

Project Name: Olympic Games Database

Version 1.0

Table of Content

1. Problem statement

2. ER diagrams

3. Relational schema

4. Insert and create statements

5. SQL queries.

## 

## 

## 

## 

## 

## **3. Project Overview**

### 3.1 Project Overview and Background

The Olympic Games database is an endeavor at tracking data of the Olympic games over a span of the past few decades. A vast amount of information and facts regarding the participants and teams are garnered over a span of approximately two weeks in which the Olympic games are played. These include scheduling information, individual feats, team records, nation-wise medal tally, and much more.

Information about players, the various gaming sports, coaching staff, and events will be stored and maintained in the database. The database also holds details regarding each event schedule and their respective results. Additionally, it maintains the data of medal winners and Olympic firsts at the Tokyo Olympics. The Olympic Games database can be used to record, maintain and conduct transactional queries regarding participant information, event schedules, medals won by a country, etc.

Olympic Database will be loaded through an ETL tool, which extracts data from the web, transforms it (clean and format), and loads it into the Olympic Database. Users will have a web-enabled database containing Olympic facts from 2000 to the present. Key features of the system include storage of participant and team details, participating nations (including exceptions like ROC - Russian Olympic Committee), and winners, amongst others. Each game’s schedule is recorded separately, and their outcomes are taken into account. It is important to note that this database management system only contains data pertaining to the schedule information of the Olympic Games post-completion and does not record data of ongoing events. However, the entity-relationship model and the system developed has the provision to deal with the latter as well.

The Olympics DBMS may be utilized by interested spectators as well as broadcasting channels. With the information from this database at hand, users will be able to compute interesting statistics of individual participants and teams and a formatted table of the result of the user’s query will be returned. Furthermore, they can also gather data regarding a particular event’s schedule and the corresponding outcome of the game. These facts can also be used in any data visualization tool for further analysis. The system can also be queried to perform descriptive analytics on data such as finding the medal tally of nations, significant participant demographics and information, and record country-wise participation in this global sporting event.

### 3.2 Project Dependencies

N/A

### 3.3 Stakeholders

The following comprises the internal and external stakeholders whose requirements are represented by this document:

| **S.No.** | **Stakeholders** |
| --- | --- |
| 1. | Sports Channels |
| 2. | Olympics Organizing Committee |
| 3. | Olympic Participants |

