DBMS: LAB 9

Functions, and Procedures

University Fest Management System

DATE:8-10-24

....

NAME: Abuthwahir H M

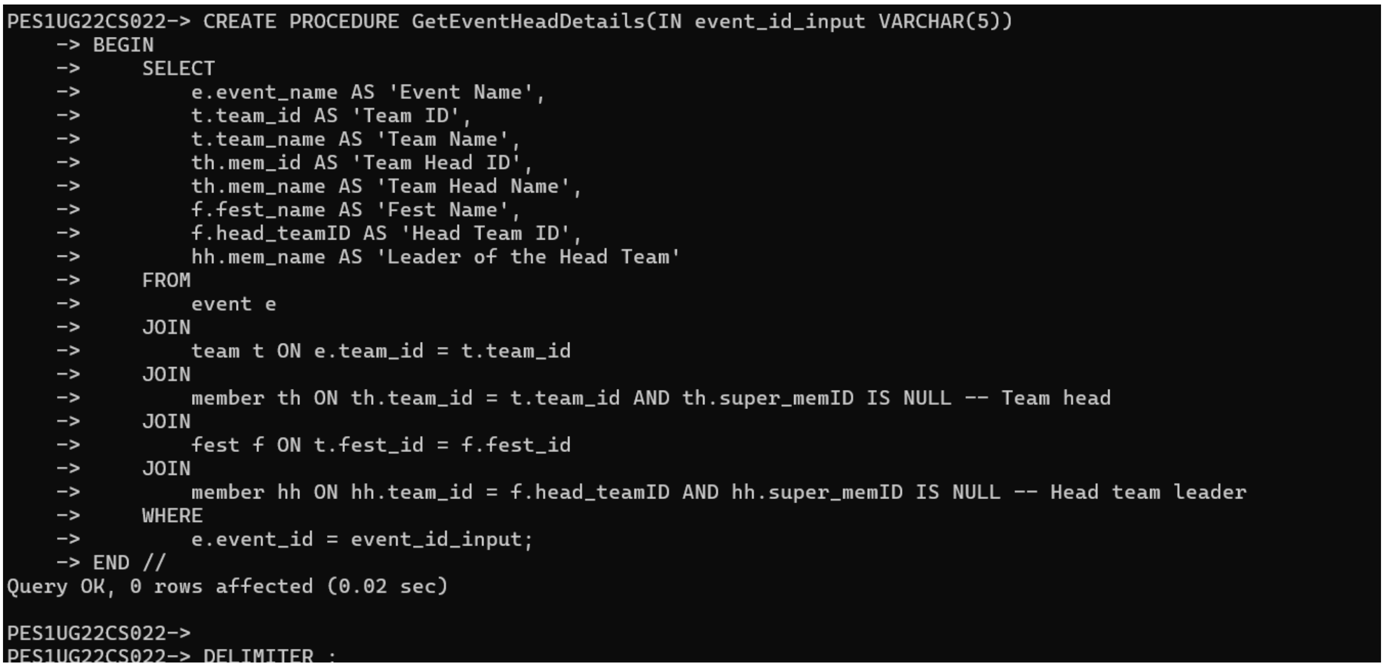
SRN: PES1UG22CS022

SECTION: ‘A’ Sec

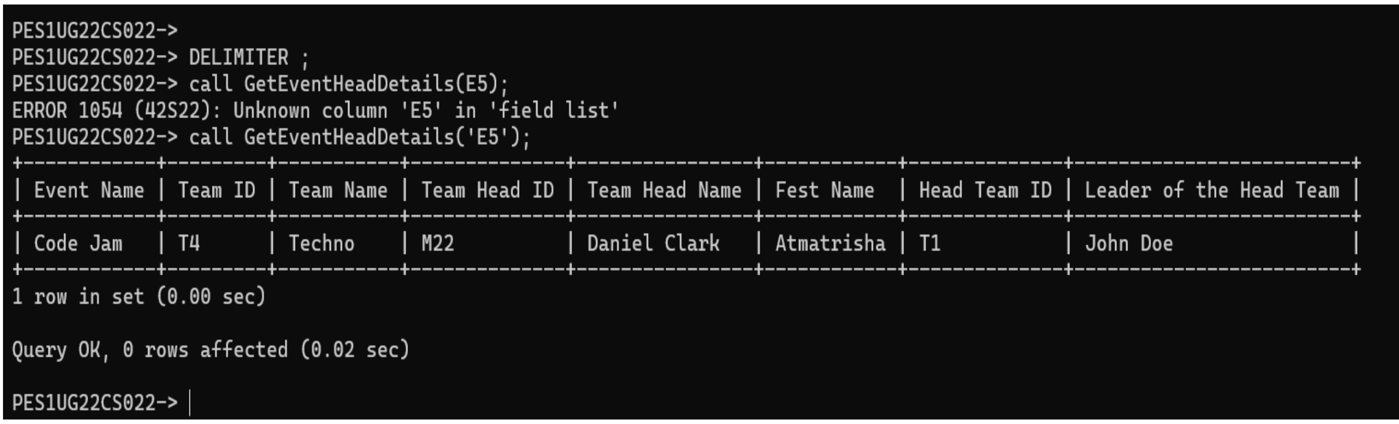
....

TASK 1:

QUESTION 1:

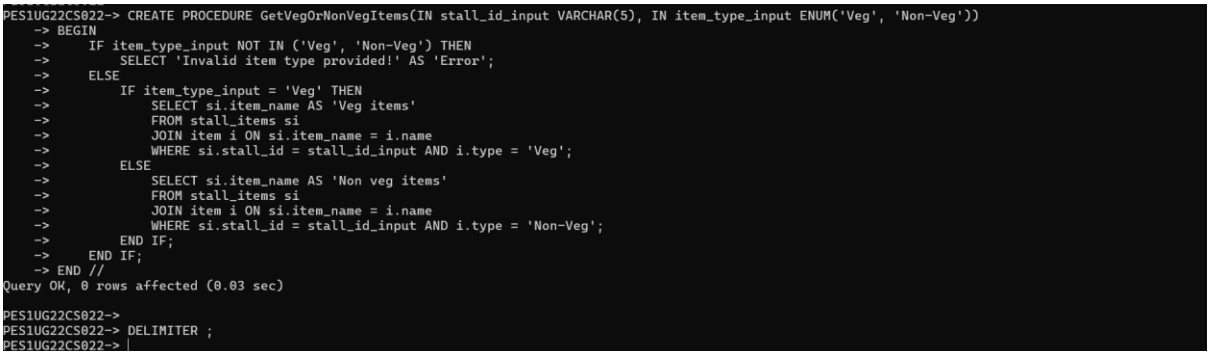


```
PES1UG22CS022-> CREATE PROCEDURE GetEventHeadDetails(IN event_id_input VARCHAR(5))
    -> BEGIN
    ->     SELECT
    ->         e.event_name AS 'Event Name',
    ->         t.team_id AS 'Team ID',
    ->         t.team_name AS 'Team Name',
    ->         th.mem_id AS 'Team Head ID',
    ->         th.mem_name AS 'Team Head Name',
    ->         f.fest_name AS 'Fest Name',
    ->         f.head_teamID AS 'Head Team ID',
    ->         hh.mem_name AS 'Leader of the Head Team'
    ->     FROM
    ->         event e
    ->     JOIN
    ->         team t ON e.team_id = t.team_id
    ->     JOIN
    ->         member th ON th.team_id = t.team_id AND th.super_memID IS NULL -- Team head
    ->     JOIN
    ->         fest f ON t.fest_id = f.fest_id
    ->     JOIN
    ->         member hh ON hh.team_id = f.head_teamID AND hh.super_memID IS NULL -- Head team leader
    ->     WHERE
    ->         e.event_id = event_id_input;
    -> END //
Query OK, 0 rows affected (0.02 sec)

PES1UG22CS022->
PES1UG22CS022-> DELIMITER ;
```

