Curtin University – Department of Computing

Assignment Cover Sheet / Declaration of Originality

Last name:	troyee	Student ID:	2663281
Other name(s):	Dhrubo jouti das		
Unit name:	Database Systems	Unit ID:	SYS2014/ISYS 5008
Lecturer / unit coordinator:	Dr. Nimalika Fernando	Tutor:	
Date of submission:	24/10/2024	Vhich Assignment?	Final

Declaration

I declare that:
- The above information is complete and accurate.
- The work I am submitting is entirely my own, except where clearly indicated otherwise and correctly referenced.
- I have taken (and will continue to take) all reasonable steps to ensure my work is not accessible to any other students who may gain unfair advantage from it.
- I have not previously submitted this work for any other unit, whether at Curtin University or elsewhere, or for prior attempts at this unit, except where clearly indicated otherwise.

I understand that:
- Plagiarism and collusion are dishonest, and unfair to all other students.
- Detection of plagiarism and collusion may be done manually or by using tools (such as Turnitin).
- If I plagiarise or collude, I risk failing the unit with a grade of ANN ("Result Annulled due to Academic Misconduct"), which will remain permanently on my academic record. I also risk termination from my course and other penalties.
- Even with correct referencing, my submission will only be marked according to what I have done myself, specifically for this assessment. I cannot re-use the work of others, or my own previously submitted work, in order to fulfil the assessment requirements.
- It is my responsibility to ensure that my submission is complete, correct and not corrupted.

Signature : Dhrubo Troyee *Date of Signature:24/10/2024*

1. Introduction

The design and execution of a MySQL relational database system created for the 2024 Olympic Games are described in this report. This database provides a foundation for querying data in a variety of formats and supports advanced database operations including stored procedures and views. Its goal is to handle information on athletes, events, medallist, teams, total medals, and schedules, coach.

In the user guide it is mentioned that there **are 5 SQl files (CreateTable.sql , LoadData.sql , Query.sql, advanceQuery.sql , DeleteTable.sql)** to efficiently use the Olympic game database . Again, there are 7 csv files that contains information to loading data into respective table like (athelete.csv , coach.csv , team.csv , schedule.csv , medallist.csv, medal_total.csv and events.csv).Additionally, it also contains **4 python file** to be connected the MYSQL Server in a python environments and do some operation such as **(pythonConnect.py , eventdata.py , sched uleData.py insert.py).** I also create a command. Txt file to see all the command together.

With an emphasis on efficiency and flexibility, the project included data modelling, database implementation, data loading, and query creation. Moreover, MySQL 8.0.39 has been used to implement the database on a Linux host running Ubuntu that can be accessed via the VMware Horizontal View Client. The design, implementation, data loading procedures, query usage, and complex database capabilities like stored procedures and views will all be covered in this study. The study will also examine the difficulties encountered and make recommendations for modifications for upcoming system versions.

2. Design of Database

2.1 Entity Sets

ENTITY SET	KEY	OTHER ATTRIBUTES
COACH	coach_code	coach_name, ,gender , coach_function , country_code , country_long , disciplines, events
ATHLETE	athlete_code	name , name_short, name_tv, gender, country_code, country_long, disciplines, events
TEAM	code	Team_gender,country_code,country,country_long,disciplines, disciplines_code, events, num_athletes
TOTAL_MEDAL	medal_id	country_code, country_long, Gold_Medal, Silver_Medal, Bronze_medal, Total
EVENT	event_id	event, sport, sport_code
MEDALLIST	medallist_id, REF KEY code_atheltes, code_team,	medal_date, medal_type, medal_code, name, gender, country_code, country_long, team_gender, discipline, events, event_type, birth_date, is_medallist,
SCHEDULE	schedule_id	start_date, end_date, status, discipline, event, event_medal, phrase, gender, venue, venue_code

Relationship Sets

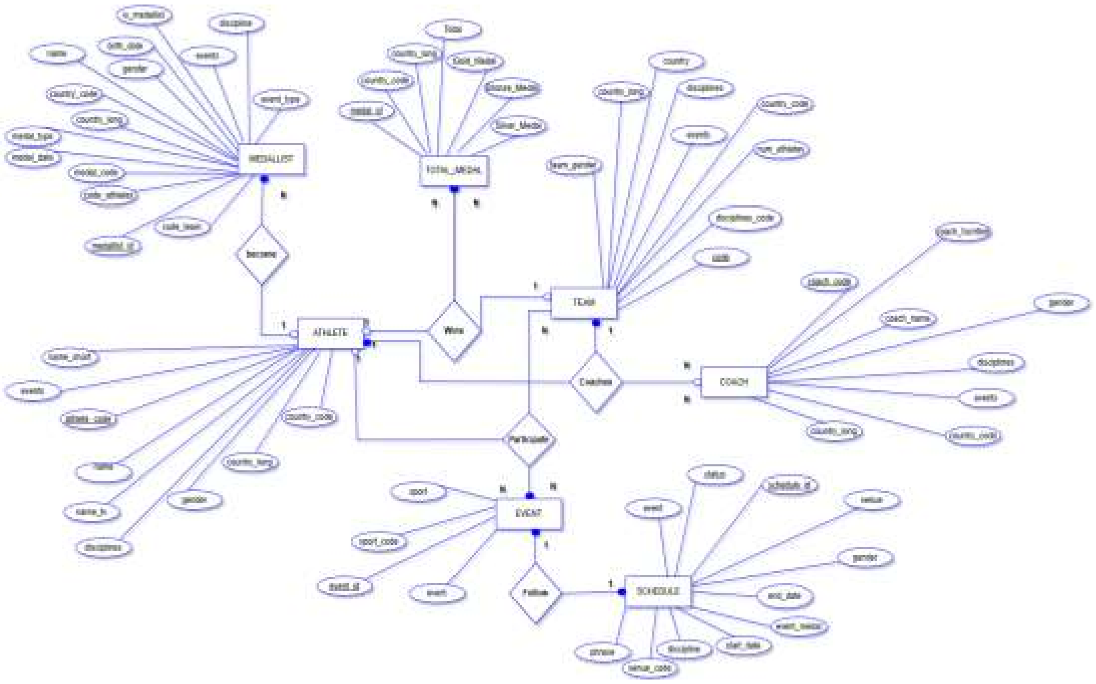
RELATIONSHIP SET	BETWEEN ENTITY SETS	ATTRIBUTES
BECOME	ATHLETE, MEDALLIST	
WINS	ATHELTE, TOTAL_MEDAL	
WINS	TEAM, TOTAL_MEDAL	
COACHES	TEAM, COACH	
COACHES	ATHLETE, COACH	
PARTICIPATE	TEAM, EVENT	
PARTICIPATE	ATHELET, EVENT	
FOLLOW	EVENT, SCHEDULE	

Constraints

Between Entity Sets	Relationship set	Cardinality	Participation
ATHLETE, MEDALLIST	become	One – Many Each athlete can win multiple medals (gold, siler, bronze) in various events. But each record refers to one athlete who won a medal.	ATHLETE - Partial MEDALLIST – Total An athlete can exit without being the medallist. But every medallist must be Athletes.
ATHELTE, TOTAL_MEDAL	wins	One – Many Each athlete can win multiple medals in various events. But each total medal record is associated with one athlete.	ATHLETE – Partial TOTAL_MEDAL - Total Not every athlete will win medals, so an athletes may exist without a medal. Every medal must be linked to an athlete who has won medals
TEAM, TOTAL_MEDAL	wins	One – Many A team can win multiple total medals in various events. But each total medal record is associated with one specific team.	TEAM – Partial, TOTAL_MEDAL - Total A team can exit without Winning any medals. Every medal must be belong to a team that won those medals.
TEAM, COACH	coaches	One -Many A team can have one coach. But a coach can coaches many team.	COACH – Partial, TEAM - Total Every team must have One coach. A coach can exit without being assigned to any team. But a team can not be exit without being trained from Coach.
COACH, ATHLETE,	coaches	One -Many	COACH – Partial, ATHLETE - Total

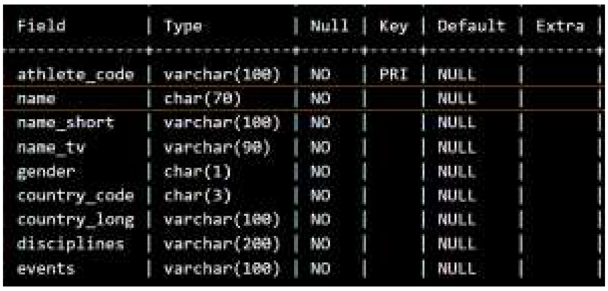
		A coach can be trains/ coaches multiple athletes. Each athlete is associated with one coach at a time.	A coach can exit without training to athletes. Every athletes must be assign to one coach for training.
TEAM, EVENT	participate	Many-Many One team can participant in multiple events But one eventscan have multiple teams	TEAM – Partial, EVENT - Partial A team can exit participating in an event An event (individual/team) can exit involving any team
ATHELET, EVENT	participate	One -Many An athlete can participate many events, But each event participation linked to Many athlete	ATHELTE – Partial EVENT- Partial Not all athletes participate in every event. An event can occur without the participation of athletes.
EVENT, SCHEDULE	follow	One- One Each event must have one schedule. Each schedule is connected to only one event	EVENT – Total SCHEDULE – Total An event can not exit without having a schedule to take place. Also, an schedule can not be exit without an events.

2.2 ER Diagram



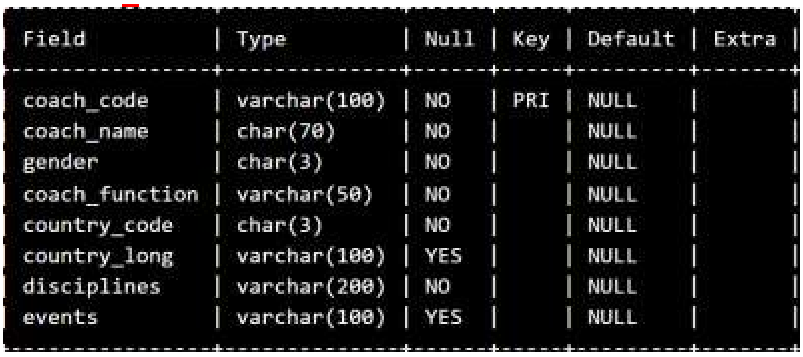
2.3 Data Description

ATHELTE _TABLE



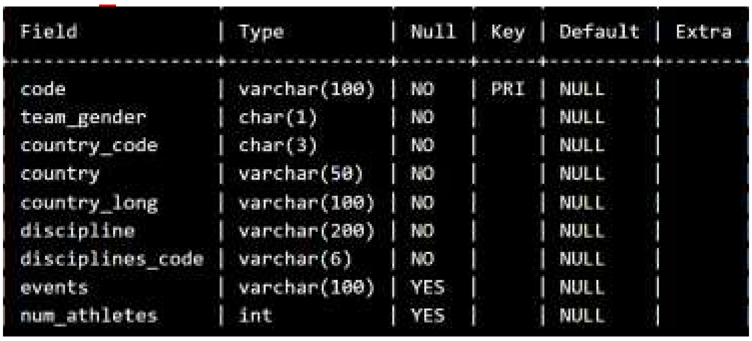
```
Field        | Type         | Null | Key | Default | Extra |
-------------+--------------+------+-----+---------+-------+
athlete_code | varchar(100) | NO   | PRI | NULL    |       |
name         | char(70)     | NO   |     | NULL    |       |
name_short   | varchar(100) | NO   |     | NULL    |       |
name_tv      | varchar(90)  | NO   |     | NULL    |       |
gender       | char(1)      | NO   |     | NULL    |       |
country_code | char(3)      | NO   |     | NULL    |       |
country_long | varchar(100) | NO   |     | NULL    |       |
disciplines  | varchar(200) | NO   |     | NULL    |       |
events       | varchar(100) | NO   |     | NULL    |       |
```

COACH_TABLE



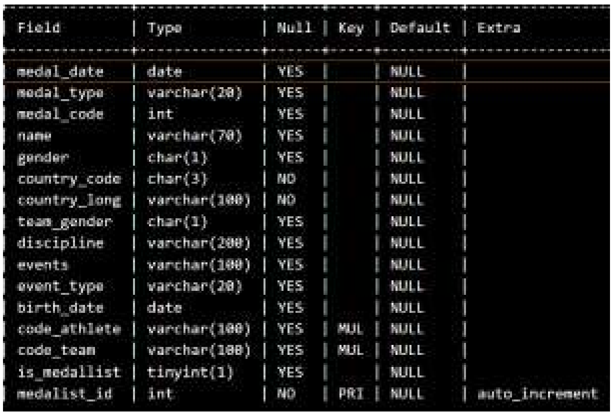
```
+----------------+--------------+------+-----+---------+-------+
| Field          | Type         | Null | Key | Default | Extra |
+----------------+--------------+------+-----+---------+-------+
| coach_code     | varchar(100) | NO   | PRI | NULL    |       |
| coach_name     | char(70)     | NO   |     | NULL    |       |
| gender         | char(3)      | NO   |     | NULL    |       |
| coach_function | varchar(50)  | NO   |     | NULL    |       |
| country_code   | char(3)      | NO   |     | NULL    |       |
| country_long   | varchar(100) | YES  |     | NULL    |       |
| disciplines    | varchar(200) | NO   |     | NULL    |       |
| events         | varchar(100) | YES  |     | NULL    |       |
+----------------+--------------+------+-----+---------+-------+
```

TEAM_TABLE



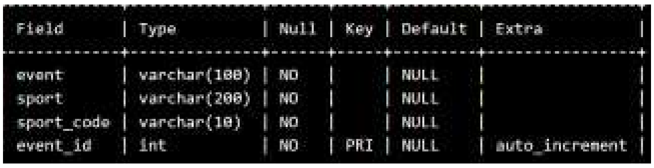
```
| Field            | Type         | Null | Key | Default | Extra
+------------------+--------------+------+-----+---------+-------
| code             | varchar(100) | NO   | PRI | NULL    |
| team_gender      | char(1)      | NO   |     | NULL    |
| country_code     | char(3)      | NO   |     | NULL    |
| country          | varchar(50)  | NO   |     | NULL    |
| country_long     | varchar(100) | NO   |     | NULL    |
| discipline       | varchar(200) | NO   |     | NULL    |
| disciplines_code | varchar(6)   | NO   |     | NULL    |
| events           | varchar(100) | YES  |     | NULL    |
| num_athletes     | int          | YES  |     | NULL    |
```

MEDALLIST_TABLE



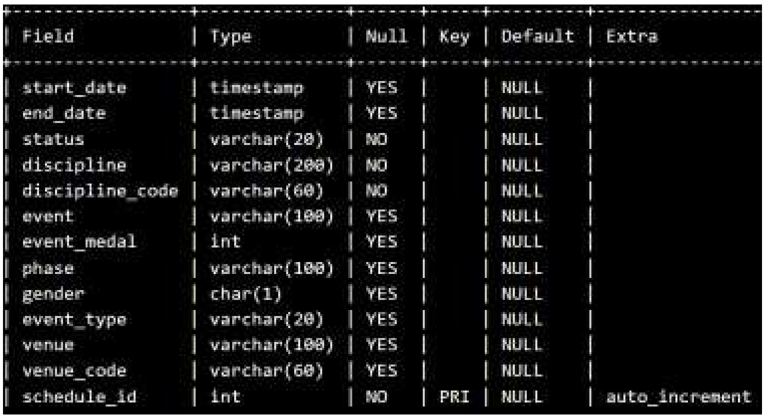
```
+--------------+--------------+------+-----+---------+----------------
| Field        | Type         | Null | Key | Default | Extra
+--------------+--------------+------+-----+---------+----------------
| medal_date   | date         | YES  |     | NULL    |
| medal_type   | varchar(20)  | YES  |     | NULL    |
| medal_code   | int          | YES  |     | NULL    |
| name         | varchar(70)  | YES  |     | NULL    |
| gender       | char(1)      | YES  |     | NULL    |
| country_code | char(3)      | NO   |     | NULL    |
| country_long | varchar(100) | NO   |     | NULL    |
| team_gender  | char(1)      | YES  |     | NULL    |
| discipline   | varchar(200) | YES  |     | NULL    |
| events       | varchar(100) | YES  |     | NULL    |
| event_type   | varchar(20)  | YES  |     | NULL    |
| birth_date   | date         | YES  |     | NULL    |
| code_athlete | varchar(100) | YES  | MUL | NULL    |
| code_team    | varchar(100) | YES  | MUL | NULL    |
| is_medallist | tinyint(1)   | YES  |     | NULL    |
| medalist_id  | int          | NO   | PRI | NULL    | auto_increment
```

EVENT_TABLE



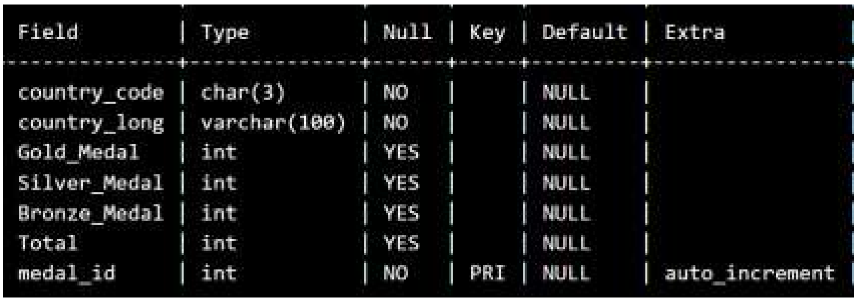
```
+------------+--------------+------+-----+---------+----------------+
| Field      | Type         | Null | Key | Default | Extra          |
+------------+--------------+------+-----+---------+----------------+
| event      | varchar(100) | NO   |     | NULL    |                |
| sport      | varchar(200) | NO   |     | NULL    |                |
| sport_code | varchar(10)  | NO   |     | NULL    |                |
| event_id   | int          | NO   | PRI | NULL    | auto_increment |
```

SCHEDULE_TABLE



```
+-----------------+--------------+------+-----+---------+----------------+
| Field           | Type         | Null | Key | Default | Extra          
+-----------------+--------------+------+-----+---------+----------------
| start_date      | timestamp    | YES  |     | NULL    |
| end_date        | timestamp    | YES  |     | NULL    |
| status          | varchar(20)  | NO   |     | NULL    |
| discipline      | varchar(200) | NO   |     | NULL    |
| discipline_code | varchar(60)  | NO   |     | NULL    |
| event           | varchar(100) | YES  |     | NULL    |
| event_medal     | int          | YES  |     | NULL    |
| phase           | varchar(100) | YES  |     | NULL    |
| gender          | char(1)      | YES  |     | NULL    |
| event_type      | varchar(20)  | YES  |     | NULL    |
| venue           | varchar(100) | YES  |     | NULL    |
| venue_code      | varchar(60)  | YES  |     | NULL    |
| schedule_id     | int          | NO   | PRI | NULL    | auto_increment
```

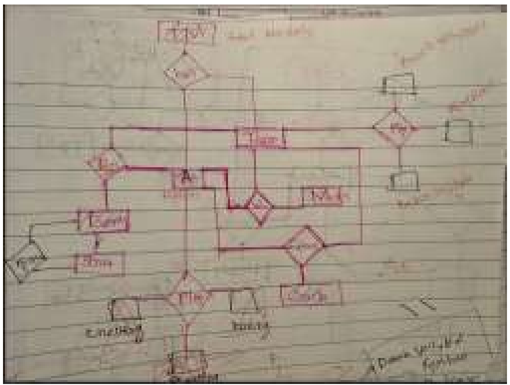
TOTAL_MEDAL TABLE

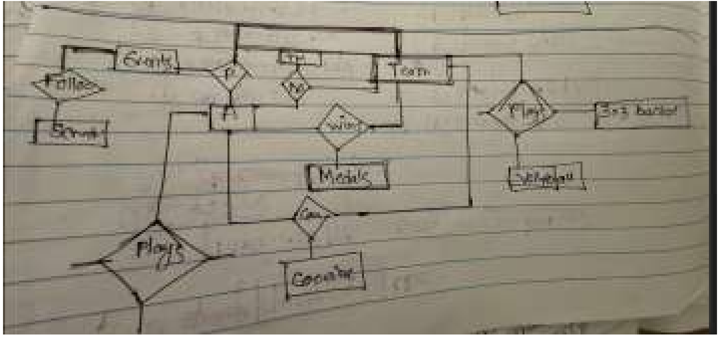


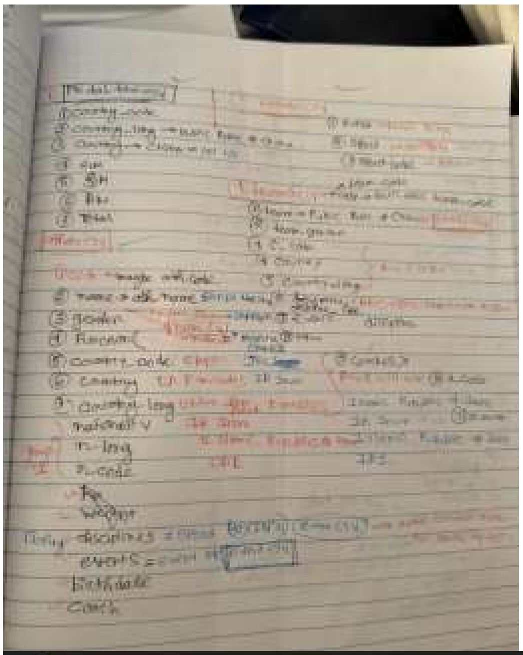
```
 Field         | Type         | Null | Key | Default | Extra          |
---------------+--------------+------+-----+---------+----------------+
 country_code  | char(3)      | NO   |     | NULL    |                |
 country_long  | varchar(100) | NO   |     | NULL    |                |
 Gold_Medal    | int          | YES  |     | NULL    |                |
 Silver_Medal  | int          | YES  |     | NULL    |                |
 Bronze_Medal  | int          | YES  |     | NULL    |                |
 Total         | int          | YES  |     | NULL    |                |
 medal_id      | int          | NO   | PRI | NULL    | auto_increment |
```

2.4 Assumptions

I did a draff from the csv file that which attribute will I use to creating the tables and try to draw the ER diagram first, after finishing the draff. I changed it so many times, to get my desire ER diagram, and attributes. Here is a sample from one of it.