## 

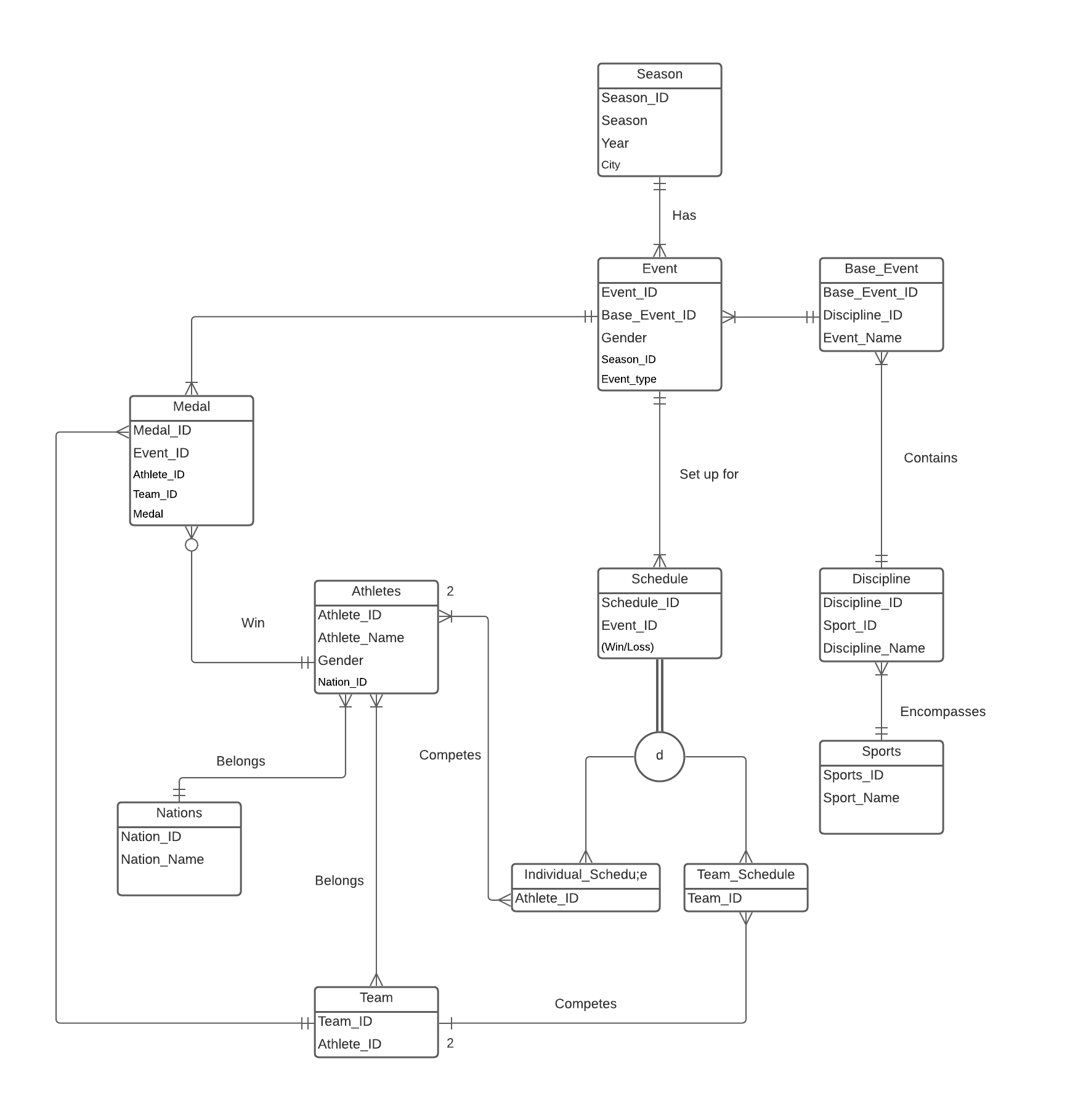
## 

## 

## 

## 

## 4. ENTITY RELATIONSHIP DIAGRAM



## 

## 

## 

## 

## 5. Entity Relationship Diagram Explanation

· An athlete playing in the Olympic Games belongs to only one nation.

· A nation participating in Olympic Games must have at least one athlete.

· The Olympic games has many sports.

· Each sport encompasses many disciplines.

· A discipline can be part of only one sport.

· Every discipline has its own base events.

· A base event belongs to only one discipline.

· A base event has events based on Gender.

· An event can have only one base event related to it.

· A schedule is set up for an event

· Every schedule has a result

· A result may or may not have a medal.

· Medals are awarded only for results. (i.e., only one medal is associated with each result in a final event)

· An athlete may be scheduled for an event.

· A schedule must have many athletes.

· An event must be part of a particular season.

· A season must have many events scheduled.

· An athlete may win many medals (1 medal for a different game).

· A medal must be secured by an athlete.