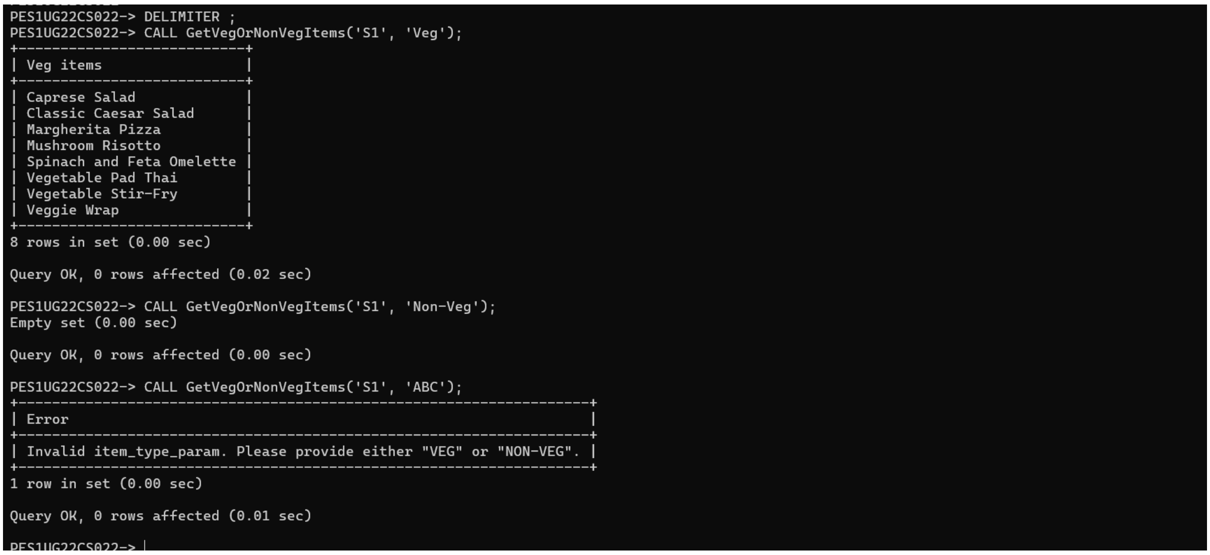


```
PES1UG22CS022->
PES1UG22CS022-> DELIMITER ;
PES1UG22CS022-> call GetEventHeadDetails(E5);
ERROR 1054 (42S22): Unknown column 'E5' in 'field list'
PES1UG22CS022-> call GetEventHeadDetails('E5');
+------------+---------+-----------+--------------+----------------+------------+--------------+-------------------------+
| Event Name | Team ID | Team Name | Team Head ID | Team Head Name | Fest Name  | Head Team ID | Leader of the Head Team |
+------------+---------+-----------+--------------+----------------+------------+--------------+-------------------------+
| Code Jam   | T4      | Techno    | M22          | Daniel Clark   | Atmatrisha | T1           | John Doe                |
+------------+---------+-----------+--------------+----------------+------------+--------------+-------------------------+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.02 sec)

PES1UG22CS022-> |
```

QUESTION 2:

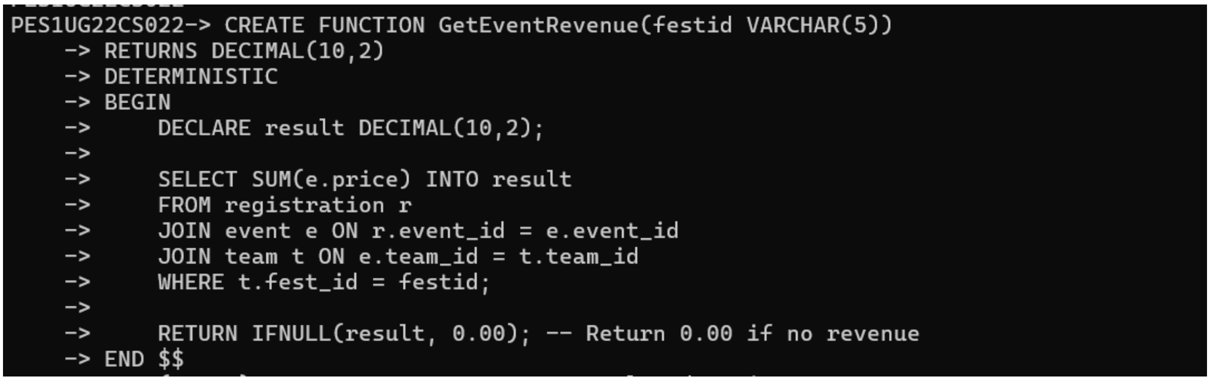


```
PES1UG22CS022-> CREATE PROCEDURE GetVegOrNonVegItems(IN stall_id_input VARCHAR(5), IN item_type_input ENUM('Veg', 'Non-Veg'))
    -> BEGIN
    ->     IF item_type_input NOT IN ('Veg', 'Non-Veg') THEN
    ->         SELECT 'Invalid item type provided!' AS 'Error';
    ->     ELSE
    ->         IF item_type_input = 'Veg' THEN
    ->             SELECT si.item_name AS 'Veg items'
    ->             FROM stall_items si
    ->             JOIN item i ON si.item_name = i.name
    ->             WHERE si.stall_id = stall_id_input AND i.type = 'Veg';
    ->         ELSE
    ->             SELECT si.item_name AS 'Non veg items'
    ->             FROM stall_items si
    ->             JOIN item i ON si.item_name = i.name
    ->             WHERE si.stall_id = stall_id_input AND i.type = 'Non-Veg';
    ->         END IF;
    ->     END IF;
    -> END //
Query OK, 0 rows affected (0.03 sec)

PES1UG22CS022->
PES1UG22CS022-> DELIMITER ;
PES1UG22CS022-> |
```

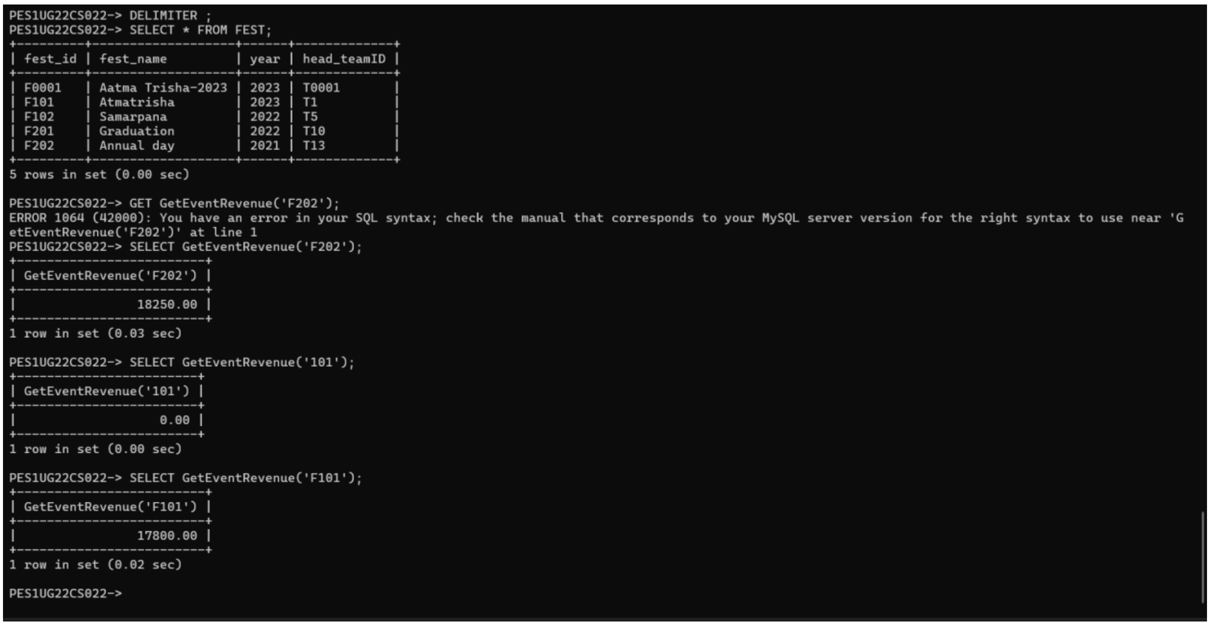


```
PES1UG22CS022-> DELIMITER ;
PES1UG22CS022-> CALL GetVegOrNonVegItems('S1', 'Veg');
+---------------------------+
| Veg items                 |
+---------------------------+
| Caprese Salad             |
| Classic Caesar Salad      |
| Margherita Pizza          |
| Mushroom Risotto          |
| Spinach and Feta Omelette |
| Vegetable Pad Thai        |
| Vegetable Stir-Fry        |
| Veggie Wrap               |
+---------------------------+
8 rows in set (0.00 sec)

Query OK, 0 rows affected (0.02 sec)

PES1UG22CS022-> CALL GetVegOrNonVegItems('S1', 'Non-Veg');
Empty set (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

PES1UG22CS022-> CALL GetVegOrNonVegItems('S1', 'ABC');
+--------------------------------------------------------------------+
| Error                                                              |
+--------------------------------------------------------------------+
| Invalid item_type_param. Please provide either "VEG" or "NON-VEG". |
+--------------------------------------------------------------------+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.01 sec)

PES1UG22CS022-> |
```