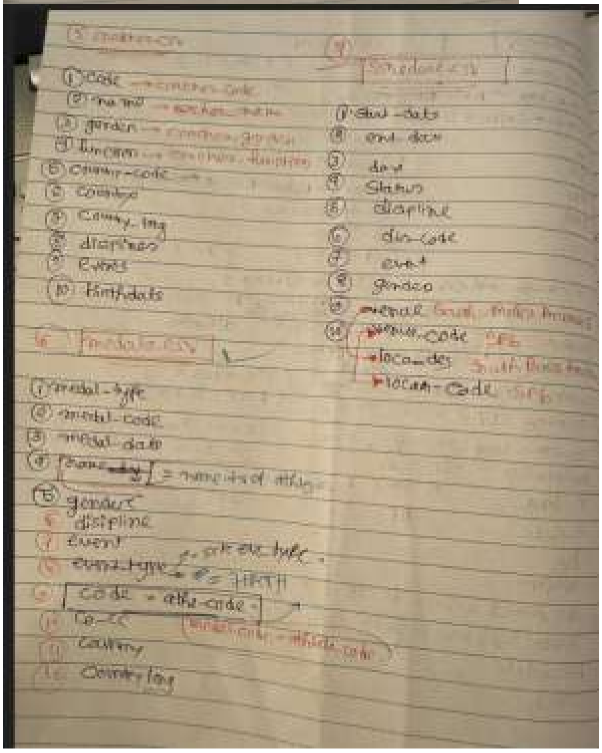






[illegible]
1. country_code
2. country_long [illegible]
3. [illegible]
4. [illegible]
5. [illegible]
6. [illegible]
7. [illegible]

[illegible]
1. [illegible]
2. name [illegible]
3. gender
4. function
5. country_code
6. country
7. country_long
nationality
n-long
n-code
disciplines [illegible]
events [illegible]
birthdate
coach



[illegible]
1. code [illegible]
2. name [illegible]
3. gender [illegible]
4. function [illegible]
5. country-code [illegible]
6. country
7. country_long
8. disciplines
9. events
10. birthdate

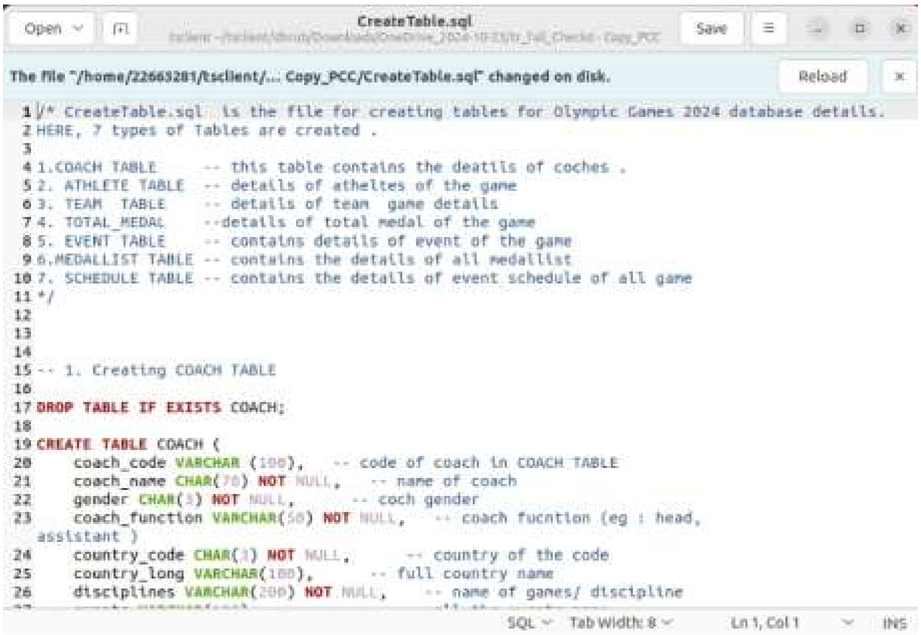
6 [illegible]
1. medal-type
2. medal-code
3. medal-date
4. ~~[illegible]~~ = [illegible]
5. gender
6. discipline
7. event
8. event-type [illegible]
9. code = athe-code
10. [illegible]
11. country
12. country-long

9 [illegible]
1. start-date
2. end-date
3. day
4. status
5. discipline
6. dis-code
7. event
8. gender
9. venue [illegible]
10. venue-code [illegible]
location-des [illegible]
location-code [illegible]

3. Implementation of the Database and Adding Sample Data

3.1 Table Creation and Constraints

The tables were created in the **CretaeTable.sql** scripts. There are constraints such as primary key foreign key, NOT NULL, NULL based on the data. Here is the file looks like



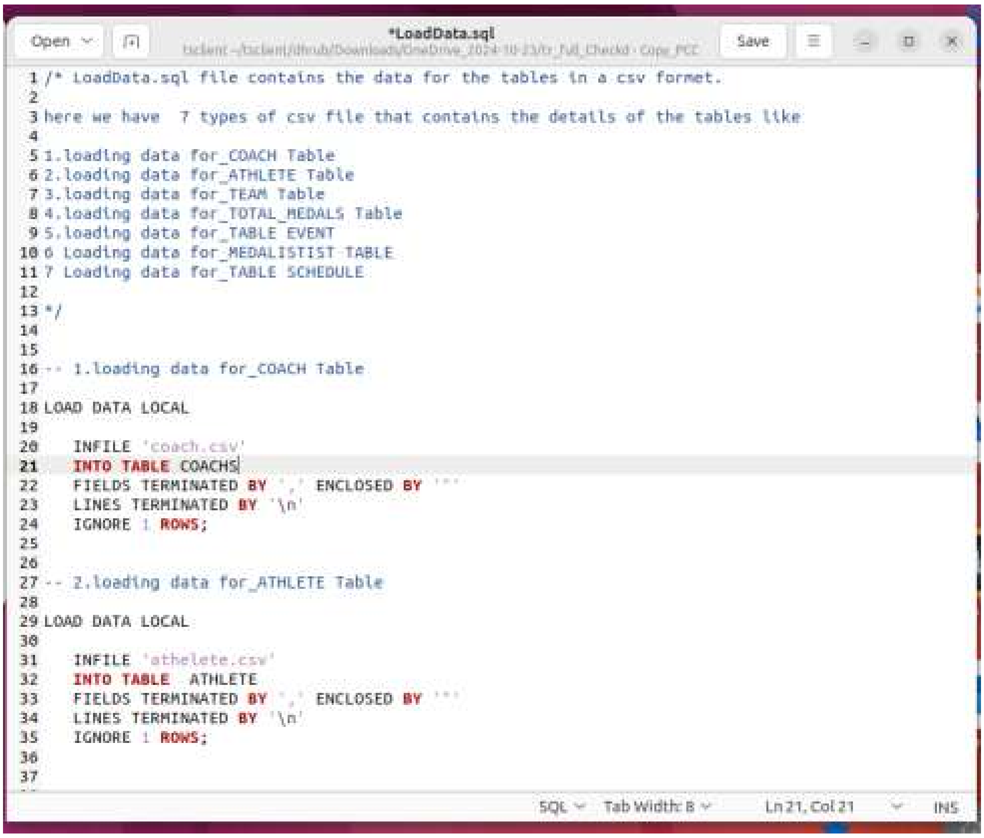
The file "/home/22663281/tsclient/... Copy_PCC/CreateTable.sql" changed on disk.

```sql
/* CreateTable.sql  is the file for creating tables for Olympic Games 2024 database details.
HERE, 7 types of Tables are created .

1.COACH TABLE      -- this table contains the deatils of coches .
2. ATHLETE TABLE  -- details of atheltes of the game
3. TEAM  TABLE     -- details of team  game details
4. TOTAL_MEDAL     --details of total medal of the game
5. EVENT TABLE     -- contains details of event of the game
6.MEDALLIST TABLE -- contains the details of all medallist
7. SCHEDULE TABLE -- contains the details of event schedule of all game
*/



-- 1. Creating COACH TABLE

DROP TABLE IF EXISTS COACH;

CREATE TABLE COACH (
    coach_code VARCHAR (100),    -- code of coach in COACH TABLE
    coach_name CHAR(70) NOT NULL,    -- name of coach
    gender CHAR(1) NOT NULL,        -- coch gender
    coach_function VARCHAR(50) NOT NULL,    -- coach fucntion (eg : head, assistant )
    country_code CHAR(3) NOT NULL,      -- country of the code
    country_long VARCHAR(100),      -- full country name
    disciplines VARCHAR(200) NOT NULL,    -- name of games/ discipline
```

3.2 Loading Data

After creating the tables to load the data , there is a file named ***LoadData.sql*** *which have all the data of csv files to loading data into tables.*

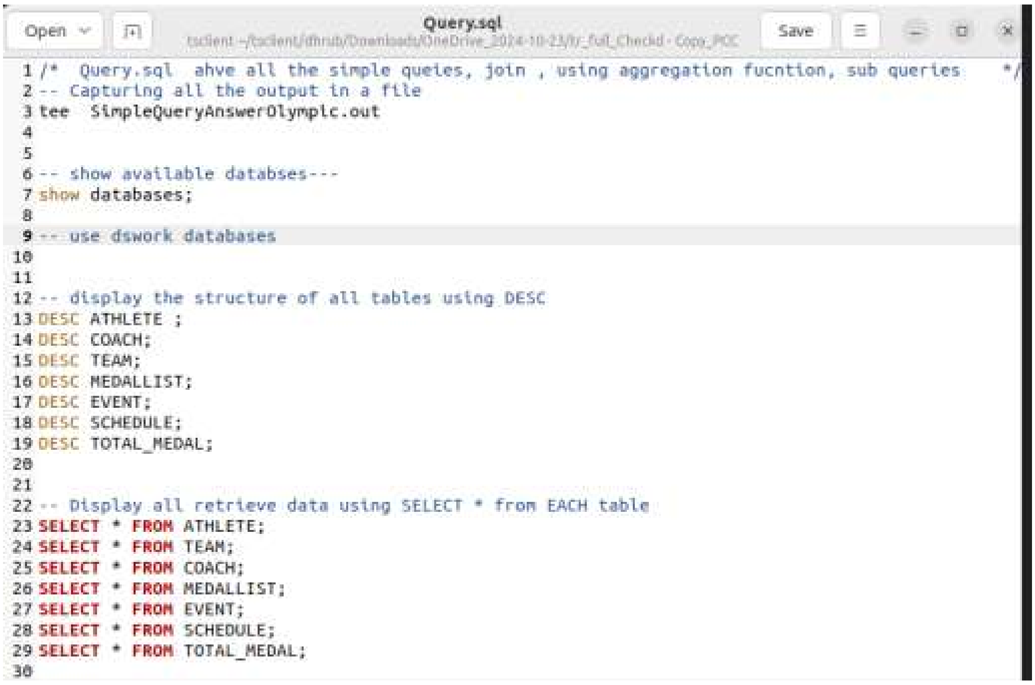
Here is ***the LoadData.sql*** *looks like:*



```sql
/* LoadData.sql file contains the data for the tables in a csv formet.

here we have  7 types of csv file that contains the details of the tables like

1.loading data for_COACH Table
2.loading data for_ATHLETE Table
3.loading data for_TEAM Table
4.loading data for_TOTAL_MEDALS Table
5.loading data for_TABLE EVENT
6 Loading data for_MEDALISTIST TABLE
7 Loading data for_TABLE SCHEDULE

*/


-- 1.loading data for_COACH Table

LOAD DATA LOCAL

    INFILE 'coach.csv'
    INTO TABLE COACHS
    FIELDS TERMINATED BY ',' ENCLOSED BY '"'
    LINES TERMINATED BY '\n'
    IGNORE 1 ROWS;


-- 2.loading data for_ATHLETE Table

LOAD DATA LOCAL

    INFILE 'athelete.csv'
    INTO TABLE  ATHLETE
    FIELDS TERMINATED BY ',' ENCLOSED BY '"'
    LINES TERMINATED BY '\n'
    IGNORE 1 ROWS;
```

3.3 File Descriptions

Each file has the explanation about the file description. Like **in Query.sql** have the description of about the file. Like this SQL file is about creating queries using *join, subqueries, aggregation function* and so on.

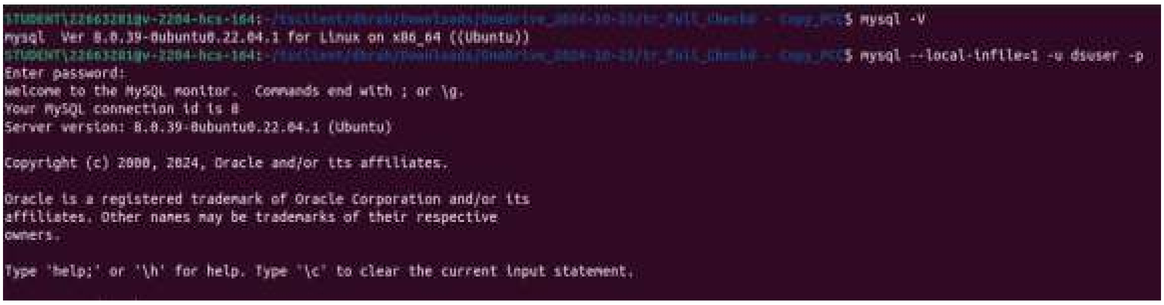


Open
Query.sql
tsclient ~/tsclient/dhrub/Downloads/OneDrive_2024-10-23/tr_full_Checkd - Copy_PCC
Save

```
/*  Query.sql  ahve all the simple queies, join , using aggregation fucntion, sub queries    */
-- Capturing all the output in a file
tee  SimpleQueryAnswerOlympic.out


-- show available databses---
show databases;

-- use dswork databases


-- display the structure of all tables using DESC
DESC ATHLETE ;
DESC COACH;
DESC TEAM;
DESC MEDALLIST;
DESC EVENT;
DESC SCHEDULE;
DESC TOTAL_MEDAL;


-- Display all retrieve data using SELECT * from EACH table
SELECT * FROM ATHLETE;
SELECT * FROM TEAM;
SELECT * FROM COACH;
SELECT * FROM MEDALLIST;
SELECT * FROM EVENT;
SELECT * FROM SCHEDULE;
SELECT * FROM TOTAL_MEDAL;

```

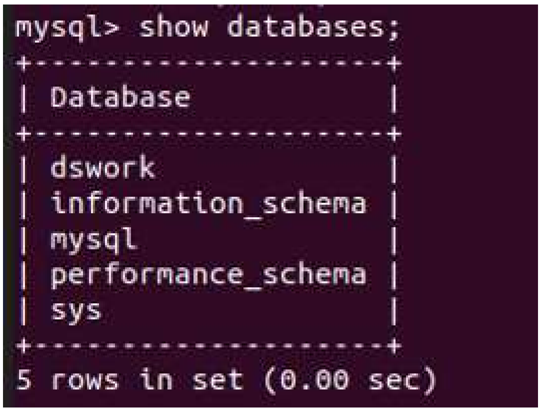
4. Use of the Database

1. *First, we need to login in VMware horizon client like this*



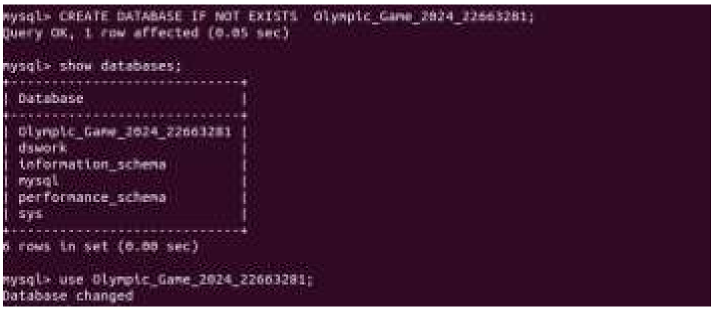
```
STUDENT\22663201@v-2204-hcs-164:~/tsclient/dhrub/Downloads/OneDrive_2024-10-23/tr_full_Checkd - Copy_PCC$ mysql -V
mysql  Ver 8.0.39-0ubuntu0.22.04.1 for Linux on x86_64 ((Ubuntu))
STUDENT\22663201@v-2204-hcs-164:~/tsclient/dhrub/Downloads/OneDrive_2024-10-23/tr_full_Checkd - Copy_PCC$ mysql --local-infile=1 -u dsuser -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.39-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

2. *Then see the existence DATABASE like this*



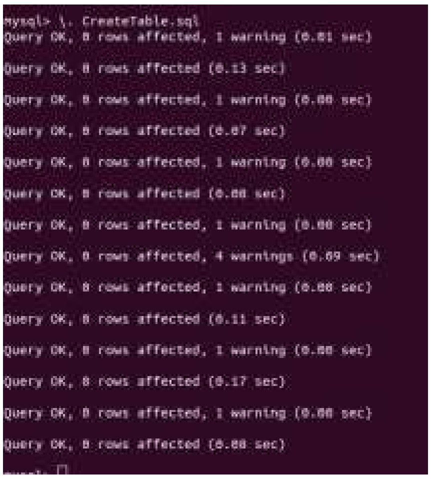
```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| dswork             |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.00 sec)
```

3. *Now need to create out own database which is;*



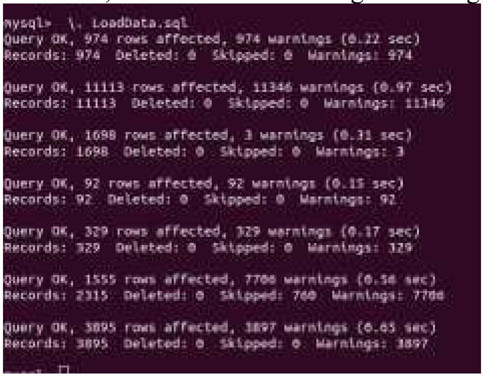
```
mysql> CREATE DATABASE IF NOT EXISTS  Olympic_Game_2024_22663281;
Query OK, 1 row affected (0.05 sec)

mysql> show databases;
+----------------------------+
| Database                   |
+----------------------------+
| Olympic_Game_2024_22663281 |
| dswork                     |
| information_schema         |
| mysql                      |
| performance_schema         |
| sys                        |
+----------------------------+
6 rows in set (0.00 sec)

mysql> use Olympic_Game_2024_22663281;
Database changed
```

4. Now create all the tables using **Createtable.sql**



```
mysql> \. CreateTable.sql
Query OK, 0 rows affected, 1 warning (0.01 sec)

Query OK, 0 rows affected (0.13 sec)

Query OK, 0 rows affected, 1 warning (0.00 sec)

Query OK, 0 rows affected (0.07 sec)

Query OK, 0 rows affected, 1 warning (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected, 1 warning (0.00 sec)

Query OK, 0 rows affected, 4 warnings (0.69 sec)

Query OK, 0 rows affected, 1 warning (0.00 sec)

Query OK, 0 rows affected (0.11 sec)

Query OK, 0 rows affected, 1 warning (0.00 sec)

Query OK, 0 rows affected (0.17 sec)

Query OK, 0 rows affected, 1 warning (0.00 sec)

Query OK, 0 rows affected (0.00 sec)
```

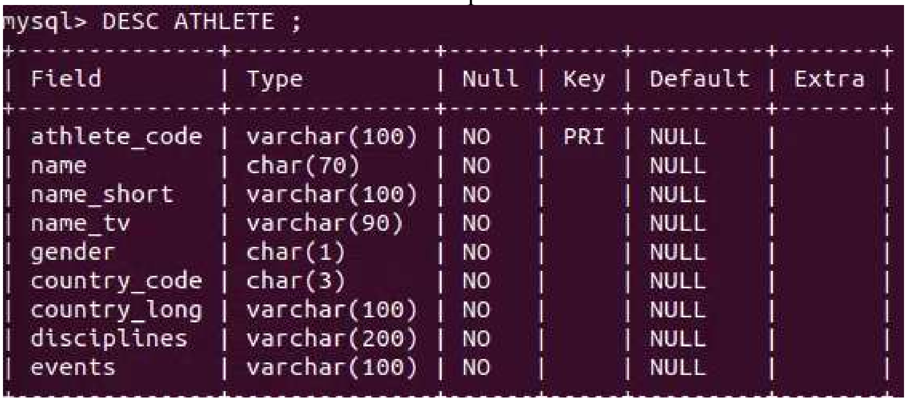
5. After this, now its time to loading data using **LoadData.sql** file



```
mysql>  \. LoadData.sql
Query OK, 974 rows affected, 974 warnings (0.22 sec)
Records: 974  Deleted: 0  Skipped: 0  Warnings: 974

Query OK, 11113 rows affected, 11346 warnings (0.97 sec)
Records: 11113  Deleted: 0  Skipped: 0  Warnings: 11346

Query OK, 1698 rows affected, 3 warnings (0.31 sec)
Records: 1698  Deleted: 0  Skipped: 0  Warnings: 3

Query OK, 92 rows affected, 92 warnings (0.15 sec)
Records: 92  Deleted: 0  Skipped: 0  Warnings: 92

Query OK, 329 rows affected, 329 warnings (0.17 sec)
Records: 329  Deleted: 0  Skipped: 0  Warnings: 329

Query OK, 1555 rows affected, 7706 warnings (0.56 sec)
Records: 2315  Deleted: 0  Skipped: 760  Warnings: 7706

Query OK, 3895 rows affected, 3897 warnings (0.65 sec)
Records: 3895  Deleted: 0  Skipped: 0  Warnings: 3897
```

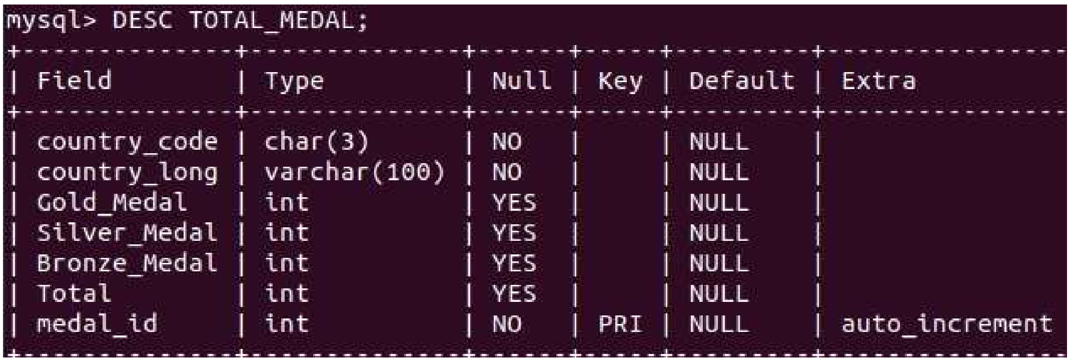
6. In this step, we will try some queries using **Queries.sql** file. As **Queries.sql** file contains almost 20 quesries. If I source the file it will show me all result together. So here, I am trying to attempt every query one to one.

This shows me the **ATHELTE TABLE** description. I have 7 tables like this.



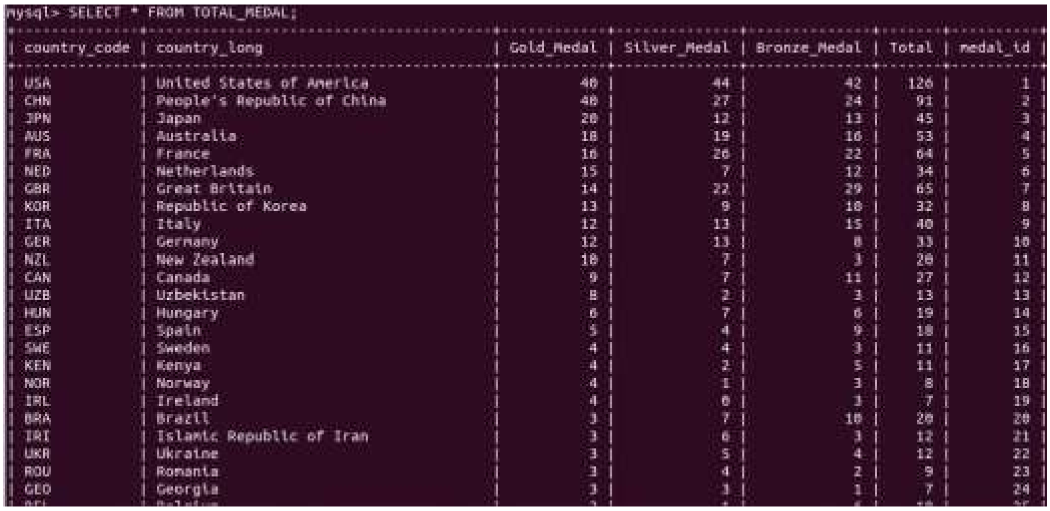
```
mysql> DESC ATHLETE ;
+---------------+--------------+------+-----+---------+-------+
| Field         | Type         | Null | Key | Default | Extra |
+---------------+--------------+------+-----+---------+-------+
| athlete_code  | varchar(100) | NO   | PRI | NULL    |       |
| name          | char(70)     | NO   |     | NULL    |       |
| name_short    | varchar(100) | NO   |     | NULL    |       |
| name_tv       | varchar(90)  | NO   |     | NULL    |       |
| gender        | char(1)      | NO   |     | NULL    |       |
| country_code  | char(3)      | NO   |     | NULL    |       |
| country_long  | varchar(100) | NO   |     | NULL    |       |
| disciplines   | varchar(200) | NO   |     | NULL    |       |
| events        | varchar(100) | NO   |     | NULL    |       |
+---------------+--------------+------+-----+---------+-------+
```

7. This one shows the TOTAL_**MEDAL TABLE** Description.