## 

## 

## 

## 

## 

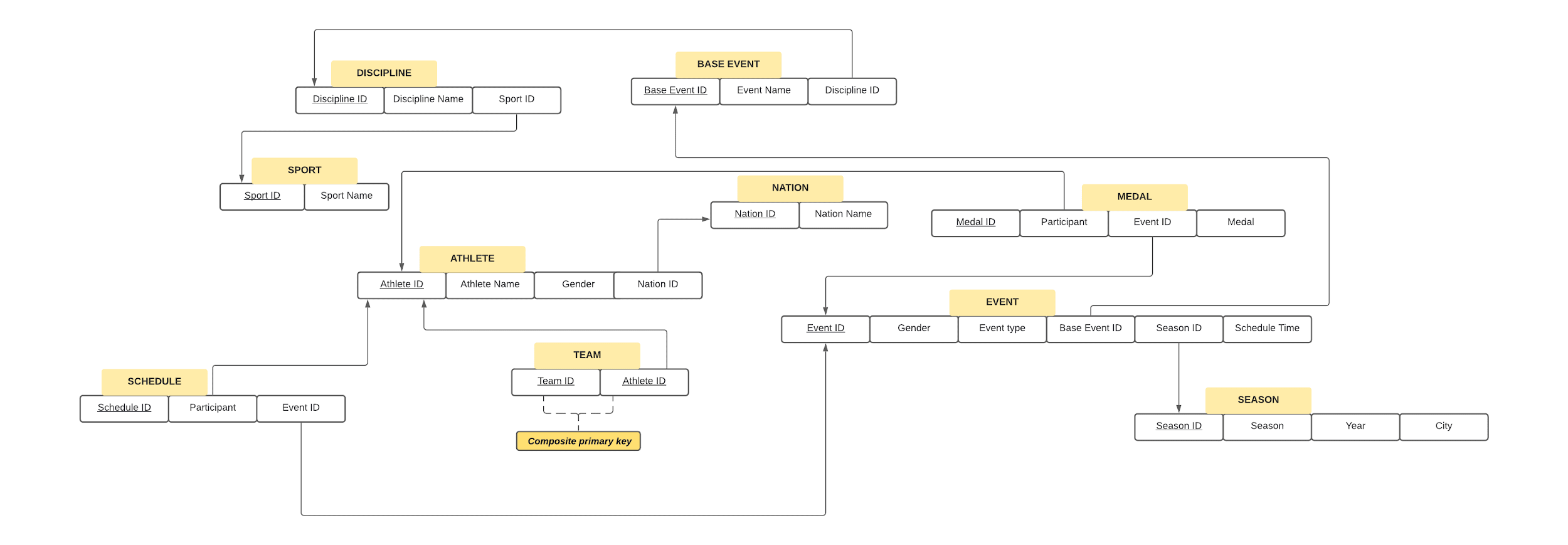
## 

## 

## 

## 

## 6. Logical Diagram



## 7. Database Structures

PFB the following link that contains the DDL Commands that were used to create the tables

<https://livesql.oracle.com/apex/livesql/s/mld4z72en2vwja14b9ez8bsws>

Database Schema for Student Admission DB

## 8. SQL Queries

1. Find the Top 10 countries with most players

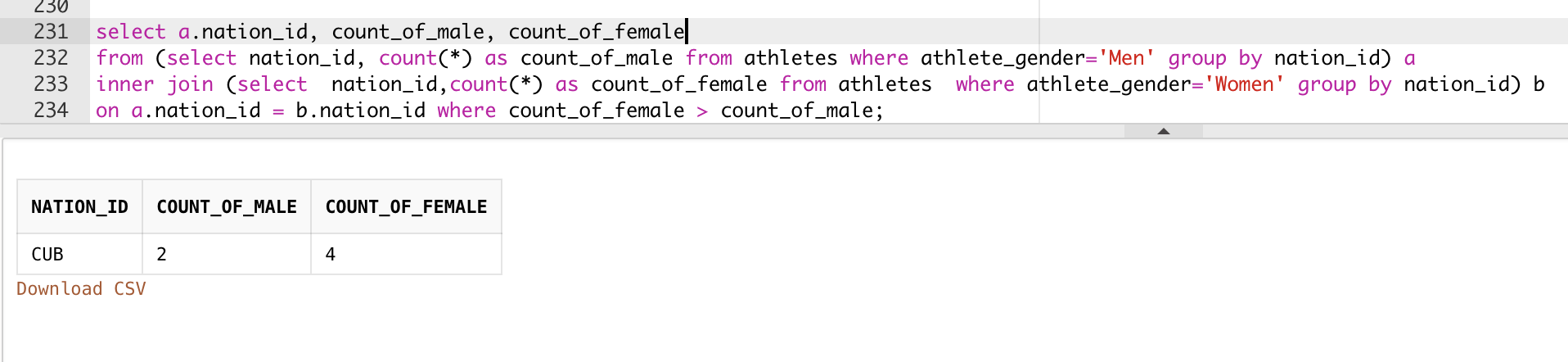
Query -

*select \* from  
 (select distinct(nation\_id), count(\*) over (partition by nation\_id ) "Total\_players" from athletes order by "Total\_players" desc) where rownum <= 10;*