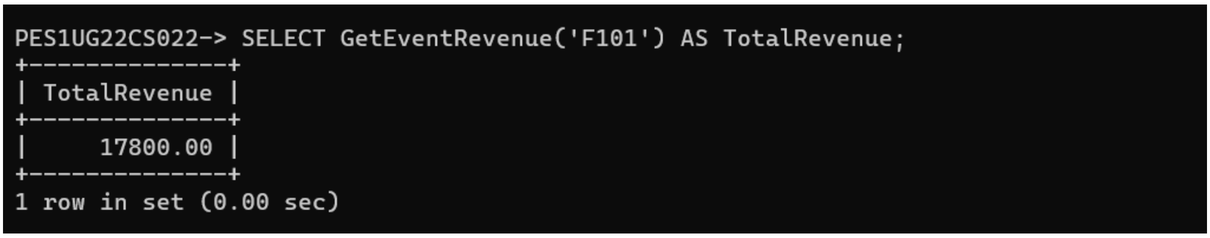
QUESTION 3:



```
PES1UG22CS022-> CREATE FUNCTION GetEventRevenue(festid VARCHAR(5))
    -> RETURNS DECIMAL(10,2)
    -> DETERMINISTIC
    -> BEGIN
    ->     DECLARE result DECIMAL(10,2);
    ->
    ->     SELECT SUM(e.price) INTO result
    ->     FROM registration r
    ->     JOIN event e ON r.event_id = e.event_id
    ->     JOIN team t ON e.team_id = t.team_id
    ->     WHERE t.fest_id = festid;
    ->
    ->     RETURN IFNULL(result, 0.00); -- Return 0.00 if no revenue
    -> END $$
```