```
mysql> DESC TOTAL_MEDAL;
+---------------+--------------+------+-----+---------+----------------
| Field         | Type         | Null | Key | Default | Extra
+---------------+--------------+------+-----+---------+----------------
| country_code  | char(3)      | NO   |     | NULL    |
| country_long  | varchar(100) | NO   |     | NULL    |
| Gold_Medal    | int          | YES  |     | NULL    |
| Silver_Medal  | int          | YES  |     | NULL    |
| Bronze_Medal  | int          | YES  |     | NULL    |
| Total         | int          | YES  |     | NULL    |
| medal_id      | int          | NO   | PRI | NULL    | auto_increment
+---------------+--------------+------+-----+---------+----------------
```

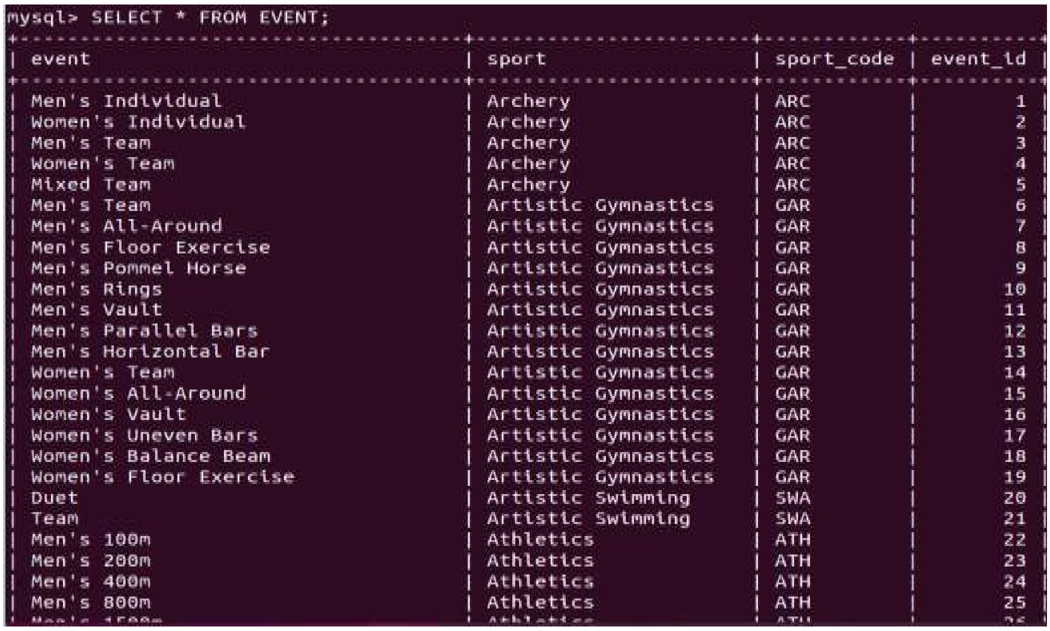
8. Now I am trying to retrieve all the data from **TOTAL_MEDAL** TABLE Like this:



```
mysql> SELECT * FROM TOTAL_MEDAL;
+--------------+------------------------------------------+------------+--------------+--------------+-------+----------+
| country_code | country_long                             | Gold_Medal | Silver_Medal | Bronze_Medal | Total | medal_id |
+--------------+------------------------------------------+------------+--------------+--------------+-------+----------+
| USA          | United States of America                 |         40 |           44 |           42 |   126 |        1 |
| CHN          | People's Republic of China               |         40 |           27 |           24 |    91 |        2 |
| JPN          | Japan                                    |         20 |           12 |           13 |    45 |        3 |
| AUS          | Australia                                |         18 |           19 |           16 |    53 |        4 |
| FRA          | France                                   |         16 |           26 |           22 |    64 |        5 |
| NED          | Netherlands                              |         15 |            7 |           12 |    34 |        6 |
| GBR          | Great Britain                            |         14 |           22 |           29 |    65 |        7 |
| KOR          | Republic of Korea                        |         13 |            9 |           10 |    32 |        8 |
| ITA          | Italy                                    |         12 |           13 |           15 |    40 |        9 |
| GER          | Germany                                  |         12 |           13 |            8 |    33 |       10 |
| NZL          | New Zealand                              |         10 |            7 |            3 |    20 |       11 |
| CAN          | Canada                                   |          9 |            7 |           11 |    27 |       12 |
| UZB          | Uzbekistan                               |          8 |            2 |            3 |    13 |       13 |
| HUN          | Hungary                                  |          6 |            7 |            6 |    19 |       14 |
| ESP          | Spain                                    |          5 |            4 |            9 |    18 |       15 |
| SWE          | Sweden                                   |          4 |            4 |            3 |    11 |       16 |
| KEN          | Kenya                                    |          4 |            2 |            5 |    11 |       17 |
| NOR          | Norway                                   |          4 |            1 |            3 |     8 |       18 |
| IRL          | Ireland                                  |          4 |            0 |            3 |     7 |       19 |
| BRA          | Brazil                                   |          3 |            7 |           10 |    20 |       20 |
| IRI          | Islamic Republic of Iran                 |          3 |            6 |            3 |    12 |       21 |
| UKR          | Ukraine                                  |          3 |            5 |            4 |    12 |       22 |
| ROU          | Romania                                  |          3 |            4 |            2 |     9 |       23 |
| GEO          | Georgia                                  |          3 |            3 |            1 |     7 |       24 |
```

It has 92 Rows .

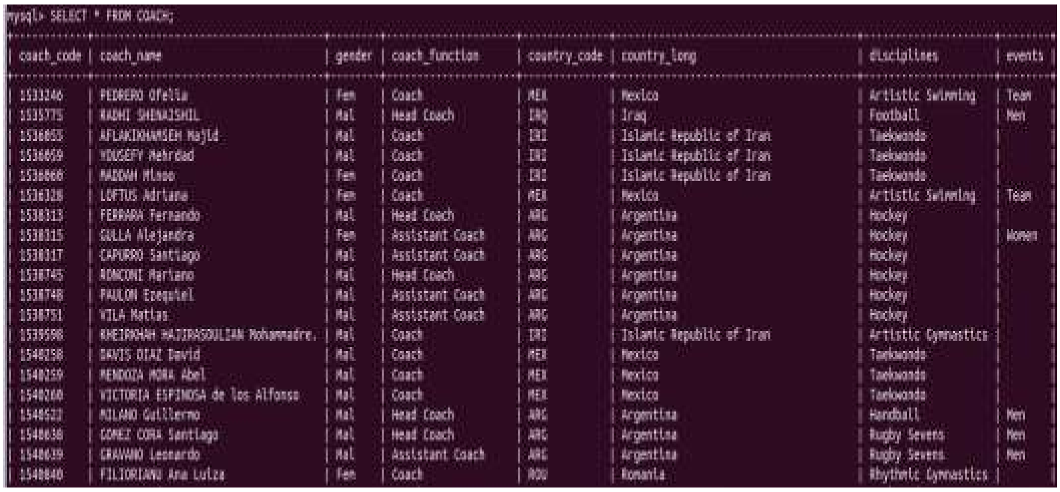
Now I am trying to retrieve all the data from **EVENT TABLE** Like this:



```
mysql> SELECT * FROM EVENT;
+----------------------------------------+------------------------+-------------+----------+
| event                                  | sport                  | sport_code  | event_id |
+----------------------------------------+------------------------+-------------+----------+
| Men's Individual                       | Archery                | ARC         |        1 |
| Women's Individual                     | Archery                | ARC         |        2 |
| Men's Team                             | Archery                | ARC         |        3 |
| Women's Team                           | Archery                | ARC         |        4 |
| Mixed Team                             | Archery                | ARC         |        5 |
| Men's Team                             | Artistic Gymnastics    | GAR         |        6 |
| Men's All-Around                       | Artistic Gymnastics    | GAR         |        7 |
| Men's Floor Exercise                   | Artistic Gymnastics    | GAR         |        8 |
| Men's Pommel Horse                     | Artistic Gymnastics    | GAR         |        9 |
| Men's Rings                            | Artistic Gymnastics    | GAR         |       10 |
| Men's Vault                            | Artistic Gymnastics    | GAR         |       11 |
| Men's Parallel Bars                    | Artistic Gymnastics    | GAR         |       12 |
| Men's Horizontal Bar                   | Artistic Gymnastics    | GAR         |       13 |
| Women's Team                           | Artistic Gymnastics    | GAR         |       14 |
| Women's All-Around                     | Artistic Gymnastics    | GAR         |       15 |
| Women's Vault                          | Artistic Gymnastics    | GAR         |       16 |
| Women's Uneven Bars                    | Artistic Gymnastics    | GAR         |       17 |
| Women's Balance Beam                   | Artistic Gymnastics    | GAR         |       18 |
| Women's Floor Exercise                 | Artistic Gymnastics    | GAR         |       19 |
| Duet                                   | Artistic Swimming      | SWA         |       20 |
| Team                                   | Artistic Swimming      | SWA         |       21 |
| Men's 100m                             | Athletics              | ATH         |       22 |
| Men's 200m                             | Athletics              | ATH         |       23 |
| Men's 400m                             | Athletics              | ATH         |       24 |
| Men's 800m                             | Athletics              | ATH         |       25 |
```

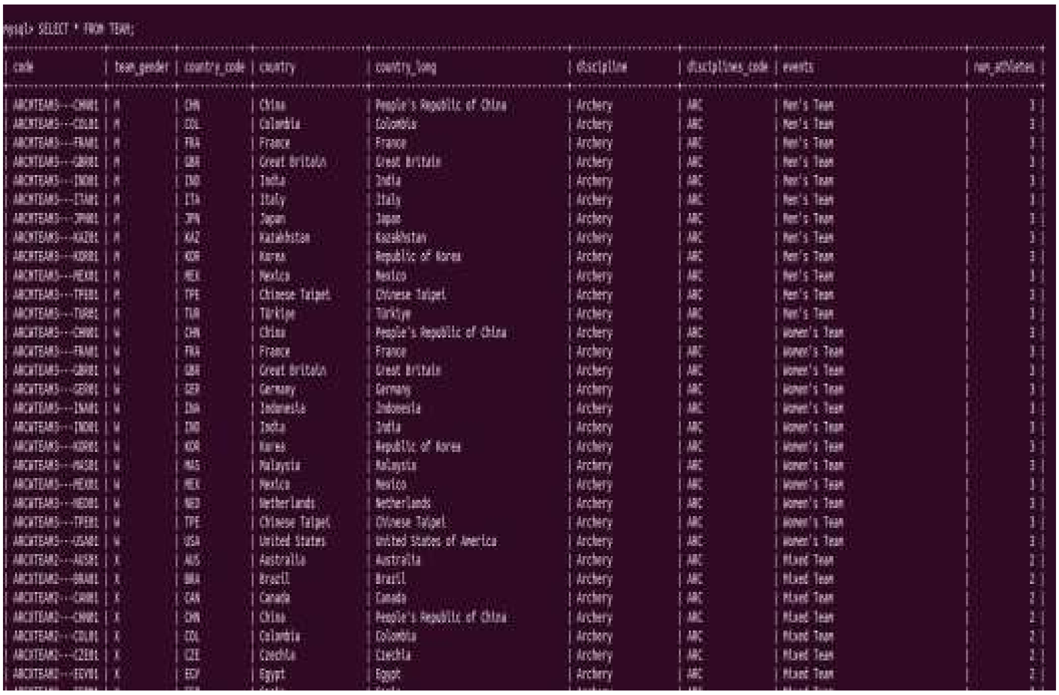
it has 329 rows

9. Now, retrieving data from **COACH TABLE** . It has rows of 974 .



```
mysql> SELECT * FROM COACH;
+------------+---------------------------------------+--------+-----------------+--------------+----------------------------+---------------------+--------+
| coach_code | coach_name                            | gender | coach_function  | country_code | country_long               | disciplines         | events |
+------------+---------------------------------------+--------+-----------------+--------------+----------------------------+---------------------+--------+
| 1533246    | PEDRERO Ofelia                        | Fem    | Coach           | MEX          | Mexico                     | Artistic Swimming   | Team   |
| 1535775    | RADHI SHENAISHIL                      | Mal    | Head Coach      | IRQ          | Iraq                       | Football            | Men    |
| 1536055    | AFLAKIKHAMSEH Majid                   | Mal    | Coach           | IRI          | Islamic Republic of Iran   | Taekwondo           |        |
| 1536059    | YOUSEFY Mehrdad                       | Mal    | Coach           | IRI          | Islamic Republic of Iran   | Taekwondo           |        |
| 1536060    | MADDAH Minoo                          | Fem    | Coach           | IRI          | Islamic Republic of Iran   | Taekwondo           |        |
| 1536328    | LOFTUS Adriana                        | Fem    | Coach           | MEX          | Mexico                     | Artistic Swimming   | Team   |
| 1538313    | FERRARA Fernando                      | Mal    | Head Coach      | ARG          | Argentina                  | Hockey              |        |
| 1538315    | GULLA Alejandra                       | Fem    | Assistant Coach | ARG          | Argentina                  | Hockey              | Women  |
| 1538317    | CAPURRO Santiago                      | Mal    | Assistant Coach | ARG          | Argentina                  | Hockey              |        |
| 1538745    | RONCONI Mariano                       | Mal    | Head Coach      | ARG          | Argentina                  | Hockey              |        |
| 1538748    | PAULON Ezequiel                       | Mal    | Assistant Coach | ARG          | Argentina                  | Hockey              |        |
| 1538751    | VILA Matias                           | Mal    | Assistant Coach | ARG          | Argentina                  | Hockey              |        |
| 1539598    | KHEIRKHAH HAJIRASOULIAN Mohammadre.   | Mal    | Coach           | IRI          | Islamic Republic of Iran   | Artistic Gymnastics |        |
| 1540258    | DAVIS DIAZ David                      | Mal    | Coach           | MEX          | Mexico                     | Taekwondo           |        |
| 1540259    | MENDOZA MORA Abel                     | Mal    | Coach           | MEX          | Mexico                     | Taekwondo           |        |
| 1540260    | VICTORIA ESPINOSA de los Alfonso      | Mal    | Coach           | MEX          | Mexico                     | Taekwondo           |        |
| 1540522    | MILANO Guillermo                      | Mal    | Head Coach      | ARG          | Argentina                  | Handball            | Men    |
| 1540638    | GOMEZ CORA Santiago                   | Mal    | Head Coach      | ARG          | Argentina                  | Rugby Sevens        | Men    |
| 1540639    | GRAVANO Leonardo                      | Mal    | Assistant Coach | ARG          | Argentina                  | Rugby Sevens        | Men    |
| 1540840    | FILIORIANU Ana Luiza                  | Fem    | Coach           | ROU          | Romania                    | Rhythmic Gymnastics |        |
```

10. Retrieving data from **TEAM TABLE**

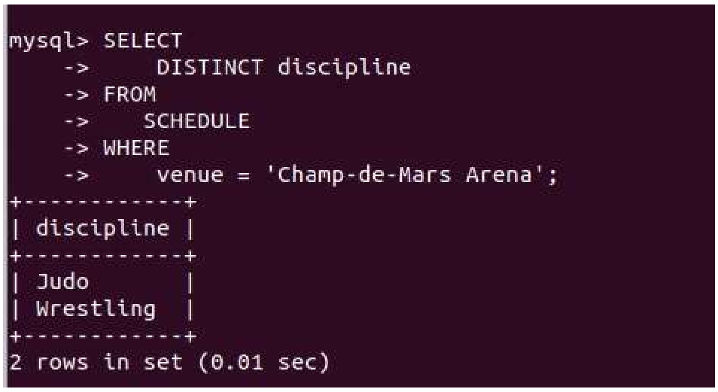


```
mysql> SELECT * FROM TEAM;
+-------------------+-------------+--------------+----------------+----------------------------+------------+------------------+---------------------------+--------------+
| code              | team_gender | country_code | country        | country_long               | discipline | disciplines_code | events                    | num_athletes |
+-------------------+-------------+--------------+----------------+----------------------------+------------+------------------+---------------------------+--------------+
| ARCMTEAM3---CHN01 | M           | CHN          | China          | People's Republic of China | Archery    | ARC              | Men's Team                |            3 |
| ARCMTEAM3---COL01 | M           | COL          | Colombia       | Colombia                   | Archery    | ARC              | Men's Team                |            3 |
| ARCMTEAM3---FRA01 | M           | FRA          | France         | France                     | Archery    | ARC              | Men's Team                |            3 |
| ARCMTEAM3---GBR01 | M           | GBR          | Great Britain  | Great Britain              | Archery    | ARC              | Men's Team                |            3 |
| ARCMTEAM3---IND01 | M           | IND          | India          | India                      | Archery    | ARC              | Men's Team                |            3 |
| ARCMTEAM3---ITA01 | M           | ITA          | Italy          | Italy                      | Archery    | ARC              | Men's Team                |            3 |
| ARCMTEAM3---JPN01 | M           | JPN          | Japan          | Japan                      | Archery    | ARC              | Men's Team                |            3 |
| ARCMTEAM3---KAZ01 | M           | KAZ          | Kazakhstan     | Kazakhstan                 | Archery    | ARC              | Men's Team                |            3 |
| ARCMTEAM3---KOR01 | M           | KOR          | Korea          | Republic of Korea          | Archery    | ARC              | Men's Team                |            3 |
| ARCMTEAM3---MEX01 | M           | MEX          | Mexico         | Mexico                     | Archery    | ARC              | Men's Team                |            3 |
| ARCMTEAM3---TPE01 | M           | TPE          | Chinese Taipei | Chinese Taipei             | Archery    | ARC              | Men's Team                |            3 |
| ARCMTEAM3---TUR01 | M           | TUR          | Türkiye        | Türkiye                    | Archery    | ARC              | Men's Team                |            3 |
| ARCWTEAM3---CHN01 | W           | CHN          | China          | People's Republic of China | Archery    | ARC              | Women's Team              |            3 |
| ARCWTEAM3---FRA01 | W           | FRA          | France         | France                     | Archery    | ARC              | Women's Team              |            3 |
| ARCWTEAM3---GBR01 | W           | GBR          | Great Britain  | Great Britain              | Archery    | ARC              | Women's Team              |            3 |
| ARCWTEAM3---GER01 | W           | GER          | Germany        | Germany                    | Archery    | ARC              | Women's Team              |            3 |
| ARCWTEAM3---INA01 | W           | INA          | Indonesia      | Indonesia                  | Archery    | ARC              | Women's Team              |            3 |
| ARCWTEAM3---IND01 | W           | IND          | India          | India                      | Archery    | ARC              | Women's Team              |            3 |
| ARCWTEAM3---KOR01 | W           | KOR          | Korea          | Republic of Korea          | Archery    | ARC              | Women's Team              |            3 |
| ARCWTEAM3---MAS01 | W           | MAS          | Malaysia       | Malaysia                   | Archery    | ARC              | Women's Team              |            3 |
| ARCWTEAM3---MEX01 | W           | MEX          | Mexico         | Mexico                     | Archery    | ARC              | Women's Team              |            3 |
| ARCWTEAM3---NED01 | W           | NED          | Netherlands    | Netherlands                | Archery    | ARC              | Women's Team              |            3 |
| ARCWTEAM3---TPE01 | W           | TPE          | Chinese Taipei | Chinese Taipei             | Archery    | ARC              | Women's Team              |            3 |
| ARCWTEAM3---USA01 | W           | USA          | United States  | United States of America   | Archery    | ARC              | Women's Team              |            3 |
| ARCXTEAM2---AUS01 | X           | AUS          | Australia      | Australia                  | Archery    | ARC              | Mixed Team                |            2 |
| ARCXTEAM2---BRA01 | X           | BRA          | Brazil         | Brazil                     | Archery    | ARC              | Mixed Team                |            2 |
| ARCXTEAM2---CAN01 | X           | CAN          | Canada         | Canada                     | Archery    | ARC              | Mixed Team                |            2 |
| ARCXTEAM2---CHN01 | X           | CHN          | China          | People's Republic of China | Archery    | ARC              | Mixed Team                |            2 |
| ARCXTEAM2---COL01 | X           | COL          | Colombia       | Colombia                   | Archery    | ARC              | Mixed Team                |            2 |
| ARCXTEAM2---CZE01 | X           | CZE          | Czechia        | Czechia                    | Archery    | ARC              | Mixed Team                |            2 |
| ARCXTEAM2---EGY01 | X           | EGY          | Egypt          | Egypt                      | Archery    | ARC              | Mixed Team                |            2 |
```

It has rows of 1698.

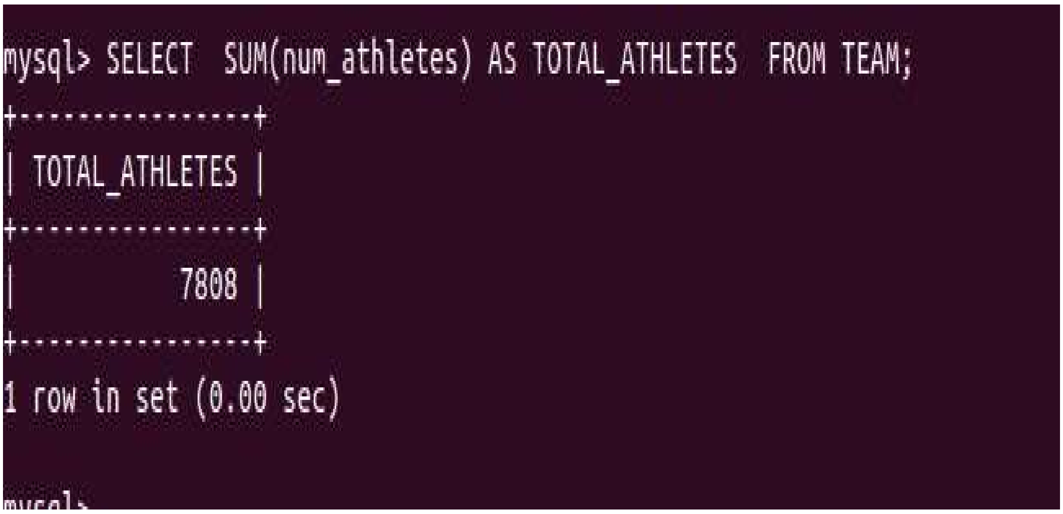
5. Implement Queries

1. By this queries we can get the discipline name which has occur venue name **‘ champ-de-Mars arena’**



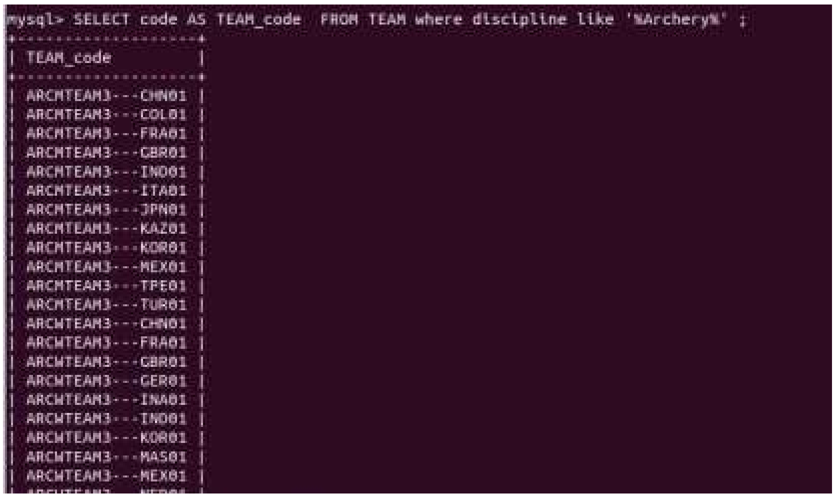
```
mysql> SELECT
    ->     DISTINCT discipline
    -> FROM
    ->    SCHEDULE
    -> WHERE
    ->     venue = 'Champ-de-Mars Arena';
+------------+
| discipline |
+------------+
| Judo       |
| Wrestling  |
+------------+
2 rows in set (0.01 sec)
```

2. By this query we can get out the total athletes number and renamed the table as **TOTAL_ATHLETES FROM TEAM TABLE.**



```
mysql> SELECT  SUM(num_athletes) AS TOTAL_ATHLETES  FROM TEAM;
+----------------+
| TOTAL_ATHLETES |
+----------------+
|           7808 |
+----------------+
1 row in set (0.00 sec)

mysql>
```

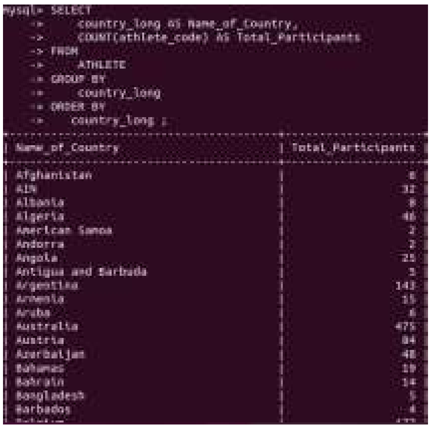
3. This query help us to get all the team code from TEAM TABLE where discipline is '%Archery%;



```
mysql> SELECT code AS TEAM_code  FROM TEAM where discipline like '%Archery%' ;
+--------------------+
| TEAM_code          |
+--------------------+
| ARCMTEAM3---CHN01  |
| ARCMTEAM3---COL01  |
| ARCMTEAM3---FRA01  |
| ARCMTEAM3---GBR01  |
| ARCMTEAM3---IND01  |
| ARCMTEAM3---ITA01  |
| ARCMTEAM3---JPN01  |
| ARCMTEAM3---KAZ01  |
| ARCMTEAM3---KOR01  |
| ARCMTEAM3---MEX01  |
| ARCMTEAM3---TPE01  |
| ARCMTEAM3---TUR01  |
| ARCWTEAM3---CHN01  |
| ARCWTEAM3---FRA01  |
| ARCWTEAM3---GBR01  |
| ARCWTEAM3---GER01  |
| ARCWTEAM3---INA01  |
| ARCWTEAM3---IND01  |
| ARCWTEAM3---KOR01  |
| ARCWTEAM3---MAS01  |
| ARCWTEAM3---MEX01  |
```

It display 57 rows.

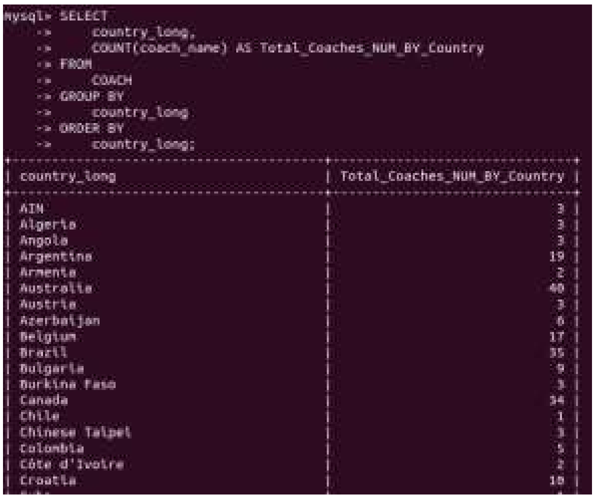
4. From ATHELTE TABLE get out the country name, count athlete based on the country. renamed the table as Total participant



```
mysql> SELECT
    ->     country_long AS Name_of_Country,
    ->     COUNT(athlete_code) AS Total_Participants
    -> FROM
    ->     ATHLETE
    -> GROUP BY
    ->     country_long
    -> ORDER BY
    ->    country_long ;
+----------------------------------------------+--------------------+
| Name_of_Country                              | Total_Participants |
+----------------------------------------------+--------------------+
| Afghanistan                                  |                  6 |
| AIN                                          |                 32 |
| Albania                                      |                  8 |
| Algeria                                      |                 46 |
| American Samoa                               |                  2 |
| Andorra                                      |                  2 |
| Angola                                       |                 25 |
| Antigua and Barbuda                          |                  3 |
| Argentina                                    |                143 |
| Armenia                                      |                 15 |
| Aruba                                        |                  6 |
| Australia                                    |                475 |
| Austria                                      |                 84 |
| Azerbaijan                                   |                 48 |
| Bahamas                                      |                 19 |
| Bahrain                                      |                 14 |
| Bangladesh                                   |                  5 |
| Barbados                                     |                  4 |
```

it displays 206 Rows

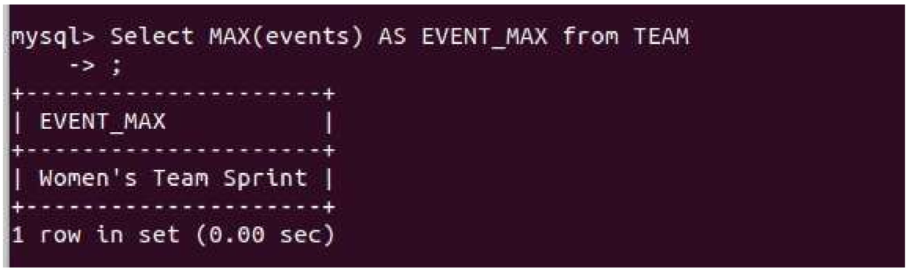
5. From **COACH TABLE** get out the country name, count coach based on the country. renamed the table as Total **_coches_num_by_country**



