad            |         249.00 |             35 |
| S1       | Classic Caesar Salad     |         349.50 |             45 |
| S1       | Margherita Pizza         |         350.00 |             25 |
| S1       | Mushroom Risotto         |         359.00 |             25 |
| S1       | Spinach and Feta Omelette |         259.00 |             40 |
| S1       | Vegetable Pad Thai       |         329.00 |             25 |
| S1       | Vegetable Stir-Fry       |         299.00 |             30 |
| S1       | Veggie Wrap              |         399.00 |             50 |
| S2       | Chicken Noodle Soup      |         529.00 |             30 |
| S2       | Fish Tacos               |         449.00 |             25 |
| S2       | Mushroom Risotto         |         387.00 |             28 |
| S2       | Vegetable Stir-Fry       |         322.00 |             28 |
| S2       | Veggie Wrap              |         429.00 |             40 |
| S3       | Chicken Noodle Soup      |         564.50 |             25 |
| S3       | Fish Tacos               |         481.50 |             23 |
| S3       | Mushroom Risotto         |         349.00 |             35 |
| S3       | Spinach and Feta Omelette |         282.50 |             38 |
| S3       | Vegetable Stir-Fry       |         293.00 |             35 |
| S3       | Veggie Wrap              |         379.00 |             55 |
| S4       | BBQ Chicken Sandwich     |         489.00 |             20 |
| S4       | Chicken Noodle Soup      |         519.75 |             35 |
| S4       | Fish Tacos               |         470.00 |             28 |
| S4       | Grilled Steak            |         573.75 |             17 |
| S4       | Shrimp Scampi            |         419.00 |             20 |
| S5       | Classic Caesar Salad     |         349.25 |             50 |
| S5       | Margherita Pizza         |         349.00 |             25 |
| S5       | Mushroom Risotto         |         360.00 |             38 |
| S5       | Vegetable Pad Thai       |         324.25 |             28 |
| S5       | Veggie Wrap              |         400.00 |             60 |
| S6       | Chicken Noodle Soup      |         550.00 |             28 |
| S6       | Fish Tacos               |         470.00 |             26 |
```

3. Delete all registrations where the event ID is 'E1' and SRN starts with 'P100'

Sol:

Before:

```
mysql> select * from registration;
+----------+-------+-----------------+
| event_id | SRN   | registration_id |
+----------+-------+-----------------+
| E1       | P1001 | R1              |
| E1       | P1002 | R2              |
| E1       | P1003 | R3              |
| E1       | P1004 | R4              |
| E1       | P1005 | R5              |
| E1       | P1006 | R1              |
| E1       | P1008 | R5              |
| E1       | P1017 | R2              |
| E1       | P1022 | R3              |
| E1       | P1025 | R4              |
| E10      | P1001 | R49             |
| E10      | P1005 | R50             |
| E10      | P1021 | R46             |
| E10      | P1022 | R47             |
| E10      | P1023 | R48             |
| E10      | P1024 | R49             |
| E10      | P1025 | R50             |
| E11      | P1001 | R51             |
| E11      | P1002 | R52             |
| E11      | P1003 | R53             |
| E11      | P1004 | R54             |
| E11      | P1005 | R55             |
| E11      | P1021 | R56             |
| E11      | P1022 | R57             |
| E11      | P1023 | R58             |
| E11      | P1024 | R59             |
| E11      | P1025 | R60             |
| E12      | P1006 | R56             |
| E12      | P1007 | R57             |
```

Command:

```
mysql> DELETE FROM registration WHERE  event_id ='E1' AND SRN LIKE 'P100%';
Query OK, 7 rows affected (0.00 sec)
```

After:

```
mysql> select * from registration;
+----------+-------+-----------------+
| event_id | SRN   | registration_id |
+----------+-------+-----------------+
| E1       | P1017 | R2              |
| E1       | P1022 | R3              |
| E1       | P1025 | R4              |
| E10      | P1001 | R49             |
| E10      | P1005 | R50             |
| E10      | P1021 | R46             |
| E10      | P1022 | R47             |
| E10      | P1023 | R48             |
| E10      | P1024 | R49             |
| E10      | P1025 | R50             |
| E11      | P1001 | R51             |
| E11      | P1002 | R52             |
| E11      | P1003 | R53             |
| E11      | P1004 | R54             |
| E11      | P1005 | R55             |
| E11      | P1021 | R56             |
| E11      | P1022 | R57             |
| E11      | P1023 | R58             |
| E11      | P1024 | R59             |
| E11      | P1025 | R60             |
| E12      | P1006 | R56             |
| E12      | P1007 | R57             |
```

4. Insert a new purchase: 'P1017' buys 3 'Fish Tacos' from stall 'S6' at '2025-07-10 14:00:00'.

Sol:

Before:

```
mysql> select * from purchased;
+-------+----------+--------------------------+---------------------+----------+
| SRN   | stall_id | item_name                | timestamp           | quantity |
+-------+----------+--------------------------+---------------------+----------+
| P1001 | S1       | Mushroom Risotto         | 2023-04-15 13:00:05 |        3 |
| P1001 | S1       | Veggie Wrap              | 2023-04-15 12:00:00 |        2 |
| P1001 | S5       | Classic Caesar Salad     | 2022-04-17 10:10:00 |        2 |
| P1001 | S5       | Veggie Wrap              | 2022-04-17 10:10:00 |        3 |
| P1002 | S1       | Caprese Salad            | 2023-04-16 13:33:00 |        3 |
| P1002 | S1       | Classic Caesar Salad     | 2023-04-16 13:33:00 |        3 |
| P1002 | S9       | Margherita Pizza         | 2023-04-16 12:30:00 |        1 |
| P1002 | S9       | Vegetable Stir-Fry       | 2023-04-16 17:19:00 |        5 |
| P1003 | S7       | Fish Tacos               | 2022-04-19 11:25:00 |        2 |
| P1003 | S7       | Veggie Wrap              | 2022-04-19 14:00:00 |        1 |
| P1003 | S8       | Fish Tacos               | 2022-04-19 12:25:00 |        1 |
| P1003 | S8       | Shrimp Scampi            | 2022-04-19 12:25:00 |        2 |
| P1004 | S7       | Margherita Pizza         | 2022-04-19 11:00:00 |        2 |
| P1004 | S7       | Veggie Wrap              | 2022-04-19 11:00:00 |        4 |
| P1005 | S2       | Fish Tacos               | 2023-04-16 10:30:00 |        4 |