1. Display the nations which has more female players than male players

*select a.nation\_id, count\_of\_male, count\_of\_female  
from (select nation\_id, count(\*) as count\_of\_male from athletes where athlete\_gender='Men' group by nation\_id) a  
inner join (select nation\_id,count(\*) as count\_of\_female from athletes where athlete\_gender='Women' group by nation\_id) b  
on a.nation\_id = b.nation\_id where count\_of\_female > count\_of\_male;*



1. Prioritize Olympic Sport Disciplines based on the number of events in each discipline for scheduling purposes.

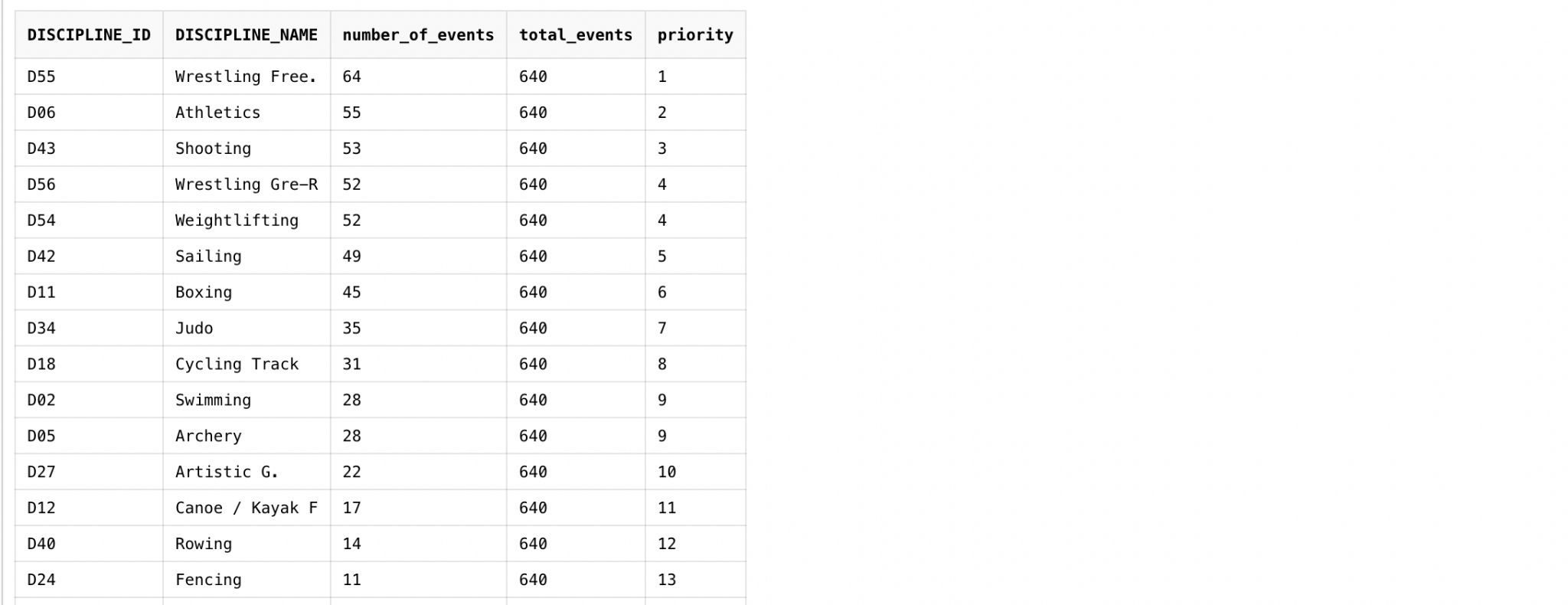
*select a.DISCIPLINE\_ID, DISCIPLINE\_NAME, "number\_of\_events", "total\_events", dense\_rank() over (order by "number\_of\_events" desc) "priority" from*

