

Queries for Data Analytics

1. Display highest-rated book author

```
SELECT B.ISBN, BOOK_AUTHOR, MAX(BOOK_RATING) AS HIGHEST_BOOKRATING
FROM book_rating_225.ratings R
JOIN
book_rating_225.books B ON B.ISBN = R.ISBN
GROUP BY B.ISBN,B.BOOK_AUTHOR;
```

2. Display highest book ratings for each publisher

```
SELECT PUBLISHER,MAX(BOOK_RATING) AS HIGHEST_BOOKRATING
FROM book_rating_225.books B JOIN book_rating_225.ratings R
ON B.ISBN = R.ISBN
GROUP BY PUBLISHER;
```

3. Display number of books that is published in the year 1983

```
SELECT COUNT(ISBN) AS BOOK_COUNT
FROM book_rating_225.books WHERE YEAR_OF_PUBLISH=1983;
```

4. Display the 8+ rating books were published in 2001

```
SELECT B.ISBN, B.book_title, B.book_author,R.book_rating
FROM book_rating_225.books B, book_rating_225.ratings R
WHERE B.ISBN = R.ISBN AND R.book_rating > 8 AND B.year_of_publish = 2001;
```

5. Display average book-ratings for each book

```
SELECT ISBN, round(avg(book_rating),2) AS Average_Rating
FROM book_rating_225.ratings
GROUP BY ISBN
```

6. Display number of books that ages 18 - 25 users read

```
SELECT U.user_id, U.age, count(ISBN) AS Total_Book_Reviewed
FROM book_rating_225.ratings R, book_rating_225.users U
WHERE R.user_id = U.user_id AND U.age >= 18 AND U.age <= 25
GROUP BY U.user_id, U.age
```

7. Display number of ratings of the books in descending order

```
SELECT ISBN, count(book_rating) AS Number_of_Ratings
FROM book_rating_225.ratings
GROUP BY ISBN
ORDER BY count(book_rating) DESC;
```

8. *How many books did the authors publish in the year '2001'?*

```
SELECT book_author, count(ISBN) AS Total_Number_of_Books
FROM book_rating_225.books
WHERE year_of_publish = 2001
GROUP BY book_author;
```

9. *How many publishers published "Jack Canfield" books in the year 1999 and 2000?*

```
select count(distinct(publisher)) as number_of_publishers, book_author, year_of_publish
from book_rating_225.books
where book_author = "Jack Canfield" and year_of_publish IN(1999, 2000)
group by book_author, year_of_publish;
```

10. *How many 7+ratings were published in the year '1995'?*

```
select count(r.isbn) as number_of_books, r.book_rating
from book_rating_225.books as B
join book_rating_225.ratings as r
on B.ISBN = r.ISBN
Where r.book_rating >= 7 and B.year_of_publish = 1995
Group by r.book_rating
Order by number_of_books DESC;
```

11. *Find the number of ratings for each author*

```
SELECT B.book_author, count(R.book_rating) AS Number_of_Ratings
FROM book_rating_225.books B, book_rating_225.ratings R
WHERE B.ISBN = R.ISBN
GROUP BY B.book_author;
```

12. *What Top cities and states do the users who read the books of the highest rated author?*

```
SELECT U.user_id, U.location
FROM book_rating_225.users U, book_rating_225.books B, book_rating_225.ratings R
WHERE B.ISBN = R.ISBN AND R.user_id = U.user_id AND B.book_author =
(SELECT BOOK_AUTHOR FROM book_rating_225.ratings R JOIN
book_rating_225.books B ON B.ISBN = R.ISBN
GROUP BY B.BOOK_AUTHOR
ORDER BY SUM(BOOK_RATING) DESC
LIMIT 1);
```

13. *Display the average book ratings partitioned by publishers?*

```
SELECT B.publisher, AVG(book_rating) OVER( PARTITION BY publisher) AS Avg_Rating
FROM book_rating_225.books B, book_rating_225.ratings R
WHERE B.ISBN = R.ISBN
ORDER BY Avg_Rating DESC;
```

Queries for Creating Tables in Google Cloud SQL Instance

```
CREATE DATABASE book_rating_225;
```

```
CREATE TABLE users (user_id INT, location VARCHAR(500), age INT);
```

```
CREATE TABLE books (ISBN VARCHAR(100), book_title VARCHAR(500), book_author  
VARCHAR(200), year_of_publish INT, publisher VARCHAR(500), Image_URL_S VARCHAR(500),  
Image_URL_M VARCHAR(500), Image_URL_L VARCHAR(500));
```

```
CREATE TABLE ratings (user_id INT, ISBN VARCHAR(100), book_rating INT);
```

Queries for Creating Tables in Google BigQuery

1. Queries for creating fact and dimension tables in Data Warehouse:

```
CREATE TABLE book_rating_225.users (user_id INT, location STRING, age INT);
```

```
CREATE TABLE book_rating_225.books (ISBN STRING, book_title STRING, book_author STRING,  
year_of_publish INT, publisher STRING, Image_URL_S STRING, Image_URL_M  
STRING, Image_URL_L STRING);
```

```
CREATE TABLE book_rating_225.ratings (user_id INT, ISBN STRING, book_rating INT, date_id  
STRING);
```

```
CREATE TABLE book_rating_225.time (date_id STRING, update_date DATE, year INT, month INT,  
date INT);
```

2. Queries for creating staging tables in Data Warehouse:

```
CREATE TABLE staging_dataset.ratings (user_id INT, ISBN STRING, book_rating INT, update_date  
STRING);
```

```
CREATE TABLE staging_dataset.users (user_id INT, location STRING, age INT, update_date STRING);
```

```
CREATE TABLE staging_dataset.books (ISBN STRING, book_title STRING, book_author STRING,  
year_of_publish INT, publisher STRING, Image_URL_S STRING, Image_URL_M  
STRING, Image_URL_L STRING, update_date STRING);
```