```
mysql> SELECT
    ->     country_long,
    ->     COUNT(coach_name) AS Total_Coaches_NUM_BY_Country
    -> FROM
    ->     COACH
    -> GROUP BY
    ->     country_long
    -> ORDER BY
    ->     country_long;
+---------------------------------------+------------------------------+
| country_long                          | Total_Coaches_NUM_BY_Country |
+---------------------------------------+------------------------------+
| AIN                                   |                            3 |
| Algeria                               |                            3 |
| Angola                                |                            3 |
| Argentina                             |                           19 |
| Armenia                               |                            2 |
| Australia                             |                           40 |
| Austria                               |                            3 |
| Azerbaijan                            |                            6 |
| Belgium                               |                           17 |
| Brazil                                |                           35 |
| Bulgaria                              |                            9 |
| Burkina Faso                          |                            3 |
| Canada                                |                           34 |
| Chile                                 |                            1 |
| Chinese Taipei                        |                            3 |
| Colombia                              |                            5 |
| Côte d'Ivoire                         |                            2 |
| Croatia                               |                           10 |
```

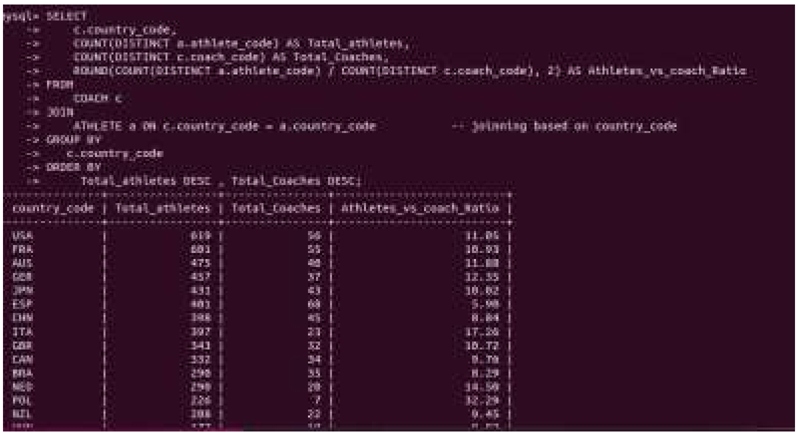
diaplays around 98 rows.

6.	This query to get out the maximum event name from **TEAM** using aggregratiom fucntion



```
mysql> Select MAX(events) AS EVENT_MAX from TEAM
    -> ;
+---------------------+
| EVENT_MAX           |
+---------------------+
| Women's Team Sprint |
+---------------------+
1 row in set (0.00 sec)
```

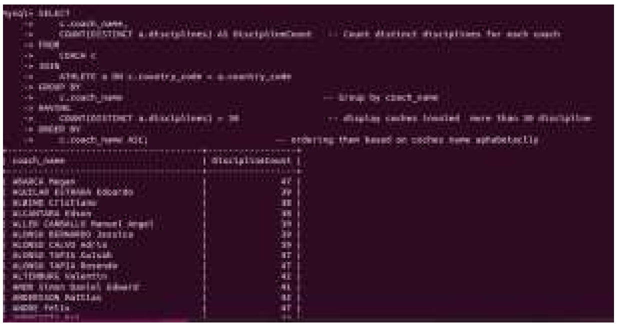
7.	Athletes and coach ratio from **ATHLTES AND CAOCH TABLE**



```
mysql> SELECT
    ->     c.country_code,
    ->     COUNT(DISTINCT a.athlete_code) AS Total_athletes,
    ->     COUNT(DISTINCT c.coach_code) AS Total_Coaches,
    ->     ROUND(COUNT(DISTINCT a.athlete_code) / COUNT(DISTINCT c.coach_code), 2) AS Athletes_vs_coach_Ratio
    -> FROM
    ->     COACH c
    -> JOIN
    ->     ATHLETE a ON c.country_code = a.country_code          -- joinning based on country_code
    -> GROUP BY
    ->    c.country_code
    -> ORDER BY
    ->      Total_athletes DESC , Total_Coaches DESC;
+--------------+----------------+---------------+-------------------------+
| country_code | Total_athletes | Total_Coaches | Athletes_vs_coach_Ratio |
+--------------+----------------+---------------+-------------------------+
| USA          |            619 |            56 |                   11.05 |
| FRA          |            601 |            55 |                   10.93 |
| AUS          |            475 |            40 |                   11.88 |
| GER          |            457 |            37 |                   12.35 |
| JPN          |            431 |            43 |                   10.02 |
| ESP          |            401 |            68 |                    5.90 |
| CHN          |            398 |            45 |                    8.84 |
| ITA          |            397 |            23 |                   17.26 |
| GBR          |            343 |            32 |                   10.72 |
| CAN          |            332 |            34 |                    9.76 |
| BRA          |            290 |            35 |                    8.29 |
| NED          |            290 |            20 |                   14.50 |
| POL          |            226 |             7 |                   32.29 |
| NZL          |            208 |            22 |                    9.45 |
```

it displays 98 rows.

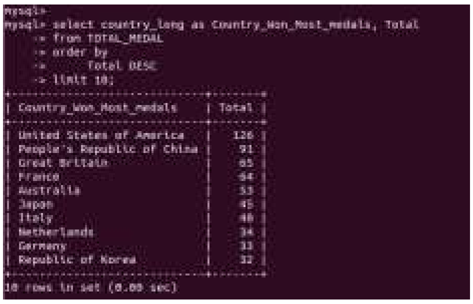
8.	List All Coaches Who Have Coached Athletes in More Than 30 Discipline



```
mysql> SELECT
    ->     c.coach_name,
    ->     COUNT(DISTINCT a.disciplines) AS DisciplineCount    -- Count distinct disciplines for each coach
    -> FROM
    ->     COACH c
    -> JOIN
    ->     ATHLETE a ON c.country_code = a.country_code
    -> GROUP BY
    ->     c.coach_name                                 -- Group by coach_name
    -> HAVING
    ->     COUNT(DISTINCT a.disciplines) > 30           -- display coaches involed  more than 30 discipline
    -> ORDER BY
    ->     c.coach_name ASC;                      -- ordering them based on coches name aphabetaclly
+-------------------------------+-----------------+
| coach_name                    | DisciplineCount |
+-------------------------------+-----------------+
| ABARCA Megan                  |              47 |
| AGUILAR ESTRADA Eduardo       |              39 |
| ALBINE Cristiano              |              38 |
| ALCANTARA Edson               |              46 |
| ALLER CARBALLO Manuel Angel   |              39 |
| ALONSO BERNARDO Jessica       |              39 |
| ALONSO CALVO Adria            |              39 |
| ALONSO TAPIA Guilab           |              47 |
| ALONSO TAPIA Rosendo          |              47 |
| ALTENBURG Valentin            |              41 |
| AMER Simon Daniel Edward      |              41 |
| ANDERSSON Mattias             |              42 |
| ANDRE Felix                   |              47 |
```

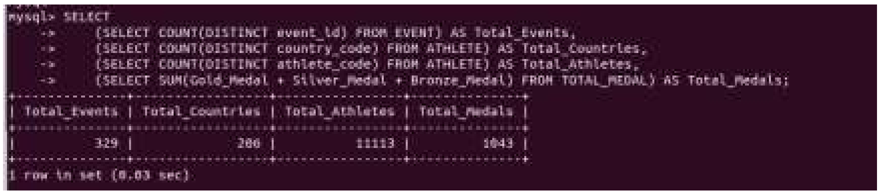
it displays 488 rows

9.	Get the most medals won by each country, using limit get the top 10



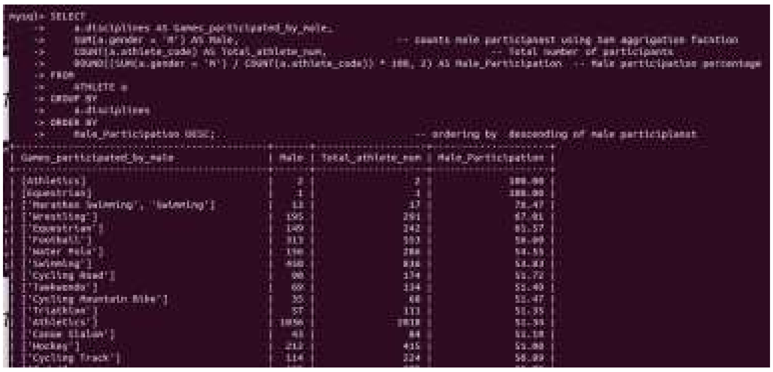
```
mysql>
mysql> select country_long as Country_Won_Most_medals, Total
    -> from TOTAL_MEDAL
    -> order by
    ->      Total DESC
    -> limit 10;
+-----------------------------+-------+
| Country_Won_Most_medals     | Total |
+-----------------------------+-------+
| United States of America    |   126 |
| People's Republic of China  |    91 |
| Great Britain               |    65 |
| France                      |    64 |
| Australia                   |    53 |
| Japan                       |    45 |
| Italy                       |    40 |
| Netherlands                 |    34 |
| Germany                     |    33 |
| Republic of Korea           |    32 |
+-----------------------------+-------+
10 rows in set (0.00 sec)
```

10. get total_events, total_atheltes, total_country and total medals select queries.



```
mysql> SELECT
    ->     (SELECT COUNT(DISTINCT event_id) FROM EVENT) AS Total_Events,
    ->     (SELECT COUNT(DISTINCT country_code) FROM ATHLETE) AS Total_Countries,
    ->     (SELECT COUNT(DISTINCT athlete_code) FROM ATHLETE) AS Total_Athletes,
    ->     (SELECT SUM(Gold_Medal + Silver_Medal + Bronze_Medal) FROM TOTAL_MEDAL) AS Total_Medals;
+--------------+-----------------+----------------+--------------+
| Total_Events | Total_Countries | Total_Athletes | Total_Medals |
+--------------+-----------------+----------------+--------------+
|          329 |             206 |          11113 |         1043 |
+--------------+-----------------+----------------+--------------+
1 row in set (0.03 sec)
```

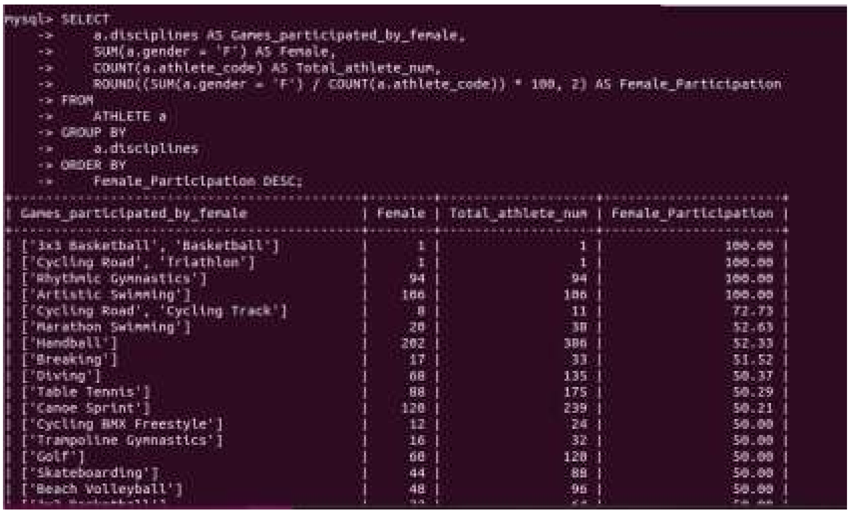
11. calculates the male participants percentages in each discipline Using **ATHLETE table**



```
mysql> SELECT
    ->     a.disciplines AS Games_participated_by_male,
    ->     SUM(a.gender = 'M') AS Male,                        -- counts male participanst using sum aggrigation facntion
    ->     COUNT(a.athlete_code) AS Total_athlete_num,               -- Total number of participants
    ->     ROUND((SUM(a.gender = 'M') / COUNT(a.athlete_code)) * 100, 2) AS Male_Participation  -- Male participation percentage
    -> FROM
    ->     ATHLETE a
    -> GROUP BY
    ->     a.disciplines
    -> ORDER BY
    ->     Male_Participation DESC;                         -- ordering by  descending of male participianst
+--------------------------------------------+------+-------------------+--------------------+
| Games_participated_by_male                 | Male | Total_athlete_num | Male_Participation |
+--------------------------------------------+------+-------------------+--------------------+
| [Athletics]                                |    2 |                 2 |             100.00 |
| [Equestrian]                               |    1 |                 1 |             100.00 |
| ['Marathon Swimming', 'Swimming']          |   13 |                17 |              76.47 |
| ['Wrestling']                              |  195 |               291 |              67.01 |
| ['Equestrian']                             |  149 |               242 |              61.57 |
| ['Football']                               |  313 |               593 |              58.00 |
| ['Water Polo']                             |  150 |               280 |              53.57 |
| ['Swimming']                               |  450 |               836 |              53.83 |
| ['Cycling Road']                           |   90 |               174 |              51.72 |
| ['Taekwondo']                              |   69 |               134 |              51.49 |
| ['Cycling Mountain Bike']                  |   35 |                68 |              51.47 |
| ['Triathlon']                              |   57 |               111 |              51.35 |
| ['Athletics']                              | 1056 |              2058 |              51.31 |
| ['Canoe Slalom']                           |   43 |                84 |              51.19 |
| ['Hockey']                                 |  212 |               415 |              51.08 |
| ['Cycling Track']                          |  114 |               224 |              50.89 |
```

it displays 52 rows.

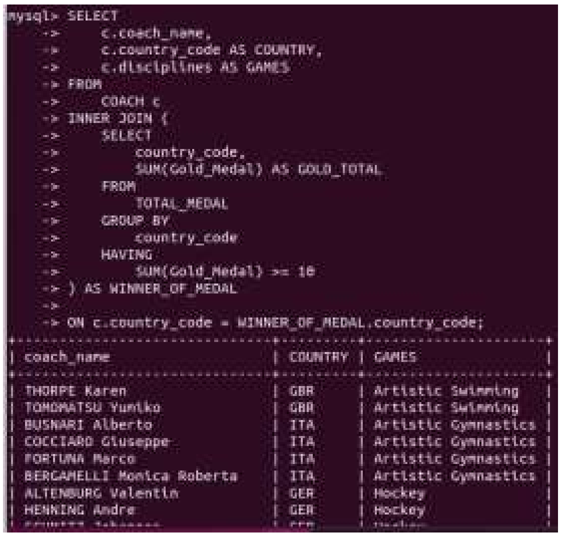
12. calculates the Female participants percentages in each discipline **Using ATHLETE table**



```
mysql> SELECT
    ->     a.disciplines AS Games_participated_by_female,
    ->     SUM(a.gender = 'F') AS Female,
    ->     COUNT(a.athlete_code) AS Total_athlete_num,
    ->     ROUND((SUM(a.gender = 'F') / COUNT(a.athlete_code)) * 100, 2) AS Female_Participation
    -> FROM
    ->     ATHLETE a
    -> GROUP BY
    ->     a.disciplines
    -> ORDER BY
    ->     Female_Participation DESC;
+-------------------------------------+--------+-------------------+----------------------+
| Games_participated_by_female        | Female | Total_athlete_num | Female_Participation |
+-------------------------------------+--------+-------------------+----------------------+
| ['3x3 Basketball', 'Basketball']    |      1 |                 1 |               100.00 |
| ['Cycling Road', 'Triathlon']       |      1 |                 1 |               100.00 |
| ['Rhythmic Gymnastics']             |     94 |                94 |               100.00 |
| ['Artistic Swimming']               |    106 |               106 |               100.00 |
| ['Cycling Road', 'Cycling Track']   |      8 |                11 |                72.73 |
| ['Marathon Swimming']               |     20 |                38 |                52.63 |
| ['Handball']                        |    202 |               386 |                52.33 |
| ['Breaking']                        |     17 |                33 |                51.52 |
| ['Diving']                          |     68 |               135 |                50.37 |
| ['Table Tennis']                    |     88 |               175 |                50.29 |
| ['Canoe Sprint']                    |    120 |               239 |                50.21 |
| ['Cycling BMX Freestyle']           |     12 |                24 |                50.00 |
| ['Trampoline Gymnastics']           |     16 |                32 |                50.00 |
| ['Golf']                            |     60 |               120 |                50.00 |
| ['Skateboarding']                   |     44 |                88 |                50.00 |
| ['Beach Volleyball']                |     48 |                96 |                50.00 |
```

it displays 52 rows.

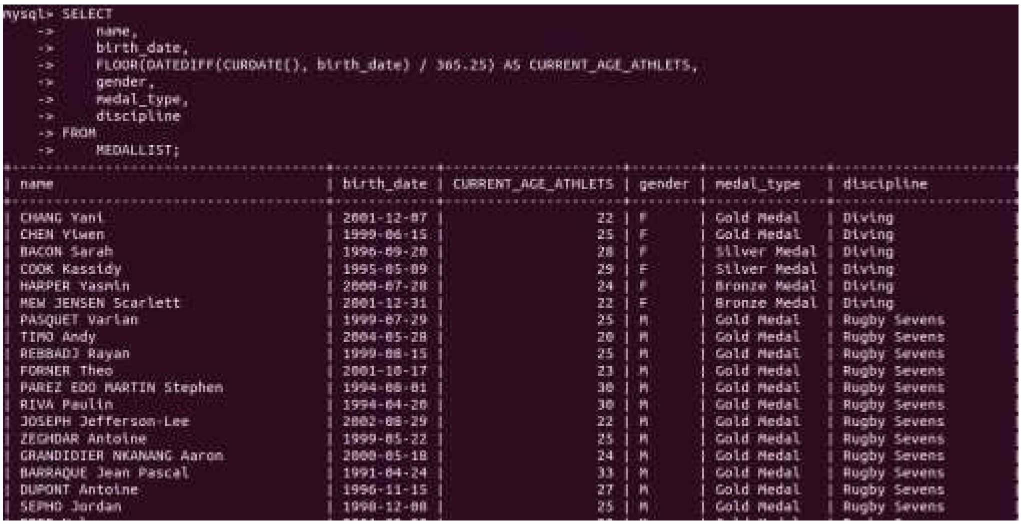
13. get the coach name ASSOCIATED WITH COUNRTY AND won least 10 gold medals , used join and sub-query



```
mysql> SELECT
    ->     c.coach_name,
    ->     c.country_code AS COUNTRY,
    ->     c.disciplines AS GAMES
    -> FROM
    ->     COACH c
    -> INNER JOIN (
    ->     SELECT
    ->         country_code,
    ->         SUM(Gold_Medal) AS GOLD_TOTAL
    ->     FROM
    ->         TOTAL_MEDAL
    ->     GROUP BY
    ->         country_code
    ->     HAVING
    ->         SUM(Gold_Medal) >= 10
    -> ) AS WINNER_OF_MEDAL
    ->
    -> ON c.country_code = WINNER_OF_MEDAL.country_code;
+-----------------------------+---------+---------------------+
| coach_name                  | COUNTRY | GAMES               |
+-----------------------------+---------+---------------------+
| THORPE Karen                | GBR     | Artistic Swimming   |
| TOMOMATSU Yumiko            | GBR     | Artistic Swimming   |
| BUSNARI Alberto             | ITA     | Artistic Gymnastics |
| COCCIARO Giuseppe           | ITA     | Artistic Gymnastics |
| FORTUNA Marco               | ITA     | Artistic Gymnastics |
| BERGAMELLI Monica Roberta   | ITA     | Artistic Gymnastics |
| ALTENBURG Valentin          | GER     | Hockey              |
| HENNING Andre               | GER     | Hockey              |
```

it displays 388 rows.

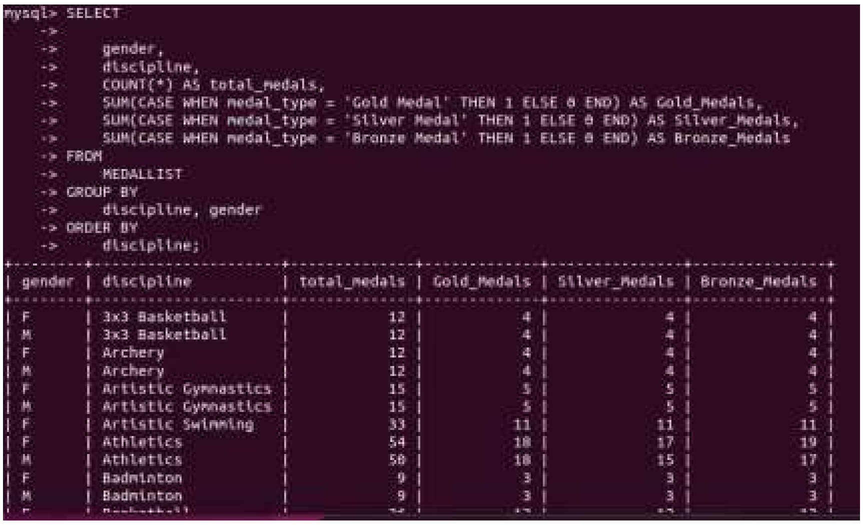
14. retrieve the current age of the athlete using DATEDIFF format and renamed the table CURRENT_AGE_ATHELETS,name, gender, medal type and discipline from MEDALLIST TABLE



```
mysql> SELECT
    ->     name,
    ->     birth_date,
    ->     FLOOR(DATEDIFF(CURDATE(), birth_date) / 365.25) AS CURRENT_AGE_ATHLETS,
    ->     gender,
    ->     medal_type,
    ->     discipline
    -> FROM
    ->     MEDALLIST;
+--------------------------------------+------------+---------------------+--------+--------------+--------------+
| name                                 | birth_date | CURRENT_AGE_ATHLETS | gender | medal_type   | discipline   |
+--------------------------------------+------------+---------------------+--------+--------------+--------------+
| CHANG Yani                           | 2001-12-07 |                  22 | F      | Gold Medal   | Diving       |
| CHEN Yiwen                           | 1999-06-15 |                  25 | F      | Gold Medal   | Diving       |
| BACON Sarah                          | 1996-09-20 |                  28 | F      | Silver Medal | Diving       |
| COOK Kassidy                         | 1995-05-09 |                  29 | F      | Silver Medal | Diving       |
| HARPER Yasmin                        | 2000-07-28 |                  24 | F      | Bronze Medal | Diving       |
| MEW JENSEN Scarlett                  | 2001-12-31 |                  22 | F      | Bronze Medal | Diving       |
| PASQUET Varian                       | 1999-07-29 |                  25 | M      | Gold Medal   | Rugby Sevens |
| TIMO Andy                            | 2004-05-28 |                  20 | M      | Gold Medal   | Rugby Sevens |
| REBBADJ Rayan                        | 1999-08-15 |                  25 | M      | Gold Medal   | Rugby Sevens |
| FORNER Theo                          | 2001-10-17 |                  23 | M      | Gold Medal   | Rugby Sevens |
| PAREZ EDO MARTIN Stephen             | 1994-08-01 |                  30 | M      | Gold Medal   | Rugby Sevens |
| RIVA Paulin                          | 1994-04-20 |                  30 | M      | Gold Medal   | Rugby Sevens |
| JOSEPH Jefferson-Lee                 | 2002-08-29 |                  22 | M      | Gold Medal   | Rugby Sevens |
| ZEGHDAR Antoine                      | 1999-05-22 |                  25 | M      | Gold Medal   | Rugby Sevens |
| GRANDIDIER NKANANG Aaron             | 2000-05-18 |                  24 | M      | Gold Medal   | Rugby Sevens |
| BARRAQUE Jean Pascal                 | 1991-04-24 |                  33 | M      | Gold Medal   | Rugby Sevens |
| DUPONT Antoine                       | 1996-11-15 |                  27 | M      | Gold Medal   | Rugby Sevens |
| SEPHO Jordan                         | 1998-12-08 |                  25 | M      | Gold Medal   | Rugby Sevens |
```

It displasy 1555 rows

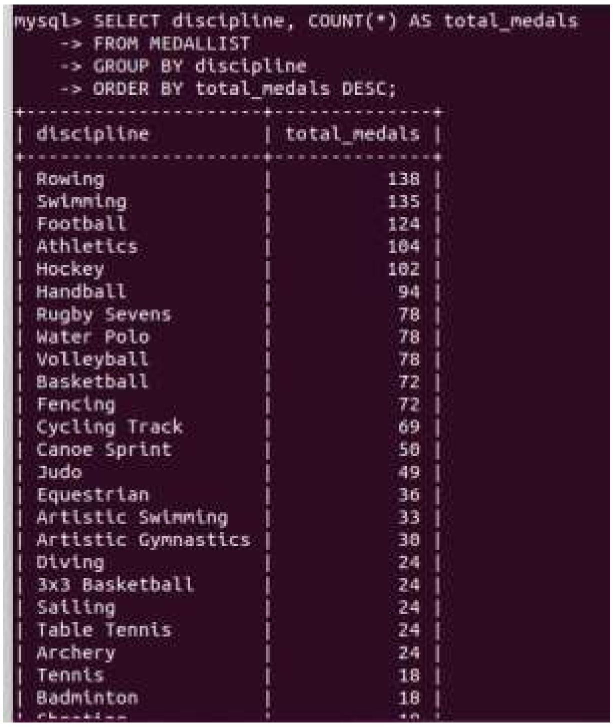
15. Compare medals between Female & Male