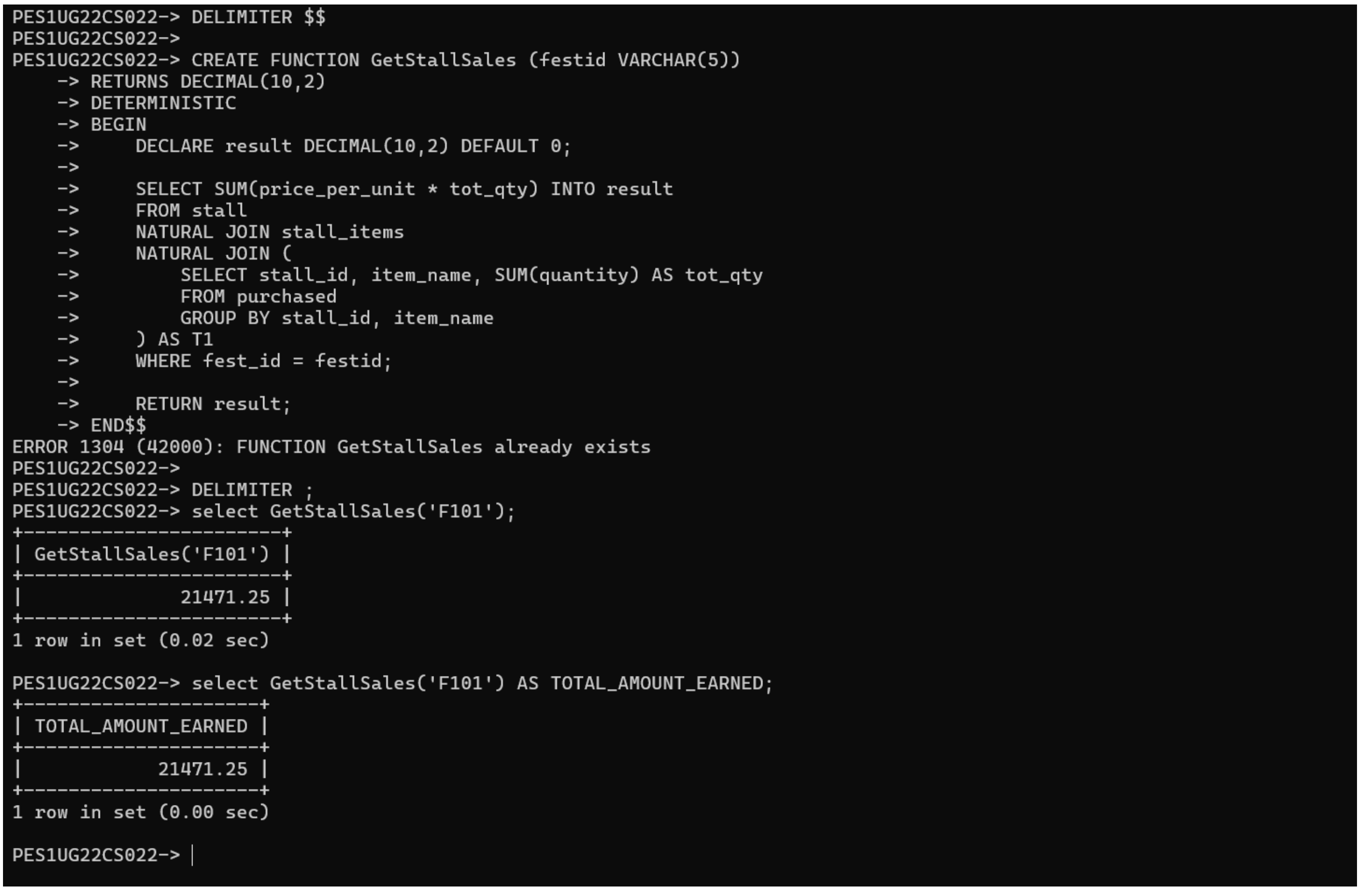


```
PES1UG22CS022-> DELIMITER ;
PES1UG22CS022-> SELECT * FROM FEST;
+---------+-------------------+------+-------------+
| fest_id | fest_name         | year | head_teamID |
+---------+-------------------+------+-------------+
| F0001   | Aatma Trisha-2023 | 2023 | T0001       |
| F101    | Atmatrisha        | 2023 | T1          |
| F102    | Samarpana         | 2022 | T5          |
| F201    | Graduation        | 2022 | T10         |
| F202    | Annual day        | 2021 | T13         |
+---------+-------------------+------+-------------+
5 rows in set (0.00 sec)

PES1UG22CS022-> GET GetEventRevenue('F202');
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'G
etEventRevenue('F202')' at line 1
PES1UG22CS022-> SELECT GetEventRevenue('F202');
+-------------------------+
| GetEventRevenue('F202') |
+-------------------------+
|                18250.00 |
+-------------------------+
1 row in set (0.03 sec)

PES1UG22CS022-> SELECT GetEventRevenue('101');
+------------------------+
| GetEventRevenue('101') |
+------------------------+
|                   0.00 |
+------------------------+
1 row in set (0.00 sec)

PES1UG22CS022-> SELECT GetEventRevenue('F101');
+-------------------------+
| GetEventRevenue('F101') |
+-------------------------+
|                17800.00 |
+-------------------------+
1 row in set (0.02 sec)

PES1UG22CS022->
```