*(select distinct(discipline\_id), count(\*) over() "total\_events", count(\*) over(partition by discipline\_id) "number\_of\_events" from baseevent) a*

*inner join*

*discipline d*

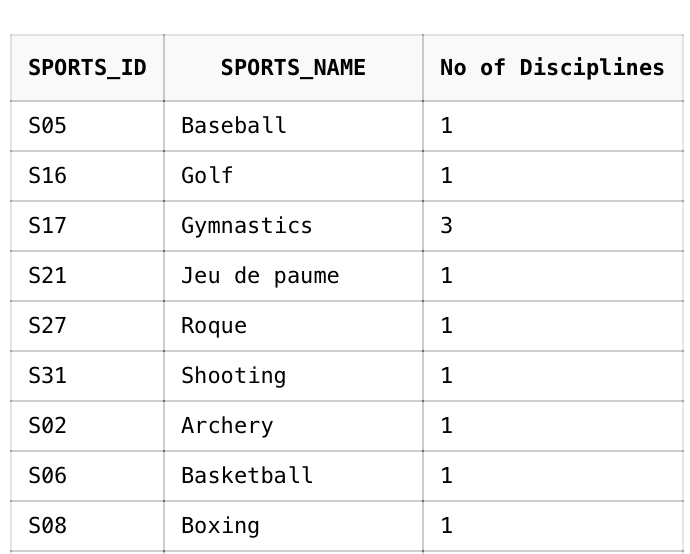
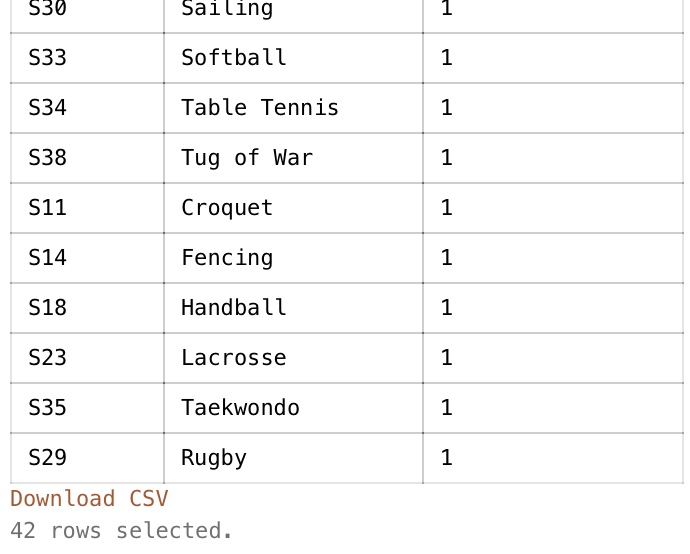
*on a.discipline\_id = d.discipline\_id;*



4.Display the number of Disciplines for each Sport

*select distinct(s.sports\_id),s.sports\_name,count(\*) over(partition by s.sports\_id) "No of Disciplines" from*

*sports s inner join Discipline d on s.sports\_id=d.sports\_id*

**

5.Display countries with both male and female athletes

*select \* from*

*(select distinct(nation\_id), count(\*) over(partition by nation\_id) "count\_of\_male" from athletes where athlete\_gender='Men' ) b1*

*inner join (select distinct(nation\_id),count(\*) over(partition by nation\_id) "count\_of\_female" from athletes where athlete\_gender='Women' ) b2*