```
mysql> SELECT
    ->
    ->     gender,
    ->     discipline,
    ->     COUNT(*) AS total_medals,
    ->     SUM(CASE WHEN medal_type = 'Gold Medal' THEN 1 ELSE 0 END) AS Gold_Medals,
    ->     SUM(CASE WHEN medal_type = 'Silver Medal' THEN 1 ELSE 0 END) AS Silver_Medals,
    ->     SUM(CASE WHEN medal_type = 'Bronze Medal' THEN 1 ELSE 0 END) AS Bronze_Medals
    -> FROM
    ->     MEDALLIST
    -> GROUP BY
    ->     discipline, gender
    -> ORDER BY
    ->     discipline;
+--------+---------------------+--------------+-------------+---------------+---------------+
| gender | discipline          | total_medals | Gold_Medals | Silver_Medals | Bronze_Medals |
+--------+---------------------+--------------+-------------+---------------+---------------+
| F      | 3x3 Basketball      |           12 |           4 |             4 |             4 |
| M      | 3x3 Basketball      |           12 |           4 |             4 |             4 |
| F      | Archery             |           12 |           4 |             4 |             4 |
| M      | Archery             |           12 |           4 |             4 |             4 |
| F      | Artistic Gymnastics |           15 |           5 |             5 |             5 |
| M      | Artistic Gymnastics |           15 |           5 |             5 |             5 |
| F      | Artistic Swimming   |           33 |          11 |            11 |            11 |
| F      | Athletics           |           54 |          18 |            17 |            19 |
| M      | Athletics           |           50 |          18 |            15 |            17 |
| F      | Badminton           |            9 |           3 |             3 |             3 |
| M      | Badminton           |            9 |           3 |             3 |             3 |
```

it display 54 rows

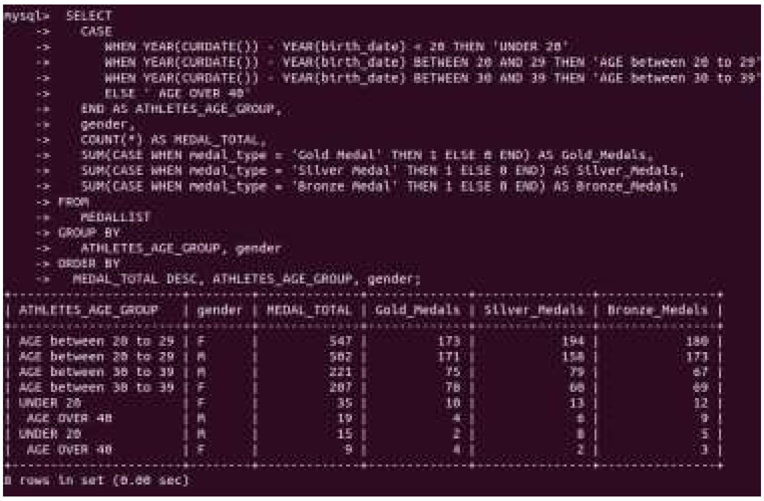
16. find out the Medals by discipline



```
mysql> SELECT discipline, COUNT(*) AS total_medals
    -> FROM MEDALLIST
    -> GROUP BY discipline
    -> ORDER BY total_medals DESC;
+---------------------+--------------+
| discipline          | total_medals |
+---------------------+--------------+
| Rowing              |          138 |
| Swimming            |          135 |
| Football            |          124 |
| Athletics           |          104 |
| Hockey              |          102 |
| Handball            |           94 |
| Rugby Sevens        |           78 |
| Water Polo          |           78 |
| Volleyball          |           78 |
| Basketball          |           72 |
| Fencing             |           72 |
| Cycling Track       |           69 |
| Canoe Sprint        |           58 |
| Judo                |           49 |
| Equestrian          |           36 |
| Artistic Swimming   |           33 |
| Artistic Gymnastics |           30 |
| Diving              |           24 |
| 3x3 Basketball      |           24 |
| Sailing             |           24 |
| Table Tennis        |           24 |
| Archery             |           24 |
| Tennis              |           18 |
| Badminton           |           18 |
```

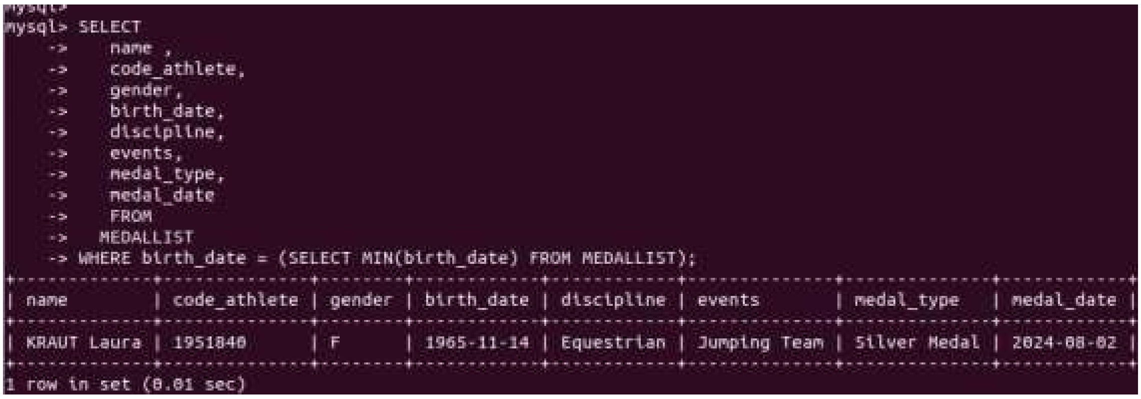
shows 28 rows

17. list down the athletes age group and compare how many medals(medal_type) they have win based on age group



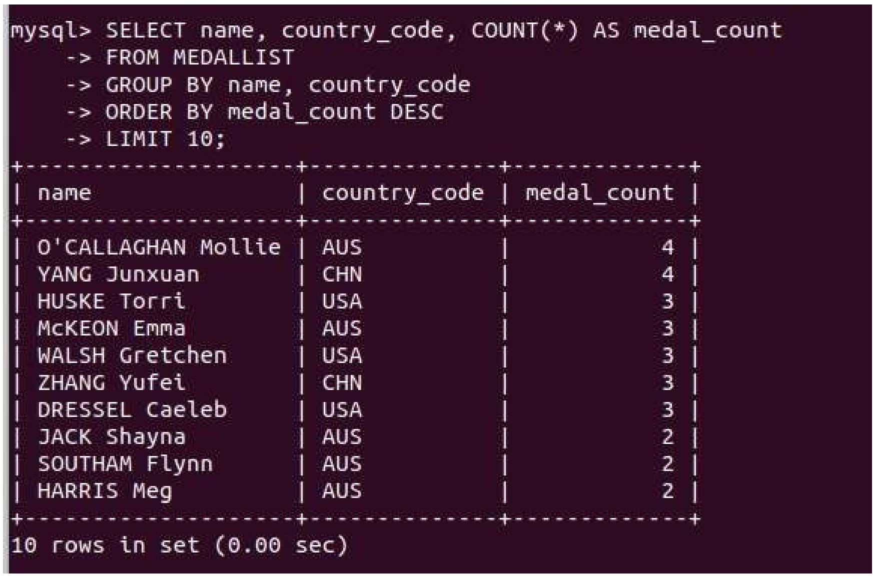
```
mysql>  SELECT
    ->    CASE
    ->       WHEN YEAR(CURDATE()) - YEAR(birth_date) < 20 THEN 'UNDER 20'
    ->       WHEN YEAR(CURDATE()) - YEAR(birth_date) BETWEEN 20 AND 29 THEN 'AGE between 20 to 29'
    ->       WHEN YEAR(CURDATE()) - YEAR(birth_date) BETWEEN 30 AND 39 THEN 'AGE between 30 to 39'
    ->       ELSE ' AGE OVER 40'
    ->    END AS ATHLETES_AGE_GROUP,
    ->    gender,
    ->    COUNT(*) AS MEDAL_TOTAL,
    ->    SUM(CASE WHEN medal_type = 'Gold Medal' THEN 1 ELSE 0 END) AS Gold_Medals,
    ->    SUM(CASE WHEN medal_type = 'Silver Medal' THEN 1 ELSE 0 END) AS Silver_Medals,
    ->    SUM(CASE WHEN medal_type = 'Bronze Medal' THEN 1 ELSE 0 END) AS Bronze_Medals
    -> FROM
    ->    MEDALLIST
    -> GROUP BY
    ->    ATHLETES_AGE_GROUP, gender
    -> ORDER BY
    ->   MEDAL_TOTAL DESC, ATHLETES_AGE_GROUP, gender;
+----------------------+--------+-------------+-------------+---------------+---------------+
| ATHLETES_AGE_GROUP   | gender | MEDAL_TOTAL | Gold_Medals | Silver_Medals | Bronze_Medals |
+----------------------+--------+-------------+-------------+---------------+---------------+
| AGE between 20 to 29 | F      |         547 |         173 |           194 |           180 |
| AGE between 20 to 29 | M      |         502 |         171 |           158 |           173 |
| AGE between 30 to 39 | M      |         221 |          75 |            79 |            67 |
| AGE between 30 to 39 | F      |         207 |          78 |            60 |            69 |
| UNDER 20             | F      |          35 |          10 |            13 |            12 |
|  AGE OVER 40         | M      |          19 |           4 |             6 |             9 |
| UNDER 20             | M      |          15 |           2 |             8 |             5 |
|  AGE OVER 40         | F      |           9 |           4 |             2 |             3 |
+----------------------+--------+-------------+-------------+---------------+---------------+
8 rows in set (0.00 sec)
```

18. find out the youngest athlete using birthdate from MEDALLIST TABLE, also name, code of the althlete, gender, discipline, events, medal_type ,



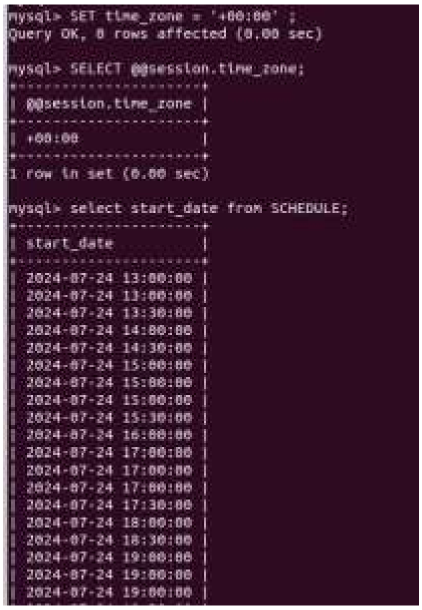
```
mysql>
mysql> SELECT
    ->    name ,
    ->    code_athlete,
    ->    gender,
    ->    birth_date,
    ->    discipline,
    ->    events,
    ->    medal_type,
    ->    medal_date
    ->    FROM
    ->   MEDALLIST
    -> WHERE birth_date = (SELECT MIN(birth_date) FROM MEDALLIST);
+-------------+--------------+--------+------------+------------+--------------+--------------+------------+
| name        | code_athlete | gender | birth_date | discipline | events       | medal_type   | medal_date |
+-------------+--------------+--------+------------+------------+--------------+--------------+------------+
| KRAUT Laura | 1951840      | F      | 1965-11-14 | Equestrian | Jumping Team | Silver Medal | 2024-08-02 |
+-------------+--------------+--------+------------+------------+--------------+--------------+------------+
1 row in set (0.01 sec)
```

19. from MEDALLIST find out top Atheltes by counting medal , use limit to get only top 10



```
mysql> SELECT name, country_code, COUNT(*) AS medal_count
    -> FROM MEDALLIST
    -> GROUP BY name, country_code
    -> ORDER BY medal_count DESC
    -> LIMIT 10;
+--------------------+--------------+-------------+
| name               | country_code | medal_count |
+--------------------+--------------+-------------+
| O'CALLAGHAN Mollie | AUS          |           4 |
| YANG Junxuan       | CHN          |           4 |
| HUSKE Torri        | USA          |           3 |
| McKEON Emma        | AUS          |           3 |
| WALSH Gretchen     | USA          |           3 |
| ZHANG Yufei        | CHN          |           3 |
| DRESSEL Caeleb     | USA          |           3 |
| JACK Shayna        | AUS          |           2 |
| SOUTHAM Flynn      | AUS          |           2 |
| HARRIS Meg         | AUS          |           2 |
+--------------------+--------------+-------------+
10 rows in set (0.00 sec)
```

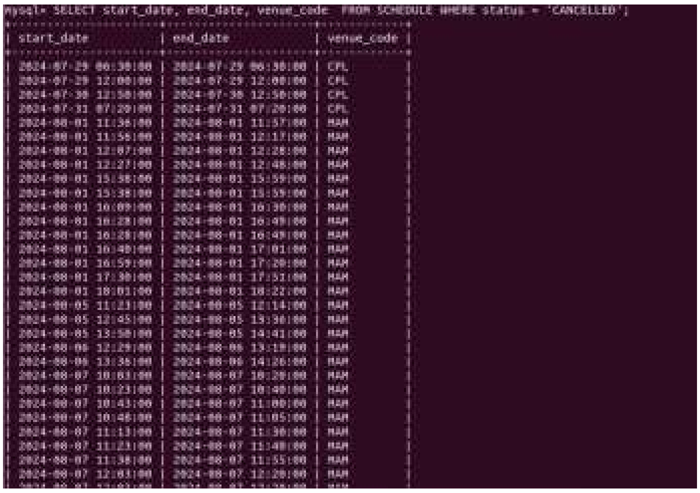
20.   from SCGEDULE TABLE get the start_date



```
mysql> SET time_zone = '+00:00' ;
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT @@session.time_zone;
+---------------------+
| @@session.time_zone |
+---------------------+
| +00:00              |
+---------------------+
1 row in set (0.00 sec)

mysql> select start_date from SCHEDULE;
+---------------------+
| start_date          |
+---------------------+
| 2024-07-24 13:00:00 |
| 2024-07-24 13:00:00 |
| 2024-07-24 13:30:00 |
| 2024-07-24 14:00:00 |
| 2024-07-24 14:30:00 |
| 2024-07-24 15:00:00 |
| 2024-07-24 15:00:00 |
| 2024-07-24 15:00:00 |
| 2024-07-24 15:30:00 |
| 2024-07-24 16:00:00 |
| 2024-07-24 17:00:00 |
| 2024-07-24 17:00:00 |
| 2024-07-24 17:00:00 |
| 2024-07-24 17:30:00 |
| 2024-07-24 18:00:00 |
| 2024-07-24 18:30:00 |
| 2024-07-24 19:00:00 |
| 2024-07-24 19:00:00 |
| 2024-07-24 19:00:00 |
```

returns 3895 rows displaying time in UTC

21.   From SCHEDULE TABLE get the start_date, end_date , vanue_code when status is cancelled



```
mysql> SELECT start_date, end_date, venue_code  FROM SCHEDULE WHERE status = 'CANCELLED';
+---------------------+---------------------+------------+
| start_date          | end_date            | venue_code |
+---------------------+---------------------+------------+
| 2024-07-29 06:30:00 | 2024-07-29 06:30:00 | CPL        |
| 2024-07-29 12:00:00 | 2024-07-29 12:00:00 | CPL        |
| 2024-07-30 12:50:00 | 2024-07-30 12:50:00 | CPL        |
| 2024-07-31 07:20:00 | 2024-07-31 07:20:00 | CPL        |
| 2024-08-01 11:36:00 | 2024-08-01 11:57:00 | MAM        |
| 2024-08-01 11:56:00 | 2024-08-01 12:17:00 | MAM        |
| 2024-08-01 12:07:00 | 2024-08-01 12:28:00 | MAM        |
| 2024-08-01 12:27:00 | 2024-08-01 12:48:00 | MAM        |
| 2024-08-01 15:38:00 | 2024-08-01 15:59:00 | MAM        |
| 2024-08-01 15:38:00 | 2024-08-01 15:59:00 | MAM        |
| 2024-08-01 16:09:00 | 2024-08-01 16:30:00 | MAM        |
| 2024-08-01 16:28:00 | 2024-08-01 16:49:00 | MAM        |
| 2024-08-01 16:28:00 | 2024-08-01 16:49:00 | MAM        |
| 2024-08-01 16:40:00 | 2024-08-01 17:01:00 | MAM        |
| 2024-08-01 16:59:00 | 2024-08-01 17:20:00 | MAM        |
| 2024-08-01 17:30:00 | 2024-08-01 17:51:00 | MAM        |
| 2024-08-01 18:01:00 | 2024-08-01 18:22:00 | MAM        |
| 2024-08-05 11:23:00 | 2024-08-05 12:14:00 | MAM        |
| 2024-08-05 12:45:00 | 2024-08-05 13:36:00 | MAM        |
| 2024-08-05 13:50:00 | 2024-08-05 14:41:00 | MAM        |
| 2024-08-06 12:29:00 | 2024-08-06 13:19:00 | MAM        |
| 2024-08-06 13:36:00 | 2024-08-06 14:26:00 | MAM        |
| 2024-08-07 10:03:00 | 2024-08-07 10:20:00 | MAM        |
| 2024-08-07 10:23:00 | 2024-08-07 10:40:00 | MAM        |
| 2024-08-07 10:43:00 | 2024-08-07 11:00:00 | MAM        |
| 2024-08-07 10:48:00 | 2024-08-07 11:05:00 | MAM        |
| 2024-08-07 11:13:00 | 2024-08-07 11:30:00 | MAM        |
| 2024-08-07 11:23:00 | 2024-08-07 11:40:00 | MAM        |
| 2024-08-07 11:38:00 | 2024-08-07 11:55:00 | MAM        |
| 2024-08-07 12:03:00 | 2024-08-07 12:20:00 | MAM        |
```

it displays 70 rows

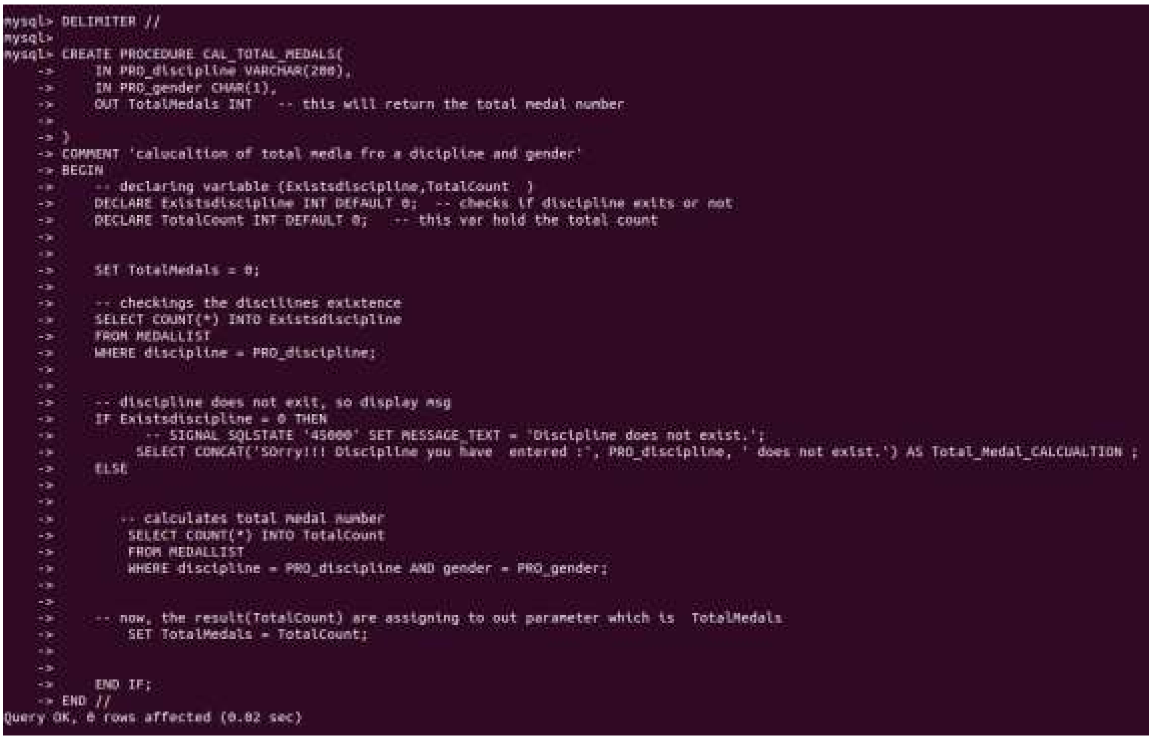
However, by sourcing **Query.sql** we can implement all these queries at a time and get all the result in a output file named “**SimpleQueryAnswerOlympic.out** ”

6. Implementation of Advanced Features (Procedures, Views)

To implement the advance features of queries by using Procedure like user variable, if-else, cursor, store procedure In, out parameter one can source the file name **“ advanceQuery.sql “** which also have the output file named **“ advanceQueryAnswerOlympic.out** “ . now we will do the queries one by one.

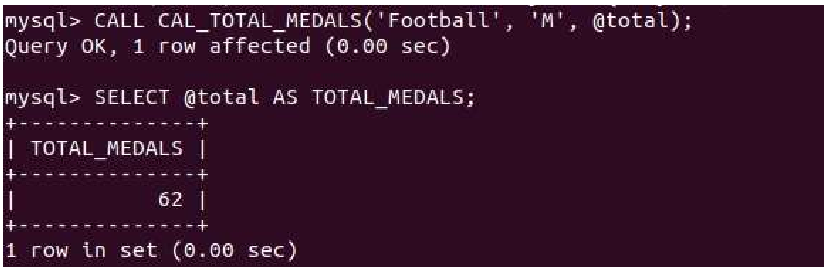
Procedure _TASK1

store procedure to calculate the total medals on a given game and gender .



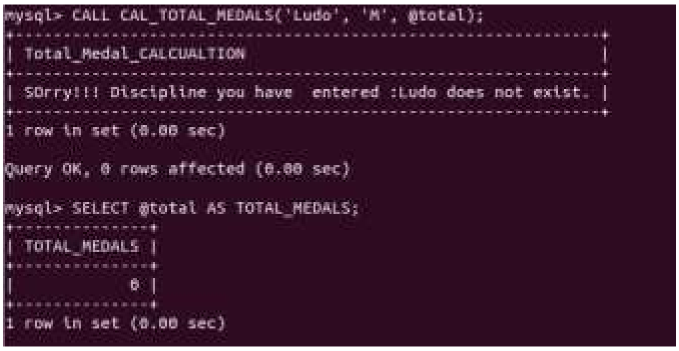
```
mysql> DELIMITER //
mysql>
mysql> CREATE PROCEDURE CAL_TOTAL_MEDALS(
    ->     IN PRO_discipline VARCHAR(200),
    ->     IN PRO_gender CHAR(1),
    ->     OUT TotalMedals INT   -- this will return the total medal number
    ->
    -> )
    -> COMMENT 'calucaltion of total medla fro a dicipline and gender'
    -> BEGIN
    ->     -- declaring variable (Existsdiscipline,TotalCount  )
    ->     DECLARE Existsdiscipline INT DEFAULT 0;  -- checks if discipline exits or not
    ->     DECLARE TotalCount INT DEFAULT 0;   -- this var hold the total count
    ->
    ->
    ->     SET TotalMedals = 0;
    ->
    ->     -- checkings the discilines exixtence
    ->     SELECT COUNT(*) INTO Existsdiscipline
    ->     FROM MEDALLIST
    ->     WHERE discipline = PRO_discipline;
    ->
    ->
    ->     -- discipline does not exit, so display msg
    ->     IF Existsdiscipline = 0 THEN
    ->           -- SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Discipline does not exist.';
    ->          SELECT CONCAT('SOrry!!! Discipline you have  entered :', PRO_discipline, ' does not exist.') AS Total_Medal_CALCUALTION ;
    ->     ELSE
    ->
    ->
    ->       -- calculates total medal number
    ->        SELECT COUNT(*) INTO TotalCount
    ->        FROM MEDALLIST
    ->        WHERE discipline = PRO_discipline AND gender = PRO_gender;
    ->
    ->
    ->     -- now, the result(TotalCount) are assigning to out parameter which is  TotalMedals
    ->        SET TotalMedals = TotalCount;
    ->
    ->
    ->     END IF;
    -> END //
Query OK, 0 rows affected (0.02 sec)
```

1.Then calling the procedure for total of Male gender in Football and displaying the total using user variable



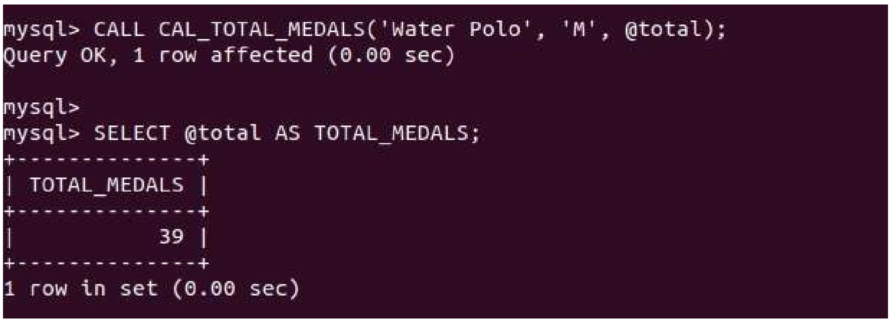
```
mysql> CALL CAL_TOTAL_MEDALS('Football', 'M', @total);
Query OK, 1 row affected (0.00 sec)

mysql> SELECT @total AS TOTAL_MEDALS;
+--------------+
| TOTAL_MEDALS |
+--------------+
|           62 |
+--------------+
1 row in set (0.00 sec)
```

2.Now, calling a PROCEDURE for a discipline that does not exit and return 0 as a total medal