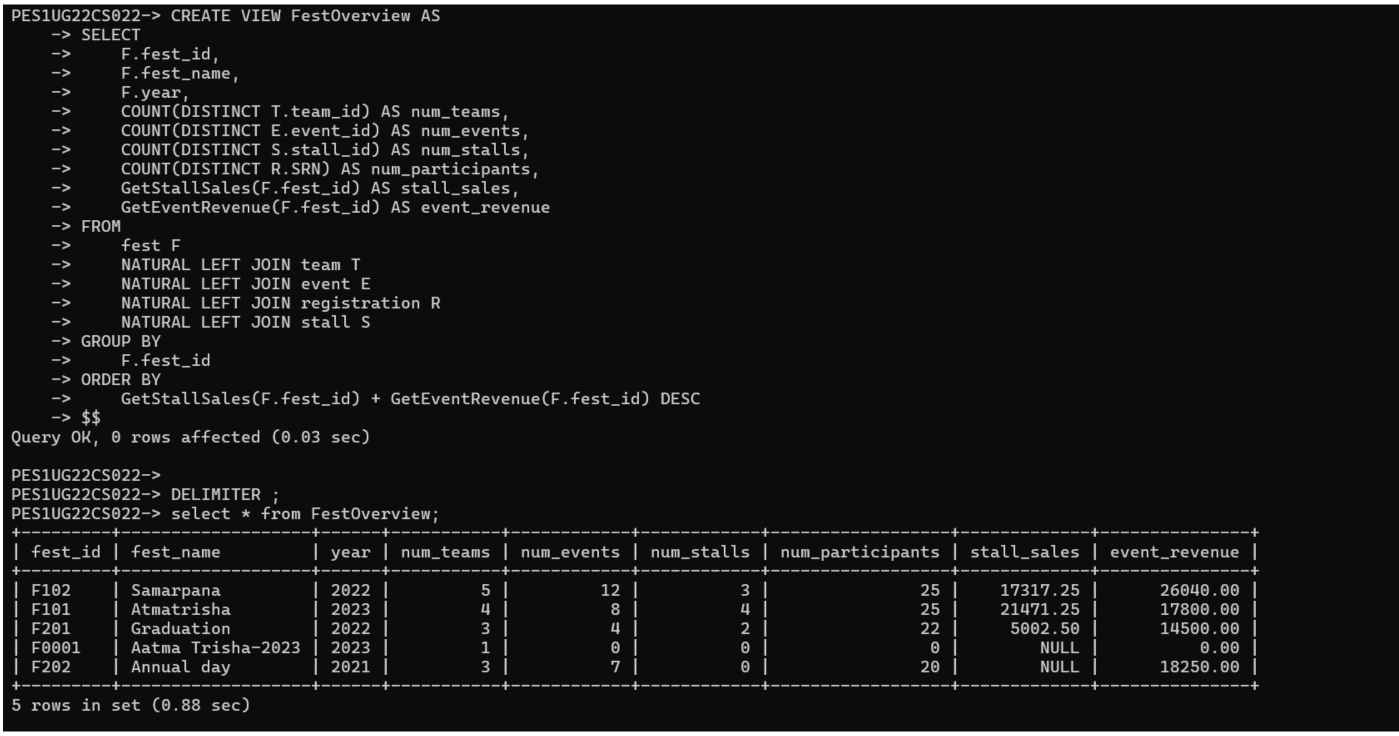
```
PES1UG22CS022-> SELECT GetEventRevenue('F101') AS TotalRevenue;
+--------------+
| TotalRevenue |
+--------------+
|     17800.00 |
+--------------+
1 row in set (0.00 sec)
```