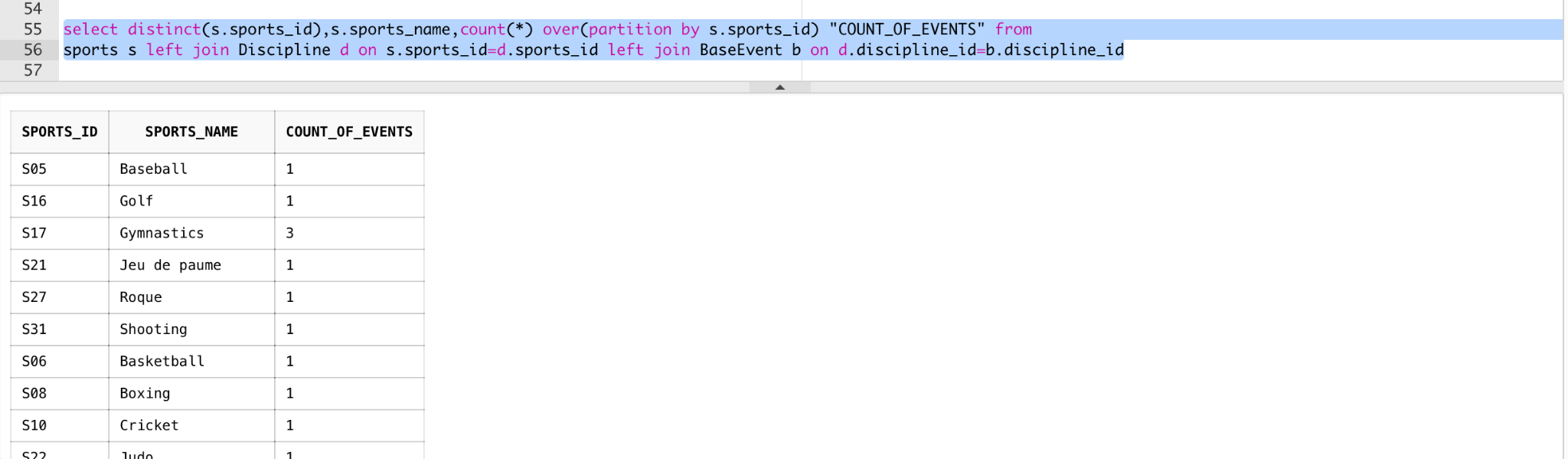
*on b1.nation\_id = b2.nation\_id;*

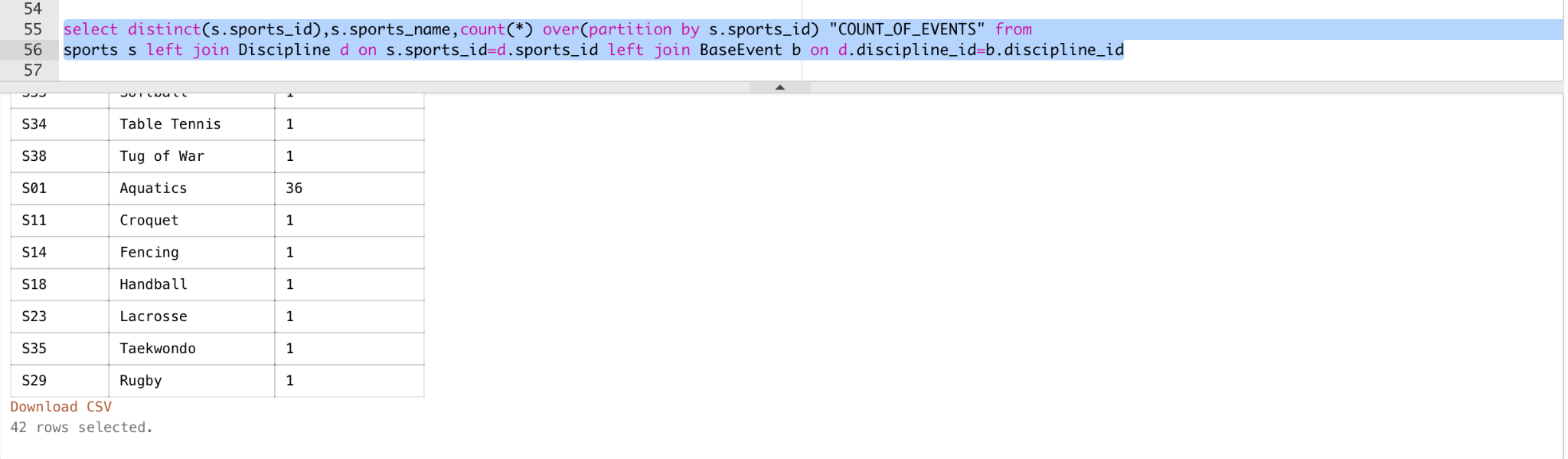


6..Display the number of events for each Sport

*select distinct(s.sports\_id),s.sports\_name,count(\*) over(partition by s.sports\_id) "COUNT\_OF\_EVENTS" from*