```
mysql> CALL CAL_TOTAL_MEDALS('Ludo', 'M', @total);
+--------------------------------------------------------------+
| Total_Medal_CALCUALTION                                      |
+--------------------------------------------------------------+
| SOrry!!! Discipline you have  entered :Ludo does not exist. |
+--------------------------------------------------------------+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

mysql> SELECT @total AS TOTAL_MEDALS;
+--------------+
| TOTAL_MEDALS |
+--------------+
|            0 |
+--------------+
1 row in set (0.00 sec)
```

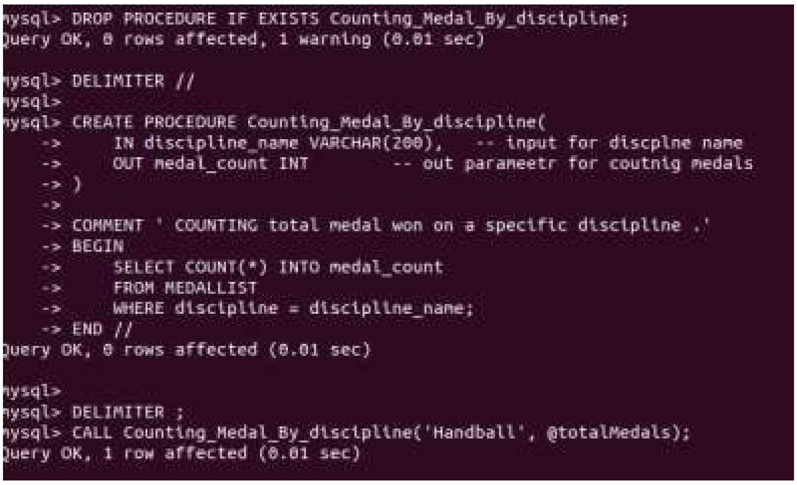
3.For “**water Polo”** discipline get out the total medals in male gender



```
mysql> CALL CAL_TOTAL_MEDALS('Water Polo', 'M', @total);
Query OK, 1 row affected (0.00 sec)

mysql>
mysql> SELECT @total AS TOTAL_MEDALS;
+--------------+
| TOTAL_MEDALS |
+--------------+
|           39 |
+--------------+
1 row in set (0.00 sec)
```

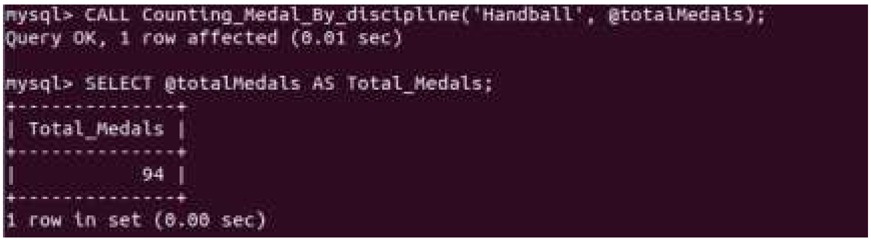
Procedure _TASK2:

using procedure user variable that counting total medals that have been won in a discipline



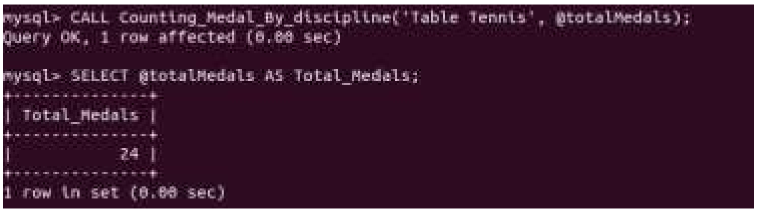
```
mysql> DROP PROCEDURE IF EXISTS Counting_Medal_By_discipline;
Query OK, 0 rows affected, 1 warning (0.01 sec)

mysql> DELIMITER //
mysql>
mysql> CREATE PROCEDURE Counting_Medal_By_discipline(
    ->     IN discipline_name VARCHAR(200),   -- input for discplne name
    ->     OUT medal_count INT        -- out parameetr for coutnig medals
    -> )
    ->
    -> COMMENT ' COUNTING total medal won on a specific discipline .'
    -> BEGIN
    ->     SELECT COUNT(*) INTO medal_count
    ->     FROM MEDALLIST
    ->     WHERE discipline = discipline_name;
    -> END //
Query OK, 0 rows affected (0.01 sec)

mysql>
mysql> DELIMITER ;
mysql> CALL Counting_Medal_By_discipline('Handball', @totalMedals);
Query OK, 1 row affected (0.01 sec)
```

1. Calling procedure for handball to get the total medals for ti



```
mysql> CALL Counting_Medal_By_discipline('Handball', @totalMedals);
Query OK, 1 row affected (0.01 sec)

mysql> SELECT @totalMedals AS Total_Medals;
+--------------+
| Total_Medals |
+--------------+
|           94 |
+--------------+
1 row in set (0.00 sec)
```

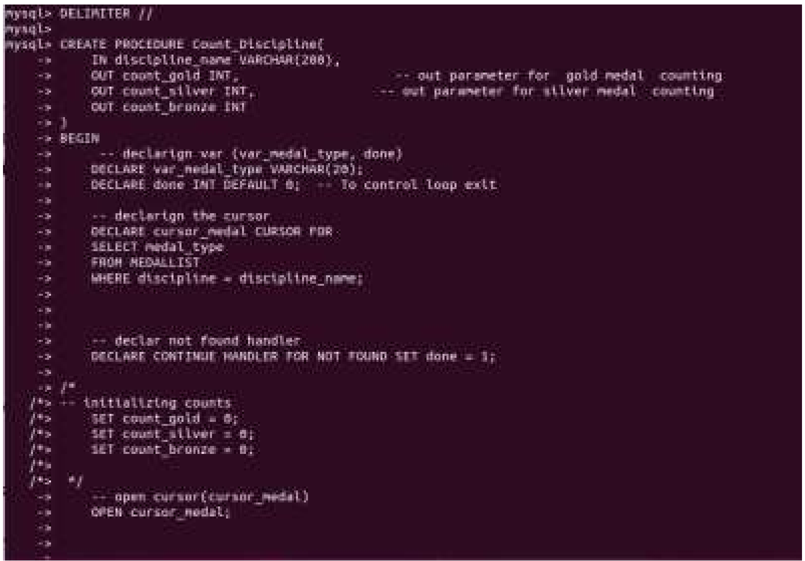
2. Calling procedure to get the total medals for table tennis



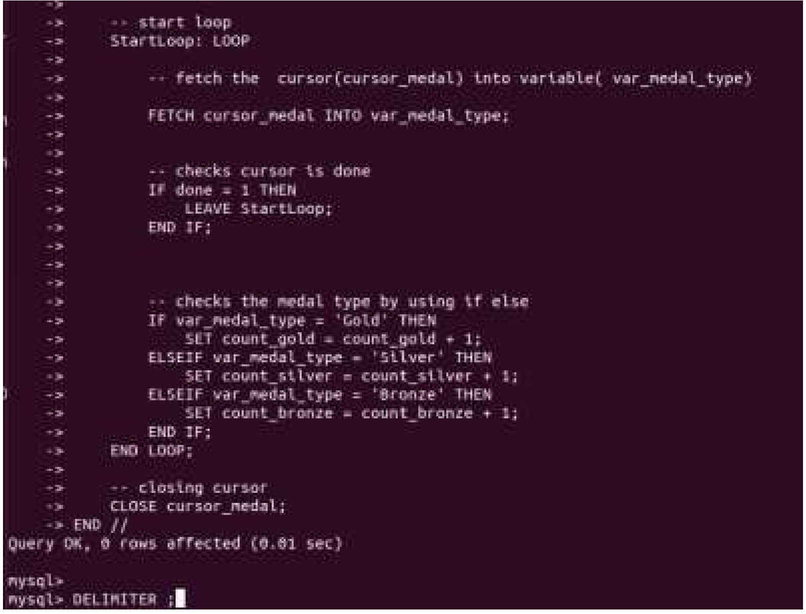
```
mysql> CALL Counting_Medal_By_discipline('Table Tennis', @totalMedals);
Query OK, 1 row affected (0.00 sec)

mysql> SELECT @totalMedals AS Total_Medals;
+--------------+
| Total_Medals |
+--------------+
|           24 |
+--------------+
1 row in set (0.00 sec)
```

Procedure _TASK3 :

store procedure using Cursor and LOOP to count medals (gold medals, silzer emdals, bronze medals) on a discipline

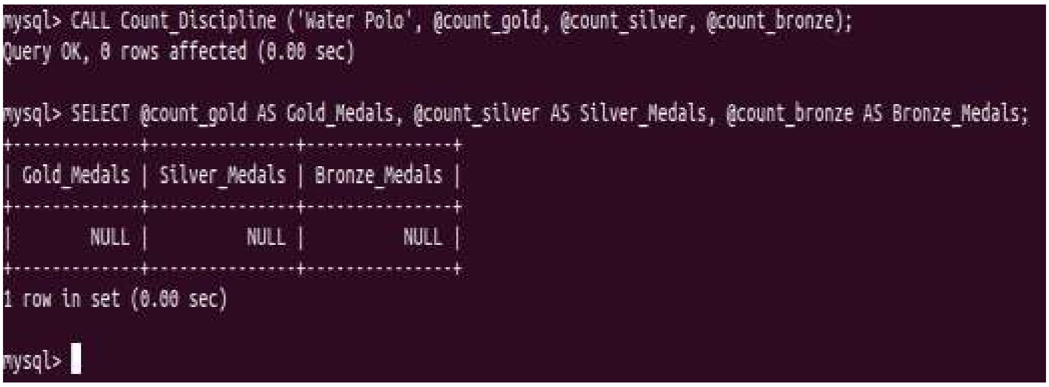


```
mysql> DELIMITER //
mysql>
mysql> CREATE PROCEDURE Count_Discipline(
    ->     IN discipline_name VARCHAR(200),
    ->     OUT count_gold INT,                    -- out parameter for  gold medal  counting
    ->     OUT count_silver INT,               -- out parameter for silver medal  counting
    ->     OUT count_bronze INT
    -> )
    -> BEGIN
    ->      -- declarign var (var_medal_type, done)
    ->     DECLARE var_medal_type VARCHAR(20);
    ->     DECLARE done INT DEFAULT 0;  -- To control loop exit
    ->
    ->     -- declarign the cursor
    ->     DECLARE cursor_medal CURSOR FOR
    ->     SELECT medal_type
    ->     FROM MEDALLIST
    ->     WHERE discipline = discipline_name;
    ->
    ->
    ->
    ->     -- declar not found handler
    ->     DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;
    ->
    -> /*
   /*> -- initializing counts
   /*>     SET count_gold = 0;
   /*>     SET count_silver = 0;
   /*>     SET count_bronze = 0;
   /*>
   /*>  */
    ->     -- open cursor(cursor_medal)
    ->     OPEN cursor_medal;
    ->
    ->
```



```
    ->
    ->     -- start loop
    ->     StartLoop: LOOP
    ->
    ->         -- fetch the  cursor(cursor_medal) into variable( var_medal_type)
    ->
    ->         FETCH cursor_medal INTO var_medal_type;
    ->
    ->
    ->         -- checks cursor is done
    ->         IF done = 1 THEN
    ->             LEAVE StartLoop;
    ->         END IF;
    ->
    ->
    ->
    ->         -- checks the medal type by using if else
    ->         IF var_medal_type = 'Gold' THEN
    ->             SET count_gold = count_gold + 1;
    ->         ELSEIF var_medal_type = 'Silver' THEN
    ->             SET count_silver = count_silver + 1;
    ->         ELSEIF var_medal_type = 'Bronze' THEN
    ->             SET count_bronze = count_bronze + 1;
    ->         END IF;
    ->     END LOOP;
    ->
    ->     -- closing cursor
    ->     CLOSE cursor_medal;
    -> END //
Query OK, 0 rows affected (0.01 sec)

mysql>
mysql> DELIMITER ;
```

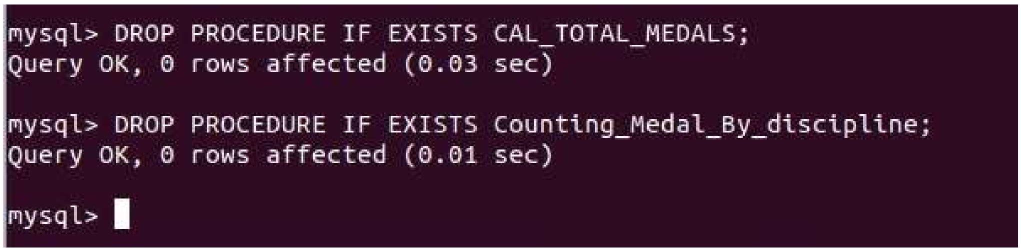
1.Calling the procedure for Water Polo to count the Gold Medal, silver medal, bronze medal



```
mysql> CALL Count_Discipline ('Water Polo', @count_gold, @count_silver, @count_bronze);
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT @count_gold AS Gold_Medals, @count_silver AS Silver_Medals, @count_bronze AS Bronze_Medals;
+-------------+---------------+---------------+
| Gold_Medals | Silver_Medals | Bronze_Medals |
+-------------+---------------+---------------+
|        NULL |          NULL |          NULL |
+-------------+---------------+---------------+
1 row in set (0.00 sec)

mysql> 
```

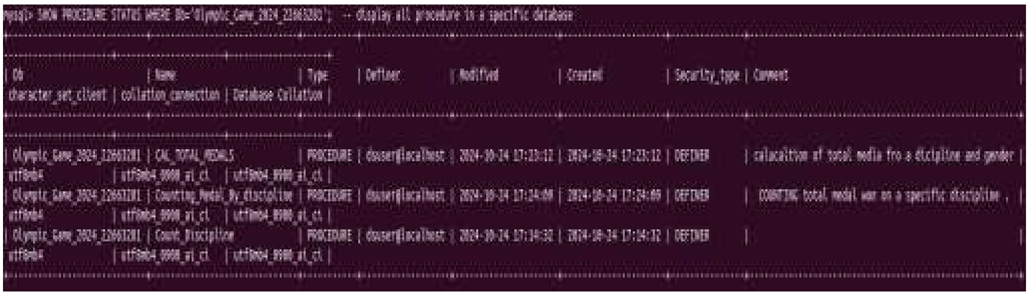
Some reason, it is not showing me the proper value. Showing me NULL.

2.To DELETE all the procedure we can use this command like this



```
mysql> DROP PROCEDURE IF EXISTS CAL_TOTAL_MEDALS;
Query OK, 0 rows affected (0.03 sec)

mysql> DROP PROCEDURE IF EXISTS Counting_Medal_By_discipline;
Query OK, 0 rows affected (0.01 sec)

mysql> 
```

3. SHOW PROCEDURE STATUS WHERE Db=**'Olympic_Game_2024_22663281';**



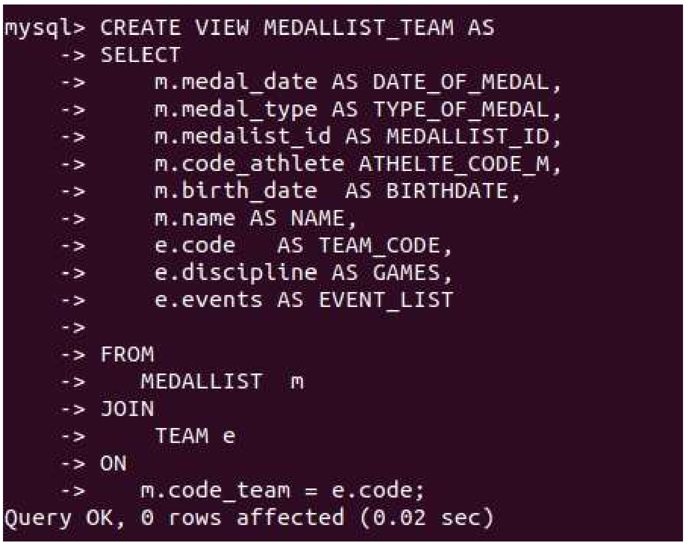
```
mysql> SHOW PROCEDURE STATUS WHERE Db='Olympic_Game_2024_22663281';  -- display all procedure in a specific database
+---------------------------+------------------------------+-----------+------------------+---------------------+---------------------+---------------+-------------------------------------------------------+
----------------------+----------------------+--------------------+
| Db                        | Name                         | Type      | Definer          | Modified            | Created             | Security_type | Comment                                               |
 character_set_client | collation_connection | Database Collation |
+---------------------------+------------------------------+-----------+------------------+---------------------+---------------------+---------------+-------------------------------------------------------+
----------------------+----------------------+--------------------+
| Olympic_Game_2024_22663281 | CAL_TOTAL_MEDALS             | PROCEDURE | dbuser@localhost | 2024-10-24 17:23:12 | 2024-10-24 17:23:12 | DEFINER       | calacaltion of total media fro a dicipline and gender |
 utf8mb4              | utf8mb4_0900_ai_ci   | utf8mb4_0900_ai_ci |
| Olympic_Game_2024_22663281 | Counting_Medal_By_discipline | PROCEDURE | dbuser@localhost | 2024-10-24 17:24:09 | 2024-10-24 17:24:09 | DEFINER       |  COUNTING total medal won on a specific discipline .  |
 utf8mb4              | utf8mb4_0900_ai_ci   | utf8mb4_0900_ai_ci |
| Olympic_Game_2024_22663281 | Count_Discipline             | PROCEDURE | dbuser@localhost | 2024-10-24 17:14:32 | 2024-10-24 17:14:32 | DEFINER       |                                                       |
 utf8mb4              | utf8mb4_0900_ai_ci   | utf8mb4_0900_ai_ci |
+---------------------------+------------------------------+-----------+------------------+---------------------+---------------------+---------------+-------------------------------------------------------+
```

VIEW_TASK1 :

Create VIEW name "**MEDALLIST_TEAM** " by joining M**EDALLIS**T and TEAM table

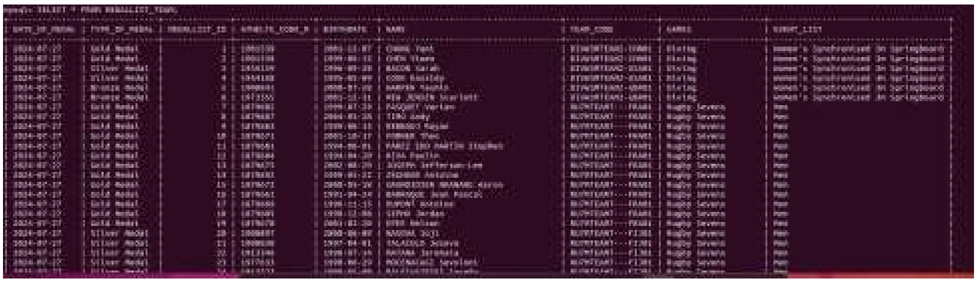
where MEDALLIST medal_date medal_type, medalist_id, code_athlete , birth_date, name and

FROM team discipline and events attribute will be included.



```
mysql> CREATE VIEW MEDALLIST_TEAM AS
    -> SELECT
    ->     m.medal_date AS DATE_OF_MEDAL,
    ->     m.medal_type AS TYPE_OF_MEDAL,
    ->     m.medallist_id AS MEDALLIST_ID,
    ->     m.code_athlete ATHELTE_CODE_M,
    ->     m.birth_date  AS BIRTHDATE,
    ->     m.name AS NAME,
    ->     e.code   AS TEAM_CODE,
    ->     e.discipline AS GAMES,
    ->     e.events AS EVENT_LIST
    ->
    -> FROM
    ->    MEDALLIST  m
    -> JOIN
    ->     TEAM e
    -> ON
    ->    m.code_team = e.code;
Query OK, 0 rows affected (0.02 sec)
```

By this command we can retrieve data from **view MEDALLIST_TEAM**

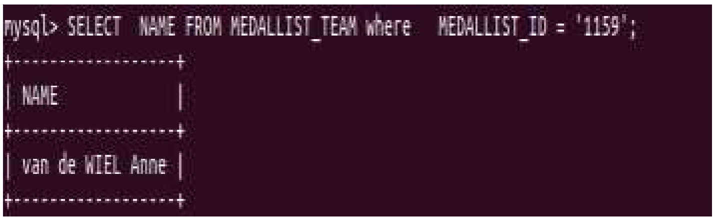


```
mysql> SELECT * FROM MEDALLIST_TEAM;
+---------------+---------------+--------------+----------------+------------+------+-----------+-------+------------+
| DATE_OF_MEDAL | TYPE_OF_MEDAL | MEDALLIST_ID | ATHELTE_CODE_M | BIRTHDATE  | NAME | TEAM_CODE | GAMES | EVENT_LIST |
+---------------+---------------+--------------+----------------+------------+------+-----------+-------+------------+
[illegible]
```

VIEW _TASK2:

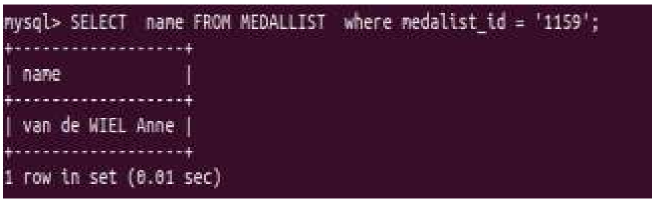
update name in **MEDALLIST TABLE** and checked on **VIEW MEDALLIST_TEAM** if the update was successful or not by calling procedure

let's say medallist_id 1159 changed the name and checked



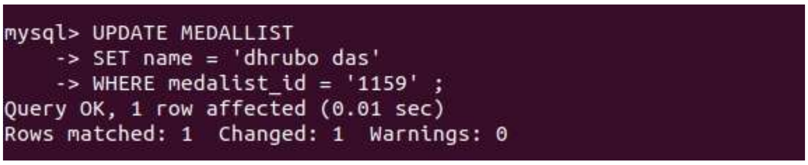
```
mysql> SELECT  NAME FROM MEDALLIST_TEAM where   MEDALLIST_ID = '1159';
+-----------------+
| NAME            |
+-----------------+
| van de WIEL Anne |
+-----------------+
```

1st checking the name in VIEW MEDALLIST_TEAM before updating



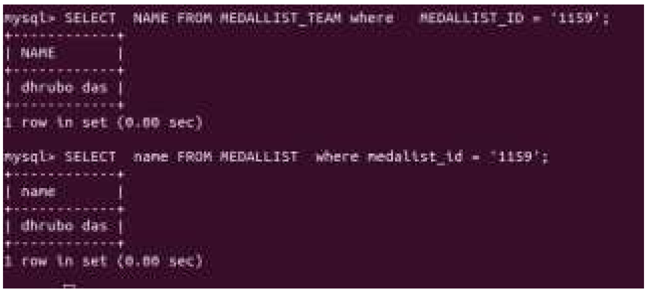
```
mysql> SELECT  name FROM MEDALLIST  where medallist_id = '1159';
+-----------------+
| name            |
+-----------------+
| van de WIEL Anne |
+-----------------+
1 row in set (0.01 sec)
```

now checking on MEDALIST TABLE



```
mysql> UPDATE MEDALLIST
    -> SET name = 'dhrubo das'
    -> WHERE medallist_id = '1159' ;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

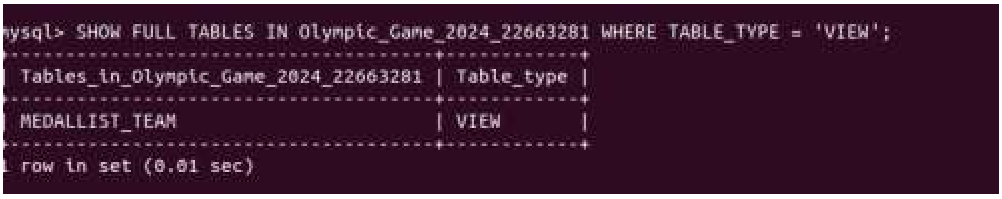
Updating the name IN MEDALLIST TABLE



```
mysql> SELECT  NAME FROM MEDALLIST_TEAM where   MEDALLIST_ID = '1159';
+------------+
| NAME       |
+------------+
| dhrubo das |
+------------+
1 row in set (0.00 sec)

mysql> SELECT  name FROM MEDALLIST  where medallist_id = '1159';
+------------+
| name       |
+------------+
| dhrubo das |
+------------+
1 row in set (0.00 sec)
```

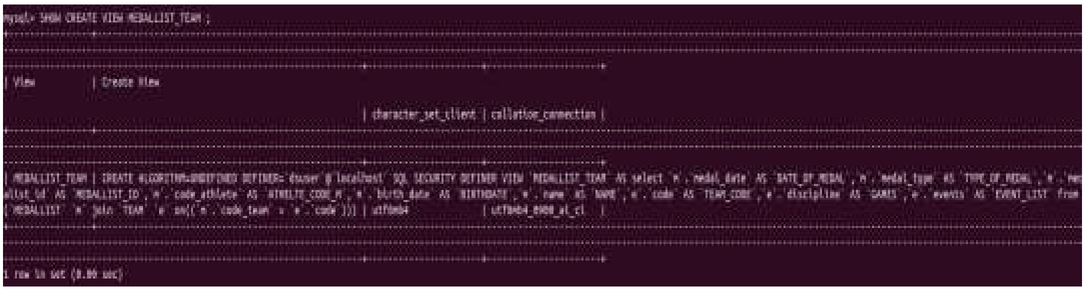
Again checking in MEDALLIST Table & MEDALLIST_TEAM VIEW if the update was successful or not

This query shows all the VIEWS in the DATABASE



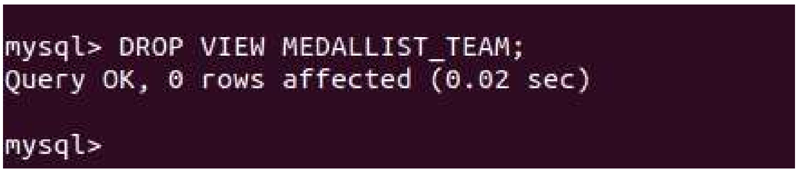
```
mysql> SHOW FULL TABLES IN Olympic_Game_2024_22663281 WHERE TABLE_TYPE = 'VIEW';
+-----------------------------------+------------+
| Tables_in_Olympic_Game_2024_22663281 | Table_type |
+-----------------------------------+------------+
| MEDALLIST_TEAM                    | VIEW       |
+-----------------------------------+------------+
1 row in set (0.01 sec)
```

This query shows the Specific VIEW in Details



```
mysql> SHOW CREATE VIEW MEDALLIST_TEAM ;
+----------------+------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------+----------------------+----------------------+
| View           | Create View
                                                           | character_set_client | collation_connection |
+----------------+------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------+----------------------+----------------------+
| MEDALLIST_TEAM | CREATE ALGORITHM=UNDEFINED DEFINER=`dsuser`@`localhost` SQL SECURITY DEFINER VIEW `MEDALLIST_TEAM` AS select `m`.`medal_date` AS `DATE_OF_MEDAL`,`m`.`medal_type` AS `TYPE_OF_MEDAL`,`m`.`me
allist_id` AS `MEDALLIST_ID`,`m`.`code_athlete` AS `ATHELTE_CODE_M`,`m`.`birth_date` AS `BIRTHDATE`,`m`.`name` AS `NAME`,`e`.`code` AS `TEAM_CODE`,`e`.`discipline` AS `GAMES`,`e`.`events` AS `EVENT_LIST` from
(`MEDALLIST` `m` join `TEAM` `e` on((`m`.`code_team` = `e`.`code`))) | utf8mb4              | utf8mb4_0900_ai_ci   |
+----------------+------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------+----------------------+----------------------+
1 row in set (0.00 sec)
```

Query to drop the VIEW



```
mysql> DROP VIEW MEDALLIST_TEAM;
Query OK, 0 rows affected (0.02 sec)

mysql>
```

7. Connecting Python to MySQL Database

I have this python files to do some operation .

