A

SQL PROJECT

ON

**MUSIC STORE DATA ANALYSIS**

**Under the guidance of:**

**Mr. Chintan Patel**

**Submitted by :**

Saket Kumar

A close-up of a white background

Description automatically generated

use music\_store;

select \* from genre;

1. **Who is the senior most employee based on job title**?

select concat(first\_name,' ',last\_name),employee\_id from employee order by substring(levels,2,1) desc limit 1;

A screenshot of a computer

Description automatically generated

**2**.**Which countries have the most Invoices**?

select billing\_country,count(\*) as'total invoice' from invoice group by billing\_country order by sum(total) desc;

A screenshot of a table

Description automatically generated

3. What are top 3 values of total invoice?

select \* from invoice order by total desc limit 3;

A screenshot of a computer

Description automatically generated

**4. Which city has the best customers? We would like to throw a promotional Music event there**

select billing\_city,sum(total) from invoice group by billing\_city order by sum(total) desc limit 1;

A screenshot of a computer

Description automatically generated

**5.Who is the best customer?The customer who has spent the most money**

select c.customer\_id,sum(total) from customer as c

inner join invoice as i using(customer\_id)

group by c.customer\_id

order by sum(total) desc limit 1;

A screenshot of a computer

Description automatically generated

6. **Write query to return the email, first name, last name, & Genre of all Rock Music**

**listeners. Return your list ordered alphabetically by email starting with A**

select distinct email,first\_name,last\_name,g.name from customer c

inner join invoice i using(customer\_id)

inner join invoice\_line using(invoice\_id)

inner join track using (track\_id)

inner join genre g using (genre\_id)

where lower(g.name)='rock'

order by email;

A screenshot of a computer

Description automatically generated

**7. Let's invite the artists who have written the most rock music in our dataset. Write a**

**query that returns the Artist name and total track count of the top 10 rock bands**

select a.artist\_id,a.name,count(a.artist\_id) as 't'

from artist a

inner join album2 using(artist\_id)

inner join track using(album\_id)

inner join genre g using(genre\_id) where g.name='Rock'

group by a.artist\_id,a.name

order by t desc limit 10;

A screenshot of a computer

Description automatically generated

**8. Return all the track names that have a song length longer than the average song length.**

**Return the Name and Milliseconds for each track. Order by the song length with the**

**longest songs listed first**

WITH av AS

( SELECT AVG(milliseconds) AS avg\_duration

FROM track)

SELECT name, milliseconds

FROM track

WHERE milliseconds > (

SELECT avg\_duration FROM av

)

ORDER BY milliseconds DESC;

A screenshot of a computer

Description automatically generated

**9.select most expensive album**

select name,media\_type\_id,unit\_price

from track

order by unit\_price desc

limit 1;

A screenshot of a computer

Description automatically generated

**10.select top 10 cheapest album**

select name,media\_type\_id,unit\_price

from track

order by unit\_price limit 10;

A screenshot of a computer

Description automatically generated

**11. COUNTRYWISE MOST SPENDING CUSTOMERS**

with x as(

select c.customer\_id,first\_name,last\_name,billing\_country,sum(total) as s,

row\_number()over(partition by billing\_country order by sum(total) desc) as r

from customer c join invoice i

using(customer\_id)

group by 1,2,3,4 )

select \* from x where r<=1;

A screenshot of a computer

Description automatically generated

**12. country top selling music genres**

with x as(

select count(quantity)as ‘total\_genre\_count’,

country,genre.name,track.genre\_id,

row\_number()over(partition by country order by count(quantity) desc) as r

from invoice

join customer using(customer\_id)

join invoice\_line using(invoice\_id)

join track using(track\_id)

join genre using(genre\_id)

group by 2,3,4

)

select \* from x where r<=1;

A screenshot of a computer

Description automatically generated

**13.list all artists whose name start with vowels**

SELECT name

FROM artist

WHERE left(name,1) in ('a','e','i','o','u','A','E','I','O','U');

A screenshot of a computer

Description automatically generated

**14. select top 10 customers according to their expenditure**

select c.customer\_id,concat(first\_name," ",last\_name) as full\_name,

round(sum(total),2) from customer as c

inner join invoice as i using(customer\_id)

group by 1,2

order by sum(total) desc limit 10;

A screenshot of a computer

Description automatically generated

**15.select all names of all employees and replace a with x.**

select replace(first\_name,'a','x')

from employee;

A screenshot of a computer

Description automatically generated

**16. Let's invite the top artist who have written the most rock music in our dataset. Write a**

**query that returns the Artist name and total track count .**

select a.artist\_id,a.name,count(a.artist\_id) as 't'

from artist a

inner join album2 using(artist\_id)

inner join track using(album\_id)

inner join genre g using(genre\_id) where g.name='Rock'

group by a.artist\_id,a.name

order by t desc limit 1;

A screenshot of a computer

Description automatically generated

**17.select dob of employee end with ‘a’**

select birthdate,first\_name from employee

where first\_name like '%A';

A screenshot of a computer

Description automatically generated

**18. select customer name,quantity,total bill**

select first\_name,sum(total),sum(quantity)

from customer join invoice using(customer\_id)

inner join invoice\_line using(invoice\_id)

group by 1;

A screenshot of a computer

Description automatically generated

**19. select bill of 2nd top customer**

select first\_name,city,total

from customer join invoice using(customer\_id)

order by total desc limit 1 offset 1;

A screenshot of a computer

Description automatically generated

**20. Write query to return the email, first name, last name, & Genre of all Rock Music**

**Listeners,order by first name**

select distinct email,first\_name,last\_name,g.name from customer c

inner join invoice i using(customer\_id)

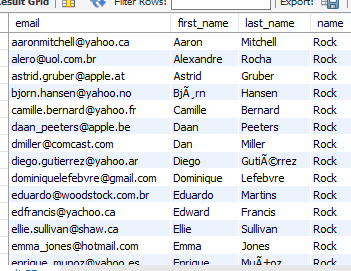
inner join invoice\_line using(invoice\_id)

inner join track using (track\_id)

inner join genre g using (genre\_id)

where lower(g.name)='rock'

order by first\_name;



**21. select customer name,quantity,total bill**

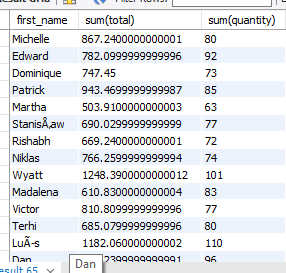
select first\_name,sum(total),sum(quantity)

from customer

join invoice using(customer\_id)

inner join invoice\_line using(invoice\_id)

group by 1;



**22.Total bill with customer name**

select first\_name,sum(total)

from invoice join customer

using(customer\_id) group by 1;

A screenshot of a computer

Description automatically generated

**23.select countries with most bill**

select billing\_country, round(sum(total),2) as total\_bill

from invoice group by 1;

A screenshot of a data

Description automatically generated

**24. CALCULATE STATWISE BILL PER COUNTRY**

select distinct billing\_country,billing\_state,

round((sum(total)over(partition by billing\_country order by billing\_state)),2) as total\_bill

from invoice;

A screenshot of a computer screen

Description automatically generated

**25.select max billing state from each country**

with x as (select billing\_country,billing\_state,round(sum(total),2)as total\_bill,

row\_number()over(partition by billing\_country order by sum(total) desc) as r from invoice group by 1,2)

select \* from x where r<=1;

A screenshot of a computer

Description automatically generated