*sports s left join Discipline d on s.sports\_id=d.sports\_id left join BaseEvent b on d.discipline\_id=b.discipline\_id*

**

**

7. Diaply all the nations participated in olympic

*select distinct nation\_id from nations;*



8. In which year was the first winter olympic held

*select min(year) from season where season = 'Winter' ;*

9. How many atheletes from each nation participated in 2016 olympic

*select nation\_id, count(\*) as count\_of\_athlete from (*

*select distinct ath.nation\_id, ath.athlete\_id*

*from (select \* from season where year = '2016') s*

*inner join event as ev*

*on s.season\_id = ev.season\_id*

*inner join individual\_schedule as ins*

*on ev.event\_id = ins.event\_id*

*inner join athlete ath*

*on ins.athlete\_id = ath.athlete\_id*

*) group by nation\_id ;*

-- display nation wise count of athlete participated in all the olympics

*select nation\_id, nation\_name, count\_of\_athlete*

*from (select nation\_id, count(\*) as count\_of\_athlete from athelete group by nation\_id) a*

*inner join nations n*

*on a.nation\_id = n.nation\_id ;*

10 . In 2016 olympic summer season, display the count of Gold, silver and Bronze medal won by each athelete in USA.

-> *select year, sport\_id, sport\_name, gender, event\_name, athlete\_name, medal from (*

*select s.year, sp.sport\_id, sp.sport\_name, ev.gender, ev.event\_name, ath.athlete\_name, m.medal*

*from athletes ath*

*inner join medal m*

*on ath.athlete\_id = m.athlete\_id*

*inner join event ev*

*on m.event\_id = ev.event\_id*

*inner join seasons*

*on ev.season\_id = s.season\_id*

*inner join base\_event be*

*on ev.baset\_event\_id = be.base\_event\_id*

*inner join discipline di*

*on be.discipline\_id = di.discipline\_id*

*inner join sports sp*

*on di.sport\_id = sp.sport\_id*

*where s.season = 'Summer' and s.year = '2016' and ath.nation\_id = 'USA'*

*) pivot (count(\*) for medal in ('Gold' gold, 'Silver' silver, 'Bronze' bronze ))*

*order by sport\_name ;*

*11. Display top 5 athletes who won the most medals.*

*Select \* from (*

*Select athlete\_id, cnt\_of\_medals, dense\_rank() over(order by cnt\_of\_medals desc) as rn*

*From (*

*Select ath.athlete\_id, count(\*) as cnt\_of\_medals*

*From Athlete as ath*

*Inner join medal as md*

*On ath.athelte\_id = md.athlete\_id*

*Group by ath.athlete\_id*

*) ) where rn <=5 ;*

*Or*

*Select athlete\_id, cnt\_of\_medals from (*

*Select ath.athlete\_id, count(\*) as cnt\_of\_medals , desnse\_rank() over(order by count(\*) desc) as rn*

*From Athlete as ath*

*Inner join medal as md*

*On ath.athelte\_id = md.athlete\_id*

*Group by ath.athlete\_id*

*) where rn < =5 ;*

*12. Display the top 5 nations who won gold medals in all the Olympic events .*

*Select nation\_id, nation\_name, nationwise\_cnt\_of\_medal from (*

*Select n.nation\_id, n.nation\_name, count(\*) as nationwise\_cnt\_of\_medal,*

*dense\_rank() over(order by count(\*) desc) as rn*

*From nation as n*

*Inner join athlete as ath*

*On n.nation\_id = ath.nation\_id*

*Inner join medal as m*

*On ath.athlete\_id = m.athlete\_id*

*Where m.medal = ‘Gold’*

*Group by n.nation\_id, n.nation\_name*

*) where rn < 5 ;*