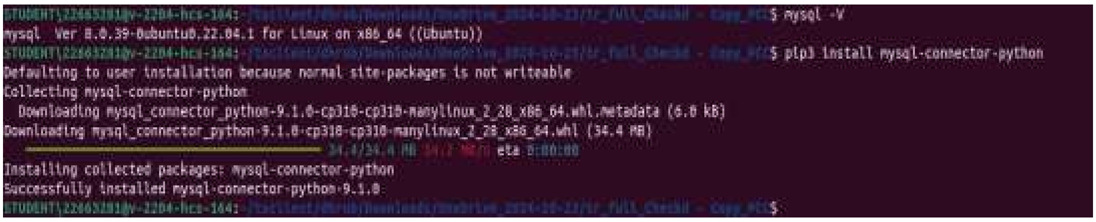
python3 eventdata.py

python3 scheduleData.py

python3 pythonConnect.py

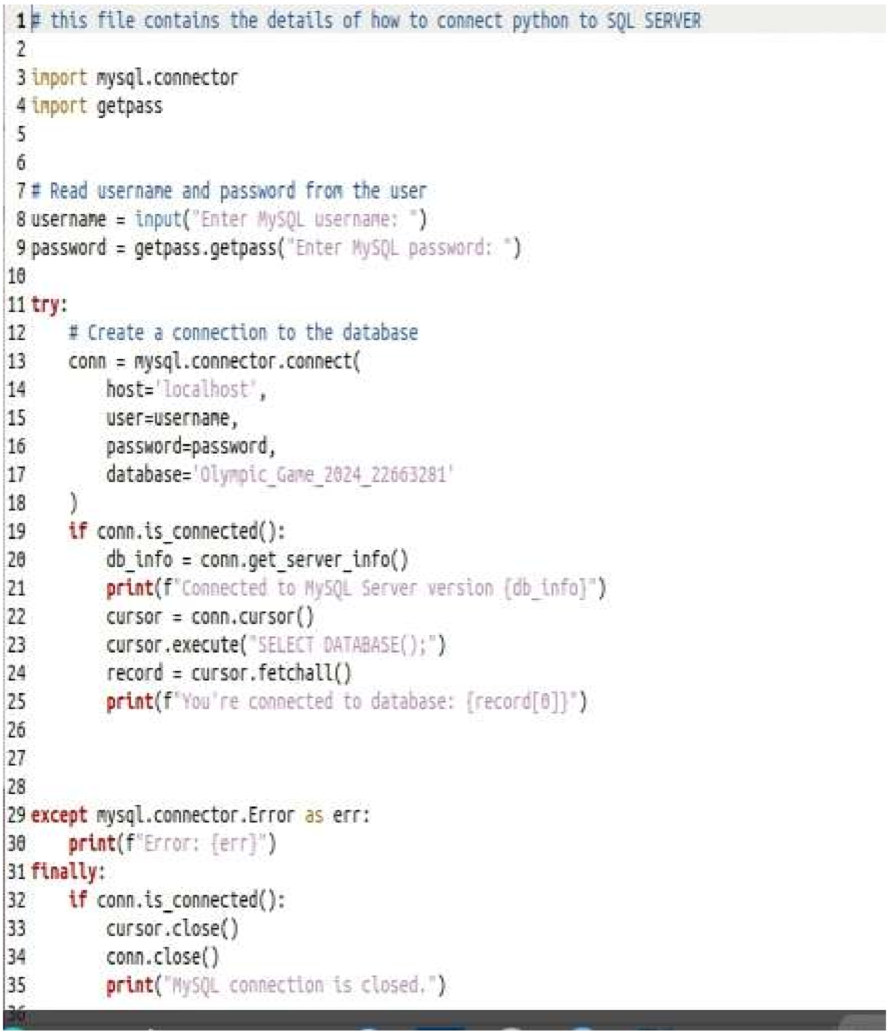
pyhton3 insert.py

7.1 **To connect MYSQL** Server with python environment need to do this at first like this:

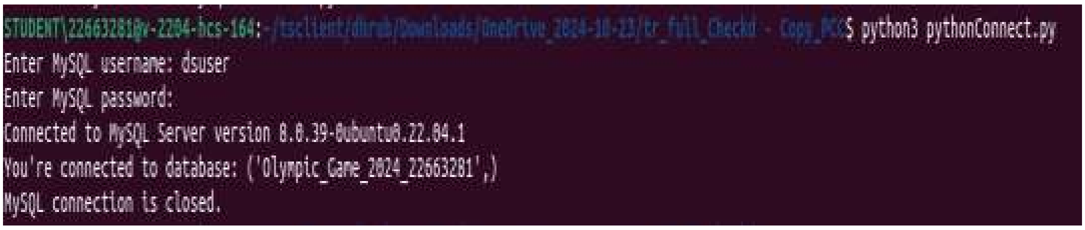


```
STUDENT\22663281@v-2204-hcs-164:~/tsclient/dhrub/Downloads/OneDrive_2024-10-23/tr_full_Checkd - Copy_PCC$ mysql -V
mysql  Ver 8.0.39-0ubuntu0.22.04.1 for Linux on x86_64 ((Ubuntu))
STUDENT\22663281@v-2204-hcs-164:~/tsclient/dhrub/Downloads/OneDrive_2024-10-23/tr_full_Checkd - Copy_PCC$ pip3 install mysql-connector-python
Defaulting to user installation because normal site-packages is not writeable
Collecting mysql-connector-python
  Downloading mysql_connector_python-9.1.0-cp310-cp310-manylinux_2_28_x86_64.whl.metadata (6.0 kB)
Downloading mysql_connector_python-9.1.0-cp310-cp310-manylinux_2_28_x86_64.whl (34.4 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 34.4/34.4 MB 54.2 MB/s eta 0:00:00
Installing collected packages: mysql-connector-python
Successfully installed mysql-connector-python-9.1.0
STUDENT\22663281@v-2204-hcs-164:~/tsclient/dhrub/Downloads/OneDrive_2024-10-23/tr_full_Checkd - Copy_PCC$
```

7.2 pythonConnect.py this file shows that we can connect the MYSQL Serve with python environment.

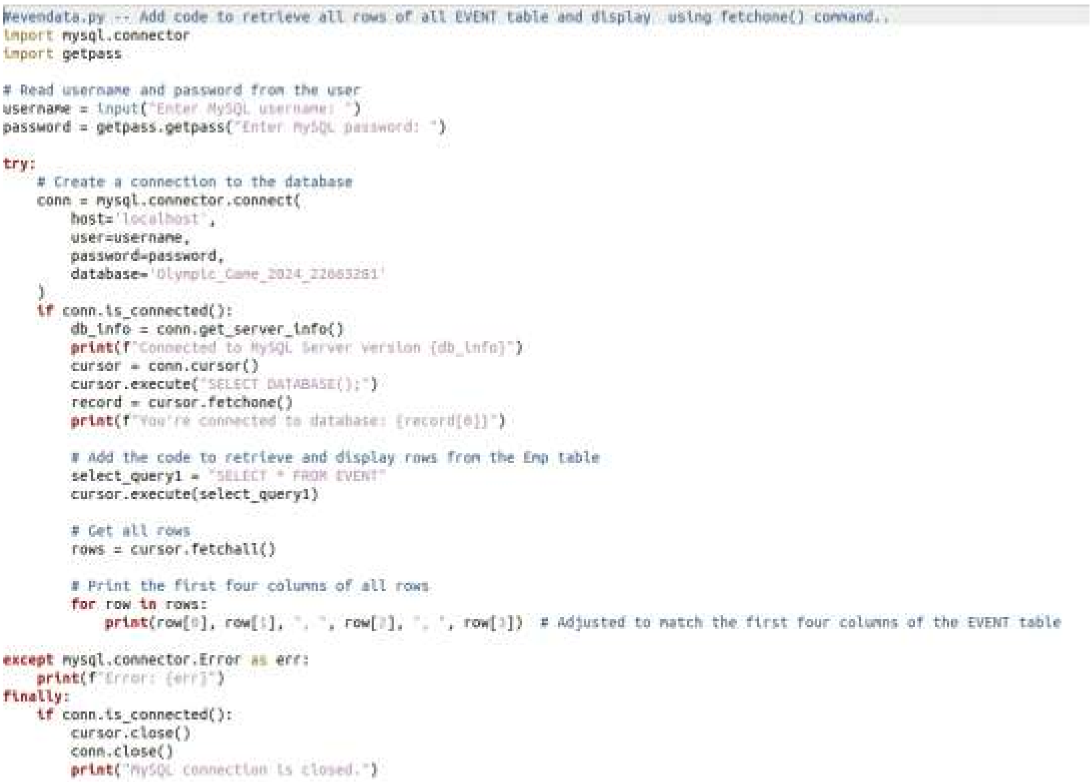


```python
# this file contains the details of how to connect python to SQL SERVER

import mysql.connector
import getpass


# Read username and password from the user
username = input("Enter MySQL username: ")
password = getpass.getpass("Enter MySQL password: ")

try:
    # Create a connection to the database
    conn = mysql.connector.connect(
        host='localhost',
        user=username,
        password=password,
        database='Olympic_Game_2024_22663281'
    )
    if conn.is_connected():
        db_info = conn.get_server_info()
        print(f"Connected to MySQL Server version {db_info}")
        cursor = conn.cursor()
        cursor.execute("SELECT DATABASE();")
        record = cursor.fetchall()
        print(f"You're connected to database: {record[0]}")



except mysql.connector.Error as err:
    print(f"Error: {err}")
finally:
    if conn.is_connected():
        cursor.close()
        conn.close()
        print("MySQL connection is closed.")
```



```
STUDENT\22663281@v-2204-hcs-164:~/tsclient/dhrub/Downloads/OneDrive_2024-10-23/tr_full_Checkd - Copy_PCC$ python3 pythonConnect.py
Enter MySQL username: dsuser
Enter MySQL password:
Connected to MySQL Server version 8.0.39-0ubuntu0.22.04.1
You're connected to database: ('Olympic_Game_2024_22663281',)
MySQL connection is closed.
```

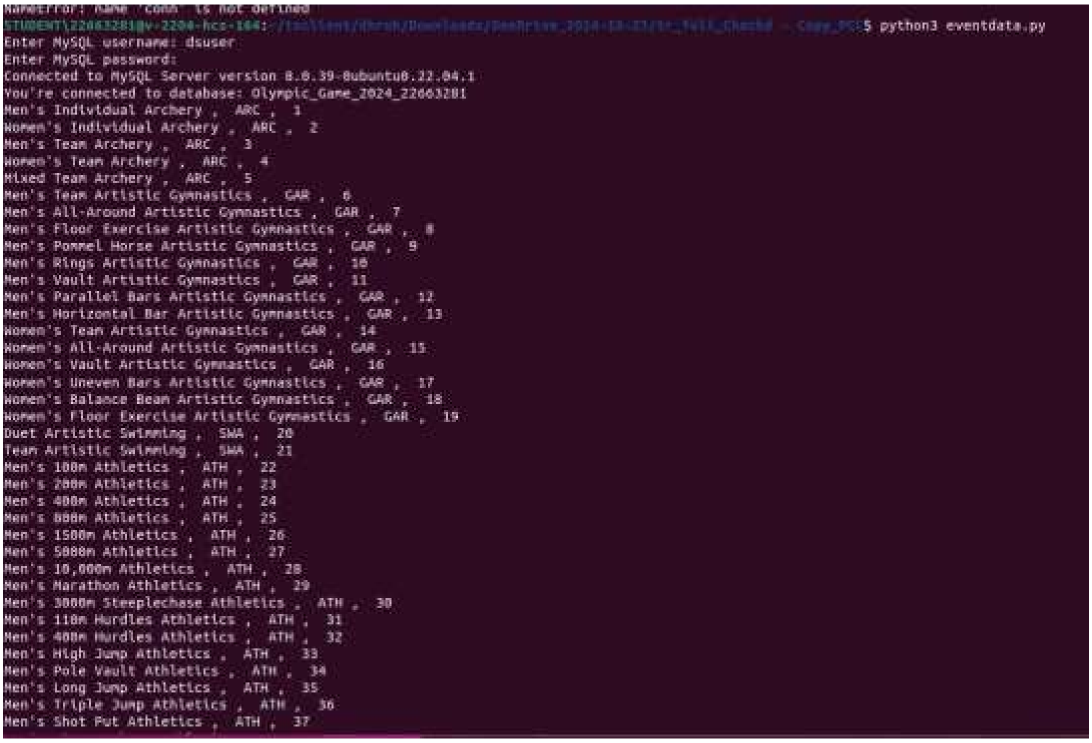
7.3 evendata.py -- to retrieve all rows of all EVENT table and display using fetchone() command..



```
#evendata.py -- Add code to retrieve all rows of all EVENT table and display  using fetchone() command..
import mysql.connector
import getpass

# Read username and password from the user
username = input("Enter MySQL username: ")
password = getpass.getpass("Enter MySQL password: ")

try:
    # Create a connection to the database
    conn = mysql.connector.connect(
        host='localhost',
        user=username,
        password=password,
        database='Olympic_Game_2024_22063281'
    )
    if conn.is_connected():
        db_info = conn.get_server_info()
        print(f"Connected to MySQL Server version {db_info}")
        cursor = conn.cursor()
        cursor.execute("SELECT DATABASE();")
        record = cursor.fetchone()
        print(f"You're connected to database: {record[0]}")

        # Add the code to retrieve and display rows from the Emp table
        select_query1 = "SELECT * FROM EVENT"
        cursor.execute(select_query1)

        # Get all rows
        rows = cursor.fetchall()

        # Print the first four columns of all rows
        for row in rows:
            print(row[0], row[1], ", ", row[2], ", ", row[3])  # Adjusted to match the first four columns of the EVENT table

except mysql.connector.Error as err:
    print(f"Error: {err}")
finally:
    if conn.is_connected():
        cursor.close()
        conn.close()
        print("MySQL connection is closed.")
```

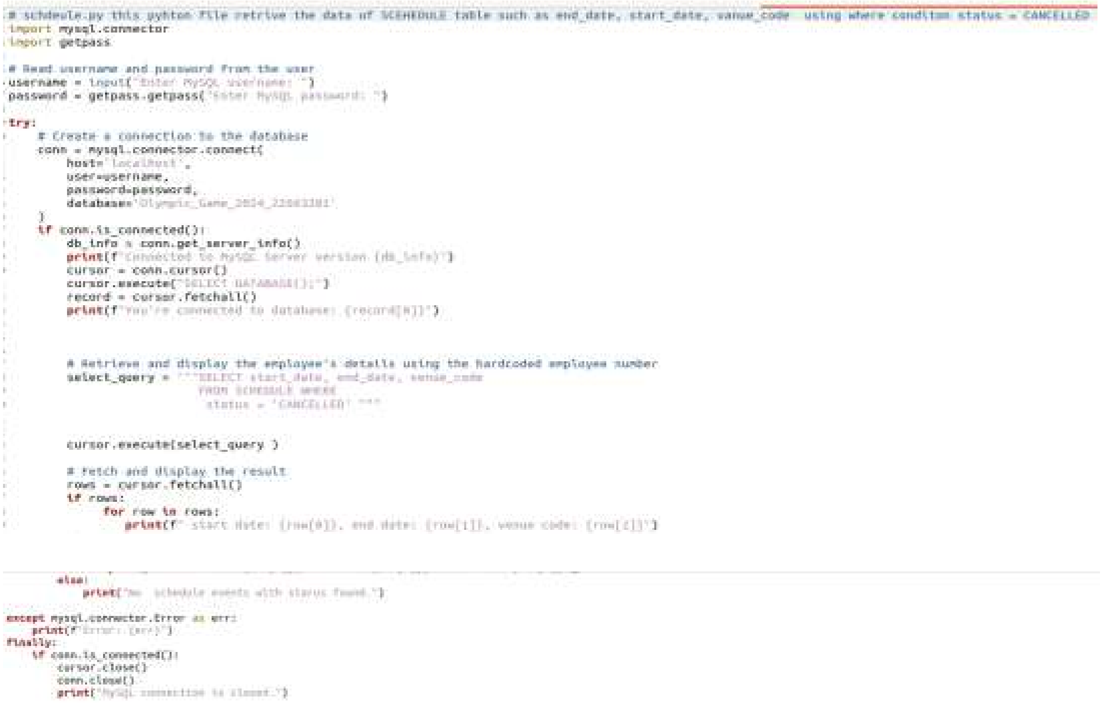
The above code we imported **library MYSQL** connector. to interacts with SQL server.

The fetchall() statement fetch all the rows from result table . The exception given prints the error if occurred. Then finally close the cursor and then the connection.



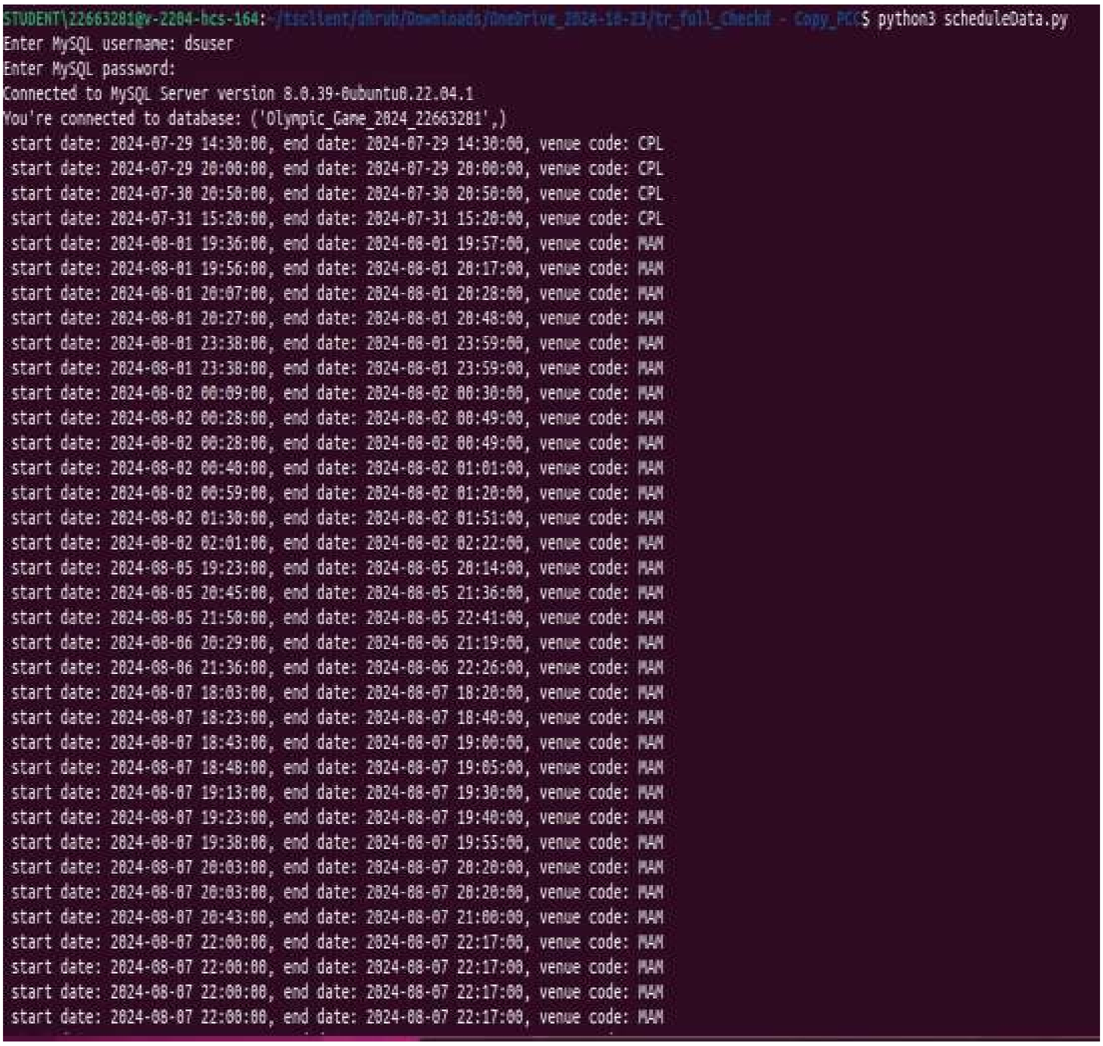
```
NameError: name 'conn' is not defined
STUDENT\22063281@v-2204-hcs-104:[illegible]$ python3 eventdata.py
Enter MySQL username: dsuser
Enter MySQL password:
Connected to MySQL Server version 8.0.39-0ubuntu0.22.04.1
You're connected to database: Olympic_Game_2024_22063281
Men's Individual Archery ,  ARC ,  1
Women's Individual Archery ,  ARC ,  2
Men's Team Archery ,  ARC ,  3
Women's Team Archery ,  ARC ,  4
Mixed Team Archery ,  ARC ,  5
Men's Team Artistic Gymnastics ,  GAR ,  6
Men's All-Around Artistic Gymnastics ,  GAR ,  7
Men's Floor Exercise Artistic Gymnastics ,  GAR ,  8
Men's Pommel Horse Artistic Gymnastics ,  GAR ,  9
Men's Rings Artistic Gymnastics ,  GAR ,  10
Men's Vault Artistic Gymnastics ,  GAR ,  11
Men's Parallel Bars Artistic Gymnastics ,  GAR ,  12
Men's Horizontal Bar Artistic Gymnastics ,  GAR ,  13
Women's Team Artistic Gymnastics ,  GAR ,  14
Women's All-Around Artistic Gymnastics ,  GAR ,  15
Women's Vault Artistic Gymnastics ,  GAR ,  16
Women's Uneven Bars Artistic Gymnastics ,  GAR ,  17
Women's Balance Beam Artistic Gymnastics ,  GAR ,  18
Women's Floor Exercise Artistic Gymnastics ,  GAR ,  19
Duet Artistic Swimming ,  SWA ,  20
Team Artistic Swimming ,  SWA ,  21
Men's 100m Athletics ,  ATH ,  22
Men's 200m Athletics ,  ATH ,  23
Men's 400m Athletics ,  ATH ,  24
Men's 800m Athletics ,  ATH ,  25
Men's 1500m Athletics ,  ATH ,  26
Men's 5000m Athletics ,  ATH ,  27
Men's 10,000m Athletics ,  ATH ,  28
Men's Marathon Athletics ,  ATH ,  29
Men's 3000m Steeplechase Athletics ,  ATH ,  30
Men's 110m Hurdles Athletics ,  ATH ,  31
Men's 400m Hurdles Athletics ,  ATH ,  32
Men's High Jump Athletics ,  ATH ,  33
Men's Pole Vault Athletics ,  ATH ,  34
Men's Long Jump Athletics ,  ATH ,  35
Men's Triple Jump Athletics ,  ATH ,  36
Men's Shot Put Athletics ,  ATH ,  37
```