QUESTION 4:



```
PES1UG22CS022-> DELIMITER $$
PES1UG22CS022->
PES1UG22CS022-> CREATE FUNCTION GetStallSales (festid VARCHAR(5))
    -> RETURNS DECIMAL(10,2)
    -> DETERMINISTIC
    -> BEGIN
    ->     DECLARE result DECIMAL(10,2) DEFAULT 0;
    ->
    ->     SELECT SUM(price_per_unit * tot_qty) INTO result
    ->     FROM stall
    ->     NATURAL JOIN stall_items
    ->     NATURAL JOIN (
    ->         SELECT stall_id, item_name, SUM(quantity) AS tot_qty
    ->         FROM purchased
    ->         GROUP BY stall_id, item_name
    ->     ) AS T1
    ->     WHERE fest_id = festid;
    ->
    ->     RETURN result;
    -> END$$
ERROR 1304 (42000): FUNCTION GetStallSales already exists
PES1UG22CS022->
PES1UG22CS022-> DELIMITER ;
PES1UG22CS022-> select GetStallSales('F101');
+-----------------------+
| GetStallSales('F101') |
+-----------------------+
|              21471.25 |
+-----------------------+
1 row in set (0.02 sec)

PES1UG22CS022-> select GetStallSales('F101') AS TOTAL_AMOUNT_EARNED;
+---------------------+
| TOTAL_AMOUNT_EARNED |
+---------------------+
|            21471.25 |
+---------------------+
1 row in set (0.00 sec)

PES1UG22CS022->
```

QUESTION 5:



```
PES1UG22CS022-> CREATE VIEW FestOverview AS
    -> SELECT
    ->     F.fest_id,
    ->     F.fest_name,
    ->     F.year,
    ->     COUNT(DISTINCT T.team_id) AS num_teams,
    ->     COUNT(DISTINCT E.event_id) AS num_events,
    ->     COUNT(DISTINCT S.stall_id) AS num_stalls,
    ->     COUNT(DISTINCT R.SRN) AS num_participants,
    ->     GetStallSales(F.fest_id) AS stall_sales,
    ->     GetEventRevenue(F.fest_id) AS event_revenue
    -> FROM
    ->     fest F
    ->     NATURAL LEFT JOIN team T
    ->     NATURAL LEFT JOIN event E
    ->     NATURAL LEFT JOIN registration R
    ->     NATURAL LEFT JOIN stall S
    -> GROUP BY
    ->     F.fest_id
    -> ORDER BY
    ->     GetStallSales(F.fest_id) + GetEventRevenue(F.fest_id) DESC
    -> $$
Query OK, 0 rows affected (0.03 sec)

PES1UG22CS022->
PES1UG22CS022-> DELIMITER ;
PES1UG22CS022-> select * from FestOverview;
+---------+-------------------+------+-----------+------------+------------+------------------+-------------+---------------+
| fest_id | fest_name         | year | num_teams | num_events | num_stalls | num_participants | stall_sales | event_revenue |
+---------+-------------------+------+-----------+------------+------------+------------------+-------------+---------------+
| F102    | Samarpana         | 2022 |         5 |         12 |          3 |               25 |    17317.25 |      26040.00 |
| F101    | Atmatrisha        | 2023 |         4 |          8 |          4 |               25 |    21471.25 |      17800.00 |
| F201    | Graduation        | 2022 |         3 |          4 |          2 |               22 |     5002.50 |      14500.00 |
| F0001   | Aatma Trisha-2023 | 2023 |         1 |          0 |          0 |                0 |        NULL |          0.00 |
| F202    | Annual day        | 2021 |         3 |          7 |          0 |               20 |        NULL |      18250.00 |
+---------+-------------------+------+-----------+------------+------------+------------------+-------------+---------------+
5 rows in set (0.88 sec)
```