```sql
create database olympics;

use olympics;

##Uploading the data

DROP TABLE IF EXISTS OLYMPICS_HISTORY;

CREATE TABLE IF NOT EXISTS OLYMPICS_HISTORY
(
    id      INT,
    name      VARCHAR(255),
    sex       VARCHAR(255),
    age       VARCHAR(255),
    height    VARCHAR(255),
    weight    VARCHAR(255),
    team      VARCHAR(255),
    noc       VARCHAR(255),
    games     VARCHAR(255),
    year      INT NULL,
    season    VARCHAR(255),
    city      VARCHAR(255),
    sport     VARCHAR(255),
    event     VARCHAR(255),
    medal     VARCHAR(255)
);


DROP TABLE IF EXISTS OLYMPICS_HISTORY_NOC_REGIONS;


CREATE TABLE IF NOT EXISTS OLYMPICS_HISTORY_NOC_REGIONS
(
    noc       VARCHAR(10),    -- Adjust the length based on your needs
    region    VARCHAR(100),   -- Adjust the length based on your needs
    notes     VARCHAR(255)    -- Adjust the length based on your needs
);
```

```
select * from olympics_history;
```

```
select * from OLYMPICS_HISTORY_NOC_REGIONS;
```

#1.   How many olympics games have been held?

```
select count(distinct games) as total_olympic_games
   from olympics_history;
```

#2.   List down all Olympics games held so far.

```
select distinct year,season, city from olympics_history
order by year ;
```

#3.   Mention the total no of nations who participated in each olympics game?

```
SELECT games, COUNT(DISTINCT noc) AS total_number_of_countries
FROM olympics_history
GROUP BY games
ORDER BY games;
```

#4.Which year saw the highest and lowest no of countries participating in olympics?

```
-- Query to find the year with the highest number of countries participating
SELECT year, COUNT(noc) AS num_countries
FROM olympics_history
GROUP BY year
ORDER BY num_countries desc
LIMIT 1;
```

```
-- Query to find the year with the lowest number of countries participating
SELECT year, COUNT(noc) AS num_countries
FROM olympics_history
```

GROUP BY year

ORDER BY num_countries ASC

LIMIT 1;

#5 Which nation has participated in all of the olympic games?

SELECT noc, COUNT(year) AS participation_years

FROM olympics_history

GROUP BY noc

HAVING participation_years = (SELECT COUNT(DISTINCT year) FROM olympics_history);

#6.   Identify the sport which was played in all summer olympics.

WITH total_games AS (

   SELECT COUNT(DISTINCT games) AS total_games

   FROM olympics_history

   WHERE season = 'Summer'

),

sport_games AS (

   SELECT sport, COUNT(DISTINCT games) AS sport_game_count

   FROM olympics_history

   WHERE season = 'Summer'

   GROUP BY sport

)

SELECT sport

FROM sport_games

JOIN total_games ON sport_games.sport_game_count = total_games.total_games;

#7.   Which Sports were just played only once in the olympics?

SELECT sport, COUNT(DISTINCT games) AS games_count

FROM olympics_history

GROUP BY sport

HAVING COUNT(DISTINCT games) = 1;

#8.   Fetch the total no of sports played in each olympic games.

SELECT games, COUNT(DISTINCT sport) AS total_no_sports

FROM olympics_history

GROUP BY games;

#9.   Fetch details of the oldest athletes to win a gold medal.

 select oh.name, oh.age , oh.sex, oh.team, oh.sport, oh.medal, oh.year from olympics_history oh

 where oh.medal = "Gold"

 order by oh.age desc;

#10.Find the Ratio of male and female athletes participated in all olympic games

WITH gender_counts AS (

   SELECT

     sex,

     COUNT(1) AS cnt

   FROM

     olympics_history

   GROUP BY

     sex

)

SELECT

   CONCAT('1 : ', ROUND(MAX(cnt) * 1.0 / NULLIF(MIN(cnt), 0), 2)) AS ratio

FROM

   gender_counts;

#11.Fetch the top 5 athletes who have won the most gold medals.

SELECT

   oh.name,

   COUNT(oh.medal) AS count_medal

FROM

```sql
   olympics_history oh
WHERE
   oh.medal = 'Gold'
GROUP BY
   oh.name
ORDER BY
   count_medal DESC
LIMIT 5;
```

#12.Fetch the top 5 athletes who have won the most medals (gold/silver/bronze).
```sql
SELECT
   oh.name,
   COUNT(oh.medal) AS count_medal
FROM
   olympics_history oh
GROUP BY
   oh.name
ORDER BY
   count_medal DESC
LIMIT 5;
```

#13.Fetch the top 5 most successful countries in olympics. Success is defined by no of medals won.
```sql
SELECT
   oh.team,
   COUNT(oh.medal) AS count_medal
FROM
   olympics_history oh
GROUP BY
   oh.team
ORDER BY
```

```
    count_medal DESC
LIMIT 5;
```

#14.List down total gold, silver and broze medals won by each country

```
SELECT
    team,
    COUNT(CASE WHEN medal = 'Gold' THEN 1 END) AS total_gold,
    COUNT(CASE WHEN medal = 'Silver' THEN 1 END) AS total_silver,
    COUNT(CASE WHEN medal = 'Bronze' THEN 1 END) AS total_bronze
FROM
    olympics_history
GROUP BY
    team
ORDER BY
    team DESC;  -- Orders the results by country name in descending order
```

#15.List down total gold, silver and broze medals won by each country corresponding to each olympic games.

```
SELECT
    team,
    games,
    COUNT(CASE WHEN medal = 'Gold' THEN 1 END) AS total_gold,
    COUNT(CASE WHEN medal = 'Silver' THEN 1 END) AS total_silver,
    COUNT(CASE WHEN medal = 'Bronze' THEN 1 END) AS total_bronze
FROM
    olympics_history
GROUP BY
    team, games
ORDER BY
    games, team;
```

#16.Identify which country won the most gold, most silver and most bronze medals in each olympic games.

WITH Medal_Counts AS (

   SELECT

     team,

     games,

     COUNT(CASE WHEN medal = 'Gold' THEN 1 END) AS total_gold,

     COUNT(CASE WHEN medal = 'Silver' THEN 1 END) AS total_silver,

     COUNT(CASE WHEN medal = 'Bronze' THEN 1 END) AS total_bronze

   FROM

     olympics_history

   GROUP BY

     team, games

),

Ranked_Teams AS (

   SELECT

     games,

     team,

     total_gold,

     total_silver,

     total_bronze,

     RANK() OVER (PARTITION BY games ORDER BY total_gold DESC) AS rank_gold,

     RANK() OVER (PARTITION BY games ORDER BY total_silver DESC) AS rank_silver,

     RANK() OVER (PARTITION BY games ORDER BY total_bronze DESC) AS rank_bronze

   FROM

     Medal_Counts

)

SELECT

   games,

   MAX(CASE WHEN rank_gold = 1 THEN team END) AS most_gold_team,

   MAX(CASE WHEN rank_gold = 1 THEN total_gold END) AS most_gold_count,

```sql
    MAX(CASE WHEN rank_silver = 1 THEN team END) AS most_silver_team,

    MAX(CASE WHEN rank_silver = 1 THEN total_silver END) AS most_silver_count,


    MAX(CASE WHEN rank_bronze = 1 THEN team END) AS most_bronze_team,

    MAX(CASE WHEN rank_bronze = 1 THEN total_bronze END) AS most_bronze_count
FROM

    Ranked_Teams
GROUP BY

    games
ORDER BY

    games;
```