7.4   schdeule.py this pyhton file retrive the data of SCEHEDULE table such as end_date, start_date, vanue_code using where conditon status = CANCELLED



```
# schdeule.py this pyhton file retrive the data of SCEHEDULE table such as end_date, start_date, venue_code  using where condtion status = CANCELLED
import mysql.connector
import getpass

# Read username and password from the user
username = input('Enter MySQL username: ')
password = getpass.getpass('Enter MySQL password: ')

try:
    # Create a connection to the database
    conn = mysql.connector.connect(
        host='localhost',
        user=username,
        password=password,
        database='Olympic_Game_2024_22663281'
    )
    if conn.is_connected():
        db_info = conn.get_server_info()
        print(f"Connected to MySQL Server version {db_info}")
        cursor = conn.cursor()
        cursor.execute("SELECT DATABASE();")
        record = cursor.fetchall()
        print(f"You're connected to database: {record[0]}")


        # Retrieve and display the employee's details using the hardcoded employee number
        select_query = """SELECT start_date, end_date, venue_code
                        FROM SCHEDULE WHERE
                         status = 'CANCELLED' """


        cursor.execute(select_query )

        # Fetch and display the result
        rows = cursor.fetchall()
        if rows:
            for row in rows:
                print(f" start date: {row[0]}, end date: {row[1]}, venue code: {row[2]}")

        else:
            print("No  schedule events with status found.")

except mysql.connector.Error as err:
    print(f"Error: {err}")
finally:
    if conn.is_connected():
        cursor.close()
        conn.close()
        print("MySQL connection is closed.")
```

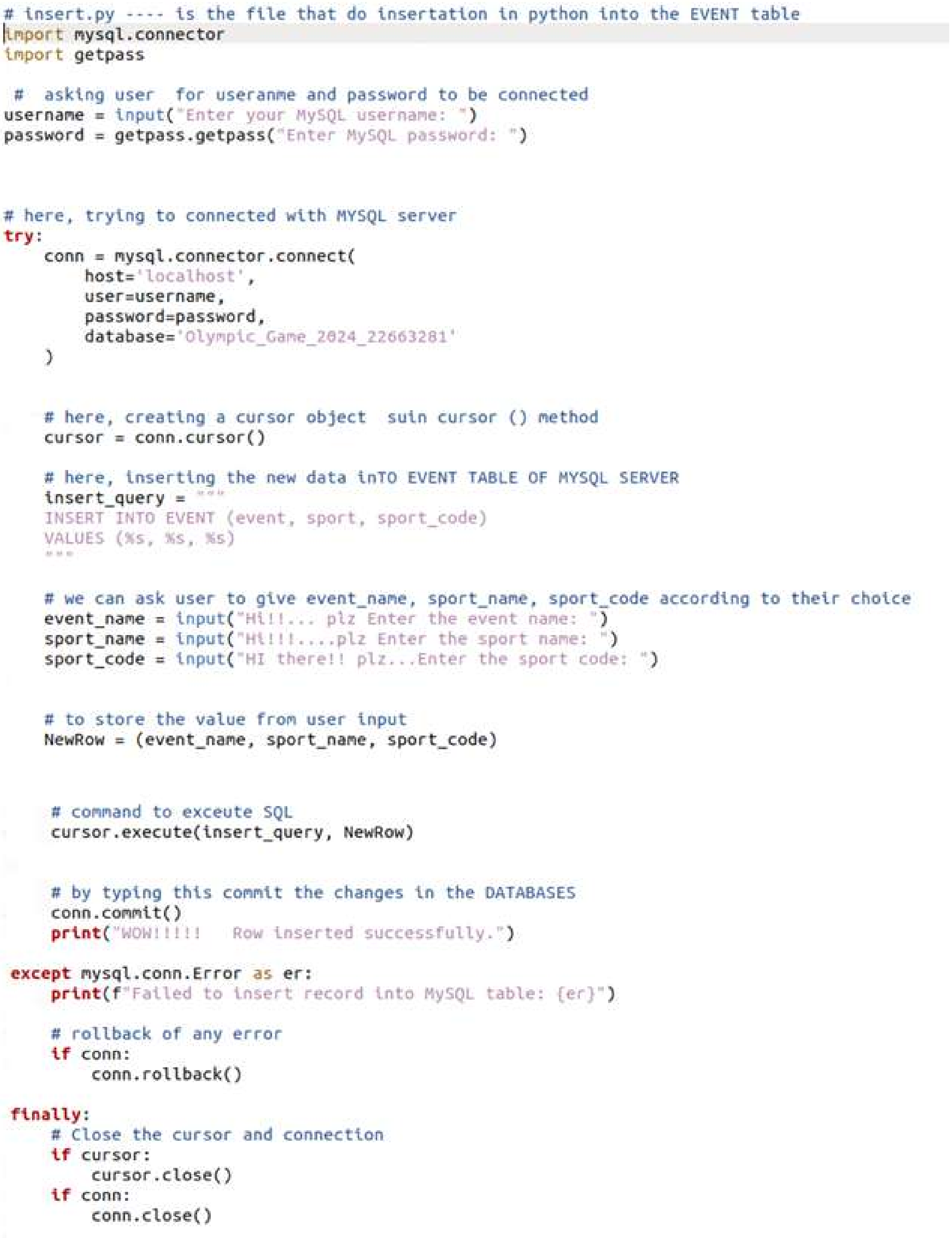
Shows me the result of start_date, end date and venue code that have been cancelled



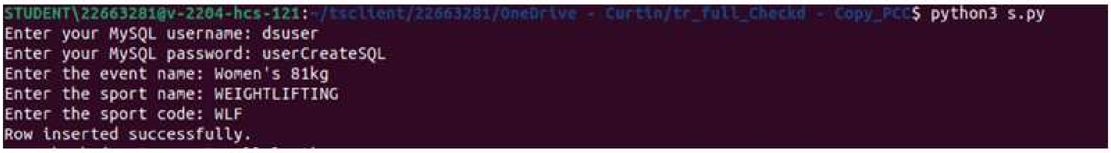
```
STUDENT\22663281@v-2204-hcs-164:~/tsclient/dhrub/Downloads/OneDrive_2024-10-23/tr_full_Checkd - Copy_PCC$ python3 scheduleData.py
Enter MySQL username: dsuser
Enter MySQL password:
Connected to MySQL Server version 8.0.39-0ubuntu0.22.04.1
You're connected to database: ('Olympic_Game_2024_22663281',)
 start date: 2024-07-29 14:30:00, end date: 2024-07-29 14:30:00, venue code: CPL
 start date: 2024-07-29 20:00:00, end date: 2024-07-29 20:00:00, venue code: CPL
 start date: 2024-07-30 20:50:00, end date: 2024-07-30 20:50:00, venue code: CPL
 start date: 2024-07-31 15:20:00, end date: 2024-07-31 15:20:00, venue code: CPL
 start date: 2024-08-01 19:36:00, end date: 2024-08-01 19:57:00, venue code: MAM
 start date: 2024-08-01 19:56:00, end date: 2024-08-01 20:17:00, venue code: MAM
 start date: 2024-08-01 20:07:00, end date: 2024-08-01 20:28:00, venue code: MAM
 start date: 2024-08-01 20:27:00, end date: 2024-08-01 20:48:00, venue code: MAM
 start date: 2024-08-01 23:38:00, end date: 2024-08-01 23:59:00, venue code: MAM
 start date: 2024-08-01 23:38:00, end date: 2024-08-01 23:59:00, venue code: MAM
 start date: 2024-08-02 00:09:00, end date: 2024-08-02 00:30:00, venue code: MAM
 start date: 2024-08-02 00:28:00, end date: 2024-08-02 00:49:00, venue code: MAM
 start date: 2024-08-02 00:28:00, end date: 2024-08-02 00:49:00, venue code: MAM
 start date: 2024-08-02 00:40:00, end date: 2024-08-02 01:01:00, venue code: MAM
 start date: 2024-08-02 00:59:00, end date: 2024-08-02 01:20:00, venue code: MAM
 start date: 2024-08-02 01:30:00, end date: 2024-08-02 01:51:00, venue code: MAM
 start date: 2024-08-02 02:01:00, end date: 2024-08-02 02:22:00, venue code: MAM
 start date: 2024-08-05 19:23:00, end date: 2024-08-05 20:14:00, venue code: MAM
 start date: 2024-08-05 20:45:00, end date: 2024-08-05 21:36:00, venue code: MAM
 start date: 2024-08-05 21:50:00, end date: 2024-08-05 22:41:00, venue code: MAM
 start date: 2024-08-06 20:29:00, end date: 2024-08-06 21:19:00, venue code: MAM
 start date: 2024-08-06 21:36:00, end date: 2024-08-06 22:26:00, venue code: MAM
 start date: 2024-08-07 18:03:00, end date: 2024-08-07 18:20:00, venue code: MAM
 start date: 2024-08-07 18:23:00, end date: 2024-08-07 18:40:00, venue code: MAM
 start date: 2024-08-07 18:43:00, end date: 2024-08-07 19:00:00, venue code: MAM
 start date: 2024-08-07 18:48:00, end date: 2024-08-07 19:05:00, venue code: MAM
 start date: 2024-08-07 19:13:00, end date: 2024-08-07 19:30:00, venue code: MAM
 start date: 2024-08-07 19:23:00, end date: 2024-08-07 19:40:00, venue code: MAM
 start date: 2024-08-07 19:38:00, end date: 2024-08-07 19:55:00, venue code: MAM
 start date: 2024-08-07 20:03:00, end date: 2024-08-07 20:20:00, venue code: MAM
 start date: 2024-08-07 20:03:00, end date: 2024-08-07 20:20:00, venue code: MAM
 start date: 2024-08-07 20:43:00, end date: 2024-08-07 21:00:00, venue code: MAM
 start date: 2024-08-07 22:00:00, end date: 2024-08-07 22:17:00, venue code: MAM
 start date: 2024-08-07 22:00:00, end date: 2024-08-07 22:17:00, venue code: MAM
 start date: 2024-08-07 22:00:00, end date: 2024-08-07 22:17:00, venue code: MAM
 start date: 2024-08-07 22:00:00, end date: 2024-08-07 22:17:00, venue code: MAM
```

7.4 : insert.py – is the file where we are inserting new value into EVENT tble. The user can insert a new event, sports sports_code according to their choice and then can cheked it into the **MYSQL SERVER** like this.

Here user can ask the event , sport_name sport code and the file will excetued it .



```
# insert.py ---- is the file that do insertation in python into the EVENT table
import mysql.connector
import getpass

 #  asking user  for useranme and password to be connected
username = input("Enter your MySQL username: ")
password = getpass.getpass("Enter MySQL password: ")



# here, trying to connected with MYSQL server
try:
    conn = mysql.connector.connect(
        host='localhost',
        user=username,
        password=password,
        database='Olympic_Game_2024_22663281'
    )


    # here, creating a cursor object  suin cursor () method
    cursor = conn.cursor()

    # here, inserting the new data inTO EVENT TABLE OF MYSQL SERVER
    insert_query = """
    INSERT INTO EVENT (event, sport, sport_code)
    VALUES (%s, %s, %s)
    """

    # we can ask user to give event_name, sport_name, sport_code according to their choice
    event_name = input("Hi!!... plz Enter the event name: ")
    sport_name = input("Hi!!!....plz Enter the sport name: ")
    sport_code = input("HI there!! plz...Enter the sport code: ")


    # to store the value from user input
    NewRow = (event_name, sport_name, sport_code)


     # command to exceute SQL
     cursor.execute(insert_query, NewRow)


     # by typing this commit the changes in the DATABASES
     conn.commit()
     print("WOW!!!!!   Row inserted successfully.")

 except mysql.conn.Error as er:
     print(f"Failed to insert record into MySQL table: {er}")

     # rollback of any error
     if conn:
         conn.rollback()

 finally:
     # Close the cursor and connection
     if cursor:
         cursor.close()
     if conn:
         conn.close()
```

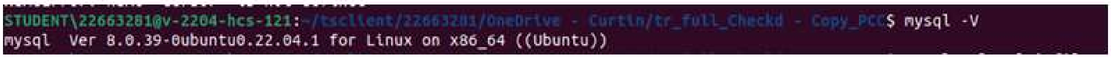


```
STUDENT\22663281@v-2204-hcs-121:~/tsclient/22663281/OneDrive - Curtin/tr_full_Checkd - Copy_PCC$ python3 s.py
Enter your MySQL username: dsuser
Enter your MySQL password: userCreateSQL
Enter the event name: Women's 81kg
Enter the sport name: WEIGHTLIFTING
Enter the sport code: WLF
Row inserted successfully.
```

Sorry, I renamed the file s.py as insert.py which contains the same information as s.py.

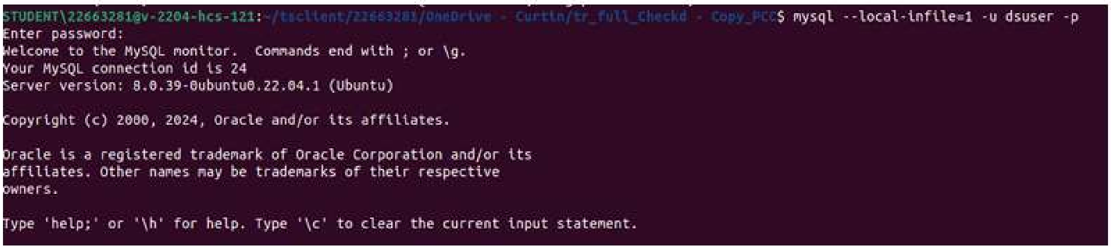
Now we are checking if the data is inserted or not . by checking into MYSQL SERVER.

Step:1.



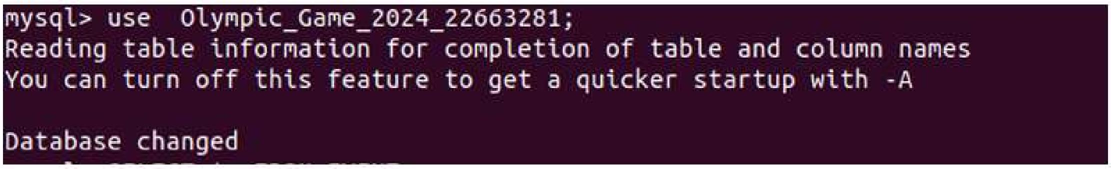
```
STUDENT\22663281@v-2204-hcs-121:~/tsclient/22663281/OneDrive - Curtin/tr_full_Checkd - Copy_PCC$ mysql -V
mysql  Ver 8.0.39-0ubuntu0.22.04.1 for Linux on x86_64 ((Ubuntu))
```

Step2:



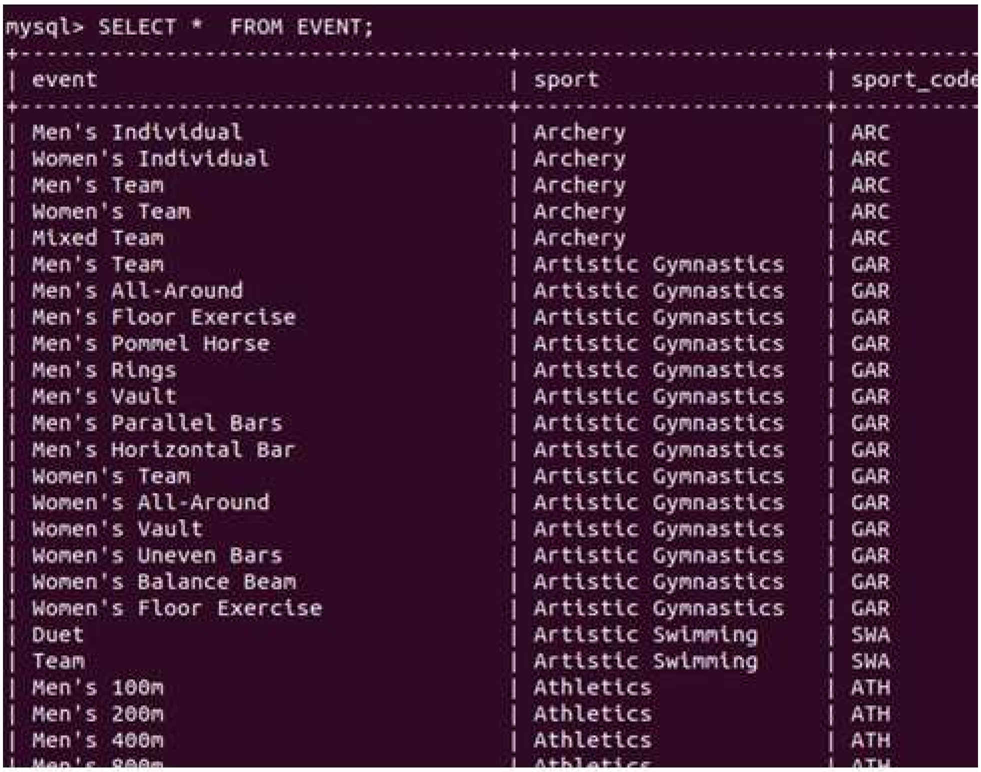
```
STUDENT\22663281@v-2204-hcs-121:~/tsclient/22663281/OneDrive - Curtin/tr_full_Checkd - Copy_PCC$ mysql --local-infile=1 -u dsuser -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 24
Server version: 8.0.39-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

Step3 :SELECT the database

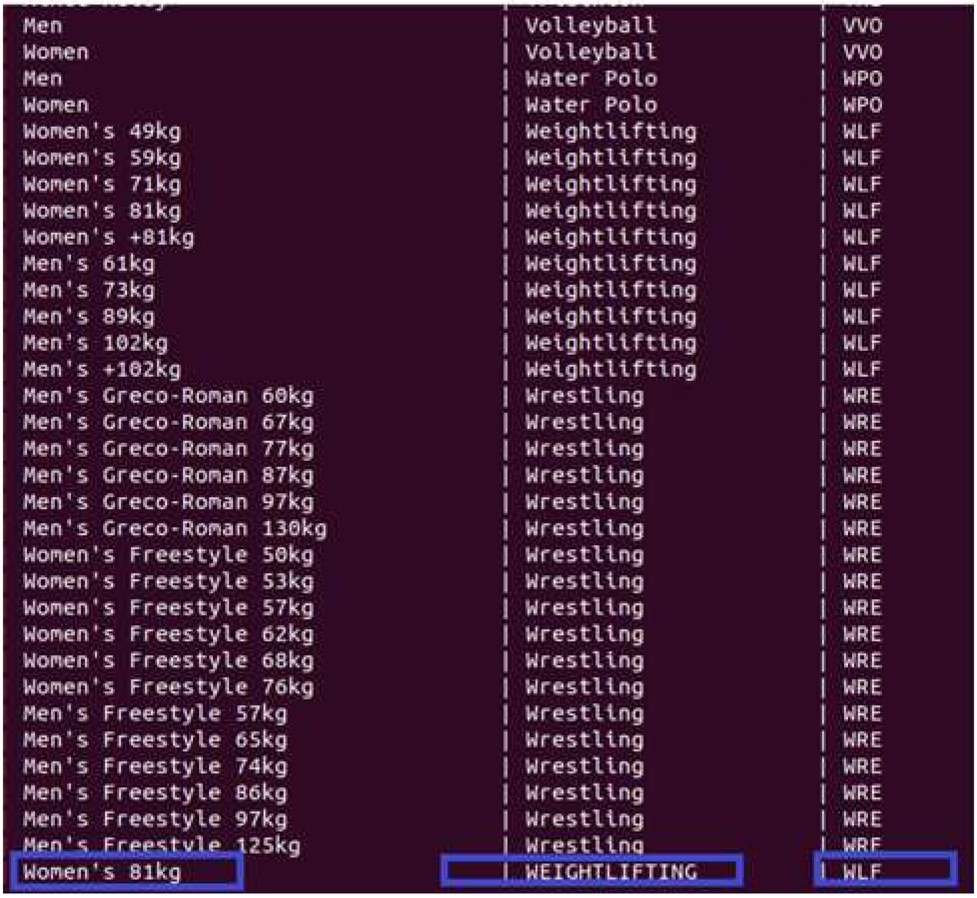


```
mysql> use  Olympic_Game_2024_22663281;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
```

Step4 : retrieving all data from EVENT table



```
mysql> SELECT *  FROM EVENT;
+-------------------------------------+-----------------------+----------
| event                               | sport                 | sport_code
+-------------------------------------+-----------------------+----------
| Men's Individual                    | Archery               | ARC
| Women's Individual                  | Archery               | ARC
| Men's Team                          | Archery               | ARC
| Women's Team                        | Archery               | ARC
| Mixed Team                          | Archery               | ARC
| Men's Team                          | Artistic Gymnastics   | GAR
| Men's All-Around                    | Artistic Gymnastics   | GAR
| Men's Floor Exercise                | Artistic Gymnastics   | GAR
| Men's Pommel Horse                  | Artistic Gymnastics   | GAR
| Men's Rings                         | Artistic Gymnastics   | GAR
| Men's Vault                         | Artistic Gymnastics   | GAR
| Men's Parallel Bars                 | Artistic Gymnastics   | GAR
| Men's Horizontal Bar                | Artistic Gymnastics   | GAR
| Women's Team                        | Artistic Gymnastics   | GAR
| Women's All-Around                  | Artistic Gymnastics   | GAR
| Women's Vault                       | Artistic Gymnastics   | GAR
| Women's Uneven Bars                 | Artistic Gymnastics   | GAR
| Women's Balance Beam                | Artistic Gymnastics   | GAR
| Women's Floor Exercise              | Artistic Gymnastics   | GAR
| Duet                                | Artistic Swimming     | SWA
| Team                                | Artistic Swimming     | SWA
| Men's 100m                          | Athletics             | ATH
| Men's 200m                          | Athletics             | ATH
| Men's 400m                          | Athletics             | ATH
| Men's 800m                          | Athletics             | ATH
```



```
 Men                                 | Volleyball            | VVO
 Women                               | Volleyball            | VVO
 Men                                 | Water Polo            | WPO
 Women                               | Water Polo            | WPO
 Women's 49kg                        | Weightlifting         | WLF
 Women's 59kg                        | Weightlifting         | WLF
 Women's 71kg                        | Weightlifting         | WLF
 Women's 81kg                        | Weightlifting         | WLF
 Women's +81kg                       | Weightlifting         | WLF
 Men's 61kg                          | Weightlifting         | WLF
 Men's 73kg                          | Weightlifting         | WLF
 Men's 89kg                          | Weightlifting         | WLF
 Men's 102kg                         | Weightlifting         | WLF
 Men's +102kg                        | Weightlifting         | WLF
 Men's Greco-Roman 60kg              | Wrestling             | WRE
 Men's Greco-Roman 67kg              | Wrestling             | WRE
 Men's Greco-Roman 77kg              | Wrestling             | WRE
 Men's Greco-Roman 87kg              | Wrestling             | WRE
 Men's Greco-Roman 97kg              | Wrestling             | WRE
 Men's Greco-Roman 130kg             | Wrestling             | WRE
 Women's Freestyle 50kg              | Wrestling             | WRE
 Women's Freestyle 53kg              | Wrestling             | WRE
 Women's Freestyle 57kg              | Wrestling             | WRE
 Women's Freestyle 62kg              | Wrestling             | WRE
 Women's Freestyle 68kg              | Wrestling             | WRE
 Women's Freestyle 76kg              | Wrestling             | WRE
 Men's Freestyle 57kg                | Wrestling             | WRE
 Men's Freestyle 65kg                | Wrestling             | WRE
 Men's Freestyle 74kg                | Wrestling             | WRE
 Men's Freestyle 86kg                | Wrestling             | WRE
 Men's Freestyle 97kg                | Wrestling             | WRE
 Men's Freestyle 125kg               | Wrestling             | WRE
 Women's 81kg                        | WEIGHTLIFTING         | WLF
```

here the **BLUE marked** row is the newly inserted data that we did by using python insert.py. So it can be said that the insert.py file is working correctly.

It is successfully inserting the data using python environment into MySQL SERVER.

8. Discussion

The implementation and the execution of Olympic game 2024 was really interesting as well was challenging. The whole part was sorting data and implementing the queries according to individual condition and create the queries. It was like making your own queries and then get the output based on that using the MYSQL theories of joining, advance features like PROCEDURE and VIEWS.

While implementing and sorting my data to get my output I also got knowledge of so many countries coach name, player name, and which country got the most medal in specific game and a lot more information. that I was unaware of.

Moreover, it was also challenging, because sorting data was not easy and to create the queries was also a long process as there are many more data and many information that I can also work. I might be work well, but happy t to present my data of Olympic games 2024.Overall, I was happy and blessed to work with this which enrich my knowledge of SQL