| P1005 | S3       | Chicken Noodle Soup      | 2023-04-16 11:05:00 |        2 |
| P1005 | S3       | Fish Tacos               | 2023-04-16 12:15:00 |        5 |
| P1005 | S4       | BBQ Chicken Sandwich     | 2022-04-17 13:20:00 |        2 |
| P1005 | S4       | Chicken Noodle Soup      | 2022-04-17 13:20:00 |        1 |
| P1005 | S4       | Fish Tacos               | 2022-04-17 12:50:00 |        3 |
| P1005 | S4       | Grilled Steak            | 2022-04-17 14:30:00 |        2 |
| P1005 | S4       | Shrimp Scampi            | 2022-04-17 14:30:00 |        1 |
| P1005 | S9       | Mutton Stroganoff        | 2023-04-16 13:35:00 |        1 |
| P1006 | S1       | Margherita Pizza         | 2023-04-16 14:50:00 |        1 |
| P1006 | S2       | Fish Tacos               | 2023-04-16 15:55:00 |        1 |
| P1006 | S9       | Vegetable Stir-Fry       | 2023-04-16 10:45:00 |        2 |
| P1008 | S4       | BBQ Chicken Sandwich     | 2022-04-17 15:10:00 |        1 |
| P1008 | S5       | Mushroom Risotto         | 2022-04-17 15:00:00 |        3 |
| P1008 | S5       | Vegetable Pad Thai       | 2022-04-17 15:00:00 |        1 |
| P1008 | S6       | Fish Tacos               | 2022-04-17 10:45:00 |        3 |
| P1008 | S6       | Grilled Steak            | 2022-04-17 10:45:00 |        2 |
| P1010 | S4       | Chicken Noodle Soup      | 2022-04-17 13:05:00 |        3 |
| P1010 | S5       | Mushroom Risotto         | 2022-04-17 17:15:00 |        1 |
| P1010 | S6       | Shrimp Scampi            | 2022-04-17 17:20:00 |        2 |
| P1010 | S9       | Mutton Stroganoff        | 2023-04-16 13:10:00 |        1 |
| P1010 | S9       | Spinach and Feta Omelette | 2023-04-16 13:10:00 |        1 |
| P1017 | S1       | Classic Caesar Salad     | 2023-04-16 10:05:00 |        3 |
| P1017 | S1       | Spinach and Feta Omelette | 2023-04-16 10:05:00 |        2 |
| P1017 | S2       | Fish Tacos               | 2023-04-16 16:00:00 |        5 |
| P1017 | S3       | Chicken Noodle Soup      | 2023-04-16 12:25:00 |        3 |
| P1017 | S3       | Vegetable Stir-Fry       | 2023-04-16 12:25:00 |        4 |
| P1017 | S9       | Bacon-Wrapped Shrimp     | 2023-04-16 17:30:00 |        2 |
| P1024 | S4       | Chicken Noodle Soup      | 2022-04-17 13:15:00 |        1 |
| P1024 | S5       | Vegetable Pad Thai       | 2022-04-17 15:00:00 |        2 |
| P1024 | S6       | Grilled Steak            | 2022-04-17 11:05:00 |        2 |
| P1024 | S6       | Shrimp Scampi            | 2022-04-17 11:05:00 |        3 |
+-------+----------+--------------------------+---------------------+----------+
46 rows in set (0.00 sec)
```

Command:

```
mysql> INSERT INTO purchased(SRN, stall_id,item_name, timestamp,quantity) VALUES ('P1017','S6','Fish Tacos','2025-07-10 14:00:00', '3');
Query OK, 1 row affected (0.00 sec)
```

After:

```
mysql> select * from purchased;
+-------+----------+--------------------------+---------------------+----------+
| SRN   | stall_id | item_name                | timestamp           | quantity |
+-------+----------+--------------------------+---------------------+----------+
| P1001 | S1       | Mushroom Risotto         | 2023-04-15 13:00:05 |        3 |
| P1001 | S1       | Veggie Wrap              | 2023-04-15 12:00:00 |        2 |
| P1001 | S5       | Classic Caesar Salad     | 2022-04-17 10:10:00 |        2 |
| P1001 | S5       | Veggie Wrap              | 2022-04-17 10:10:00 |        3 |
| P1002 | S1       | Caprese Salad            | 2023-04-16 13:33:00 |        3 |
| P1002 | S1       | Classic Caesar Salad     | 2023-04-16 13:33:00 |        3 |
| P1002 | S9       | Margherita Pizza         | 2023-04-16 12:30:00 |        1 |
| P1002 | S9       | Vegetable Stir-Fry       | 2023-04-16 17:19:00 |        5 |
| P1003 | S7       | Fish Tacos               | 2022-04-19 11:25:00 |        2 |
| P1003 | S7       | Veggie Wrap              | 2022-04-19 14:00:00 |        1 |
| P1003 | S8       | Fish Tacos               | 2022-04-19 12:25:00 |        1 |
| P1003 | S8       | Shrimp Scampi            | 2022-04-19 12:25:00 |        2 |
| P1004 | S7       | Margherita Pizza         | 2022-04-19 11:00:00 |        2 |
| P1004 | S7       | Veggie Wrap              | 2022-04-19 11:00:00 |        4 |
| P1005 | S2       | Fish Tacos               | 2023-04-16 10:30:00 |        4 |
| P1005 | S3       | Chicken Noodle Soup      | 2023-04-16 11:05:00 |        2 |
| P1005 | S3       | Fish Tacos               | 2023-04-16 12:15:00 |        5 |
| P1005 | S4       | BBQ Chicken Sandwich     | 2022-04-17 13:20:00 |        2 |
| P1005 | S4       | Chicken Noodle Soup      | 2022-04-17 13:20:00 |        1 |
| P1005 | S4       | Fish Tacos               | 2022-04-17 12:50:00 |        3 |
| P1005 | S4       | Grilled Steak            | 2022-04-17 14:30:00 |        2 |
| P1005 | S4       | Shrimp Scampi            | 2022-04-17 14:30:00 |        1 |
| P1005 | S9       | Mutton Stroganoff        | 2023-04-16 13:35:00 |        1 |
| P1006 | S1       | Margherita Pizza         | 2023-04-16 14:50:00 |        1 |
| P1006 | S2       | Fish Tacos               | 2023-04-16 15:55:00 |        1 |
| P1006 | S9       | Vegetable Stir-Fry       | 2023-04-16 10:45:00 |        2 |
| P1008 | S4       | BBQ Chicken Sandwich     | 2022-04-17 15:10:00 |        1 |
| P1008 | S5       | Mushroom Risotto         | 2022-04-17 15:00:00 |        3 |
| P1008 | S5       | Vegetable Pad Thai       | 2022-04-17 15:00:00 |        1 |
| P1008 | S6       | Fish Tacos               | 2022-04-17 10:45:00 |        3 |
| P1008 | S6       | Grilled Steak            | 2022-04-17 10:45:00 |        2 |
| P1010 | S4       | Chicken Noodle Soup      | 2022-04-17 13:05:00 |        3 |
| P1010 | S5       | Mushroom Risotto         | 2022-04-17 17:15:00 |        1 |
| P1010 | S6       | Shrimp Scampi            | 2022-04-17 17:20:00 |        2 |
| P1010 | S9       | Mutton Stroganoff        | 2023-04-16 13:10:00 |        1 |
| P1010 | S9       | Spinach and Feta Omelette | 2023-04-16 13:10:00 |        1 |
| P1017 | S1       | Classic Caesar Salad     | 2023-04-16 10:05:00 |        3 |
| P1017 | S1       | Spinach and Feta Omelette | 2023-04-16 10:05:00 |        2 |
| P1017 | S2       | Fish Tacos               | 2023-04-16 16:00:00 |        5 |
| P1017 | S3       | Chicken Noodle Soup      | 2023-04-16 12:25:00 |        3 |
| P1017 | S3       | Vegetable Stir-Fry       | 2023-04-16 12:25:00 |        4 |
| P1017 | S6       | Fish Tacos               | 2025-07-10 14:00:00 |        3 |
| P1017 | S9       | Bacon-Wrapped Shrimp     | 2023-04-16 17:30:00 |        2 |
| P1024 | S4       | Chicken Noodle Soup      | 2022-04-17 13:15:00 |        1 |
| P1024 | S5       | Vegetable Pad Thai       | 2022-04-17 15:00:00 |        2 |
| P1024 | S6       | Grilled Steak            | 2022-04-17 11:05:00 |        2 |
| P1024 | S6       | Shrimp Scampi            | 2022-04-17 11:05:00 |        3 |
+-------+----------+--------------------------+---------------------+----------+
47 rows in set (0.00 sec)
```

TASK2

5. **Retrieve participants's SRN who are registered only for event 'E2' or 'E5', but not both (using SET operations).**

Sol:

Before:

```
mysql> select * from registration;
+----------+-------+-----------------+
| event_id | SRN   | registration_id |
+----------+-------+-----------------+
| E1       | P1017 | R2              |
| E1       | P1022 | R3              |
| E1       | P1025 | R4              |
| E10      | P1001 | R49             |
| E10      | P1005 | R50             |
| E10      | P1021 | R46             |
| E10      | P1022 | R47             |
| E10      | P1023 | R48             |
| E10      | P1024 | R49             |
| E10      | P1025 | R50             |
| E11      | P1001 | R51             |
| E11      | P1002 | R52             |
| E11      | P1003 | R53             |
| E11      | P1004 | R54             |
| E11      | P1005 | R55             |
| E11      | P1021 | R56             |
| E11      | P1022 | R57             |
| E11      | P1023 | R58             |
| E11      | P1024 | R59             |
| E11      | P1025 | R60             |
| E12      | P1006 | R56             |
| E12      | P1007 | R57             |
| E12      | P1008 | R58             |
| E12      | P1009 | R59             |
| E12      | P1010 | R60             |
| E13      | P1011 | R61             |
| E13      | P1012 | R62             |
| E13      | P1013 | R63             |
| E13      | P1014 | R64             |
| E13      | P1015 | R65             |
| E14      | P1016 | R66             |
| E14      | P1017 | R67             |
| E14      | P1018 | R68             |
| E14      | P1019 | R69             |
| E14      | P1020 | R70             |
| E15      | P1021 | R71             |
| E15      | P1022 | R72             |
| E15      | P1023 | R73             |
| E15      | P1024 | R74             |
| E15      | P1025 | R75             |
| E16      | P1001 | R76             |
| E16      | P1002 | R77             |
| E16      | P1003 | R78             |
| E16      | P1004 | R79             |
| E16      | P1005 | R80             |
| E16      | P1021 | R71             |
| E16      | P1022 | R72             |
| E16      | P1023 | R73             |
| E16      | P1024 | R74             |
| E16      | P1025 | R75             |
| E17      | P1006 | R81             |
| E17      | P1007 | R82             |
```

Command and after :

```
mysql> SELECT SRN FROM registration WHERE event_id IN ('E2', 'E5') GROUP BY SRN HAVING COUNT(DISTINCT event_id) = 1;
+--------+
| SRN    |
+--------+
| P1004  |
| P1005  |
| P1006  |
| P1007  |
| P1008  |
| P1009  |
| P1010  |
| P1021  |
| P1022  |
| P1023  |
| P1024  |
| P1025  |
+--------+
```

6. **Display all participants and the names of all their visitors(if any) with a count of visitors**

Sol:

Before:

```
mysql> select * from visitor;
+--------+--------------------+------+--------+
| SRN    | name               | age  | gender |
+--------+--------------------+------+--------+
| P1001  | David Wilson       |   35 | Male   |
| P1001  | John Doe           |   25 | Male   |
| P1001  | Michael Johnson    |   22 | Male   |
| P1002  | Jane Smith         |   30 | Female |
| P1003  | Andrew Thomas      |   26 | Male   |
| P1003  | Daniel Taylor      |   23 | Male   |
| P1014  | Emily Davis        |   28 | Female |
| P1014  | Sophia Brown       |   27 | Female |
| P1016  | Isabella Martinez  |   31 | Female |
| P1016  | Olivia Anderson    |   29 | Female |
+--------+--------------------+------+--------+
10 rows in set (0.00 sec)
```

Command and after:

```
mysql> SELECT SRN, GROUP_CONCAT(name SEPARATOR ', ') AS visitor_names, COUNT(name) AS visitor_count FROM visitor GROUP BY SRN;
+--------+----------------------------------------+---------------+
| SRN    | visitor_names                          | visitor_count |
+--------+----------------------------------------+---------------+
| P1001  | David Wilson, John Doe, Michael Johnson |             3 |
| P1002  | Jane Smith                             |             1 |
| P1003  | Andrew Thomas, Daniel Taylor           |             2 |
| P1014  | Emily Davis, Sophia Brown              |             2 |
| P1016  | Isabella Martinez, Olivia Anderson     |             2 |
+--------+----------------------------------------+---------------+
5 rows in set (0.00 sec)
```

7. **List events that have equal number of male and female participants.**

Sol:

Before:

```
mysql> desc registration;
+-----------------+-------------+------+-----+---------+-------+
| Field           | Type        | Null | Key | Default | Extra |
+-----------------+-------------+------+-----+---------+-------+
| event_id        | varchar(5)  | NO   | PRI | NULL    |       |
| SRN             | varchar(10) | NO   | PRI | NULL    |       |
| registration_id | varchar(5)  | NO   |     | NULL    |       |
+-----------------+-------------+------+-----+---------+-------+
3 rows in set (0.01 sec)
```

```
mysql> desc participant;
+------------+----------------------+------+-----+---------+-------+
| Field      | Type                 | Null | Key | Default | Extra |
+------------+----------------------+------+-----+---------+-------+
| SRN        | varchar(10)          | NO   | PRI | NULL    |       |
| name       | varchar(25)          | NO   |     | NULL    |       |
| department | varchar(20)          | YES  |     | NULL    |       |
| semester   | int                  | YES  |     | NULL    |       |
| gender     | enum('Male','Female') | YES  |     | NULL    |       |
+------------+----------------------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

Command and after:

```
mysql> SELECT r.event_id FROM registration r JOIN participant p ON r.SRN = p.SRN GROUP BY r.event_id HAVING SUM(CASE WHE
N p.gender = 'Male' THEN 1 ELSE 0 END) = SUM(CASE WHEN p.gender = 'Female' THEN 1 ELSE 0 END);
Empty set (0.01 sec)
```

8. **Display each event's name and a binary indicator of whether it occurred after the Golden Jubilee (year > 2047).**
   Sol:
   Before:

```
mysql> select * from event_conduction;
+----------+--------------------+
| event_id | date_of_conduction |
+----------+--------------------+
| E1       | 2023-04-15         |
| E10      | 2022-04-17         |
| E11      | 2022-04-18         |
| E12      | 2022-04-19         |
| E13      | 2022-04-15         |
| E14      | 2022-04-16         |
| E15      | 2022-04-17         |
| E16      | 2022-04-18         |
| E17      | 2022-04-19         |
| E17      | 2022-04-20         |
| E18      | 2022-04-15         |
| E19      | 2022-04-16         |
| E2       | 2023-04-16         |
| E20      | 2022-04-17         |
| E21      | 2022-04-18         |
| E22      | 2022-04-19         |
| E23      | 2022-04-15         |
| E24      | 2022-04-16         |
| E25      | 2021-04-15         |
| E26      | 2021-04-16         |
| E27      | 2021-04-17         |
| E29      | 2021-04-19         |
| E3       | 2023-04-17         |
| E30      | 2021-04-15         |
| E31      | 2021-04-16         |
| E32      | 2021-04-17         |
| E32      | 2021-04-18         |
| E4       | 2023-04-18         |
| E5       | 2023-04-19         |
| E6       | 2023-04-15         |
| E7       | 2023-04-16         |
| E8       | 2023-04-17         |
| E9       | 2022-04-15         |
| E9       | 2022-04-16         |
+----------+--------------------+
```

Command:

```
mysql> SELECT
    ->     event_id,
    ->     date_of_conduction,
    ->     CASE
    ->         WHEN YEAR(date_of_conduction) > 2047 THEN 1
    ->         ELSE 0
    ->     END AS after_golden_jubilee
    -> FROM event_conduction;
```

After:

```
+----------+--------------------+----------------------+
| event_id | date_of_conduction | after_golden_jubilee |
+----------+--------------------+----------------------+
| E1       | 2023-04-15         |                    0 |
| E10      | 2022-04-17         |                    0 |
| E11      | 2022-04-18         |                    0 |
| E12      | 2022-04-19         |                    0 |
| E13      | 2022-04-15         |                    0 |
| E14      | 2022-04-16         |                    0 |
| E15      | 2022-04-17         |                    0 |
| E16      | 2022-04-18         |                    0 |
| E17      | 2022-04-19         |                    0 |
| E17      | 2022-04-20         |                    0 |
| E18      | 2022-04-15         |                    0 |
| E19      | 2022-04-16         |                    0 |
| E2       | 2023-04-16         |                    0 |
| E20      | 2022-04-17         |                    0 |
| E21      | 2022-04-18         |                    0 |
| E22      | 2022-04-19         |                    0 |
| E23      | 2022-04-15         |                    0 |
| E24      | 2022-04-16         |                    0 |
| E25      | 2021-04-15         |                    0 |
| E26      | 2021-04-16         |                    0 |
| E27      | 2021-04-17         |                    0 |
| E29      | 2021-04-19         |                    0 |
| E3       | 2023-04-17         |                    0 |
| E30      | 2021-04-15         |                    0 |
| E31      | 2021-04-16         |                    0 |
| E32      | 2021-04-17         |                    0 |
| E32      | 2021-04-18         |                    0 |
| E4       | 2023-04-18         |                    0 |
| E5       | 2023-04-19         |                    0 |
| E6       | 2023-04-15         |                    0 |
| E7       | 2023-04-16         |                    0 |
| E8       | 2023-04-17         |                    0 |
| E9       | 2022-04-15         |                    0 |
| E9       | 2022-04-16         |                    0 |
+----------+--------------------+----------------------+
34 rows in set (0.01 sec)
```

Task3:

1. **Which SQL command is used to insert new records into a table?**

Sol: The INSERT INTO command is used to add new rows of data into a table.

2. **What is the difference between DELETE and TRUNCATE in SQL?**

Sol: DELETE is a DML (Data Manipulation Language) command that removes rows one by one. It can be filtered with a WHERE clause and is a transactional operation, allowing it to be rolled back.

TRUNCATE is a DDL (Data Definition Language) command that removes all rows quickly by deallocating the data pages. It's much faster than DELETE, cannot be filtered with a WHERE clause, and is generally non-transactional.

3. **Which DML command is used to modify existing values in a table?**

Sol: The UPDATE command is used to modify existing values in a table.

4. **Can a single INSERT statement add multiple rows at once?**

Sol: Yes, a single INSERT statement can add multiple rows at once by providing a comma-separated list of value sets.

5. **Which DML command is used to fetch records from one or more tables?**

Sol: The SELECT command is used to retrieve records from one or more tables.

6. **What is the purpose of using a JOIN in SQL?**

Sol: A JOIN is used to combine rows from two or more tables based on a related column between them, allowing you to query data from multiple sources at once.

7. **Which JOIN returns only the rows that have matching values in both tables?**

Sol: An INNER JOIN returns only the rows that have matching values in both tables.

8. **What does a LEFT JOIN return when there is no matching row in the right table?**

Sol: A LEFT JOIN returns all rows from the left table and NULL values for the columns from the right table where there is no matching row.

9. **Can MySQL directly perform a FULL OUTER JOIN? If not, how can it be achieved?**

Sol: No, MySQL does not directly support the FULL OUTER JOIN command. It can be simulated by using a UNION to combine the results of a LEFT JOIN and a RIGHT JOIN.

10. **What is the difference between INNER JOIN and CROSS JOIN?**

Sol: An INNER JOIN returns only the matching rows based on a specified condition. A CROSS JOIN returns the Cartesian product of the two tables, which means it combines every row from the first table with every row from the second table, without any condition.