



SQL Editor

```

1 SELECT * FROM Album;
2
3 -- Q1. Who is the senior most employee based on joining date?
4
5 SELECT * FROM Employee ORDER BY HireDate LIMIT 1;
6
7 -- Q2. Which counties have the most Invoices?
8
9 SELECT Count(*) as Invoice_count, BillingCountry FROM Invoice GROUP BY BillingCountry
10 ORDER BY Invoice_count DESC;
11
12 -- Q3. What are top 3 values of total invoices?
13
14 SELECT InvoiceId, total FROM Invoice ORDER BY total DESC LIMIT 3;
15
16 -- Q4. Which city has the best customers?
17 --- We would like to throw a promotional Music Festival in the city we made the most money.
18 --- Write a query that returns one city that has the highest sum of invoice totals.
19 --- Return both the city name & sum of all invoice totals.
20
21 SELECT SUM(total) as invoice_total, BillingCity FROM invoice GROUP BY BillingCity
22 ORDER BY invoice_total DESC;
23
24 -- Q5. Who is the best customer?
25 --- The customer who has spent the most money will be declared the best customer.

```

SQL Project by  
Shiva Sagar H M

	AlbumId	Title	ArtistId
1	1	For Those About To Rock We Salute You	1
2	2	Balls to the Wall	2
3	3	Restless and Wild	2
4	4	Let There Be Rock	1
5	5	Big Ones	3
6	6	Jagged Little Pill	4
7	7	Facelift	5
8	8	Warner 25 Anos	6
9	9	Plays Metallica By Four Cellos	7
10	10	Audioslave	8

Name	Type	Schema
Tables (11)		
Album		CREATE TABLE [Album] (
Artist		CREATE TABLE [Artist] (
Customer		CREATE TABLE [Customer]
Employee		CREATE TABLE [Employee]
Genre		CREATE TABLE [Genre] (
Invoice		CREATE TABLE [Invoice] (
InvoiceLine		CREATE TABLE [InvoiceLi
MediaType		CREATE TABLE [MediaTyp
Playlist		CREATE TABLE [Playlist] (
PlaylistTrack		CREATE TABLE [PlaylistTr
Track		CREATE TABLE [Track] (
Indices (10)		
IFK_AlbumArtistId		CREATE INDEX [IFK_Albu
IFK_CustomerSupportRepId		CREATE INDEX [IFK_Cust
IFK_EmployeeReportsTo		CREATE INDEX [IFK_Empl
IFK_InvoiceCustomerId		CREATE INDEX [IFK_Invo
IFK_InvoiceLineInvoiceId		CREATE INDEX [IFK_Invo
IFK_InvoiceLineTrackId		CREATE INDEX [IFK_Invo
IFK_PlaylistTrackTrackId		CREATE INDEX [IFK_Play
IFK_TrackAlbumId		CREATE INDEX [IFK_Trac
IFK_TrackGenreId		CREATE INDEX [IFK_Trac
IFK_TrackMediaTypeId		CREATE INDEX [IFK_Trac
Views (0)		
Triggers (0)		



S...

```
1 SELECT * FROM Album;
2
3 -- Q1. Who is the senior most employee based on joining date?
4
5 SELECT * FROM Employee ORDER BY HireDate LIMIT 1;
6
7 -- Q2. Which counties have the most Invoices?
8
9 SELECT Count(*) as Invoice_count, BillingCountry FROM Invoice GROUP BY BillingCountry
10 ORDER BY Invoice_count DESC;
11
12 -- Q3. What are top 3 values of total invoices?
13
14 SELECT InvoiceId, total FROM Invoice ORDER BY total DESC LIMIT 3;
15
16 -- Q4. Which city has the best customers?
17 --- We would like to throw a promotional Music Festival in the city we made the most money.
18 --- Write a query that returns one city that has the highest sum of invoice totals.
19 --- Return both the city name & sum of all invoice totals.
20
21 SELECT SUM(total) as invoice_total, BillingCity FROM invoice GROUP BY BillingCity
22 ORDER BY invoice_total DESC;
23
24 -- Q5. Who is the best customer?
25 --- The customer who has spent the most money will be declared the best customer.
```

SQL Project by  
Shiva Sagar H M

	EmployeeId	LastName	FirstName	Title	ReportsTo	BirthDate	HireDate	Address	City	S
1	3	Peacock	Jane	Sales Support Agent	2	1973-08-29 00:00:00	2002-04-01 00:00:00	1111 6 Ave SW	Calgary	A

Name	Type	Schema
Tables (11)		
Album	CREATE TABLE [Album] (	
Artist	CREATE TABLE [Artist] (	
Customer	CREATE TABLE [Customer]	
Employee	CREATE TABLE [Employee]	
Genre	CREATE TABLE [Genre] (	
Invoice	CREATE TABLE [Invoice] (	
InvoiceLine	CREATE TABLE [InvoiceLi	
MediaType	CREATE TABLE [MediaTyp	
Playlist	CREATE TABLE [Playlist] (	
PlaylistTrack	CREATE TABLE [PlaylistTr	
Track	CREATE TABLE [Track] (	
Indices (10)		
IFK_AlbumArtistId	CREATE INDEX [IFK_Albu	
IFK_CustomerSupportRepId	CREATE INDEX [IFK_Cust	
IFK_EmployeeReportsTo	CREATE INDEX [IFK_Empl	
IFK_InvoiceCustomerId	CREATE INDEX [IFK_Invoi	
IFK_InvoiceLineInvoiceId	CREATE INDEX [IFK_Invoi	
IFK_InvoiceLineTrackId	CREATE INDEX [IFK_Invoi	
IFK_PlaylistTrackTrackId	CREATE INDEX [IFK_Playl	
IFK_TrackAlbumId	CREATE INDEX [IFK_Track	
IFK_TrackGenreId	CREATE INDEX [IFK_Track	
IFK_TrackMediaTypeId	CREATE INDEX [IFK_Track	
Views (0)		
Triggers (0)		





× S...

```

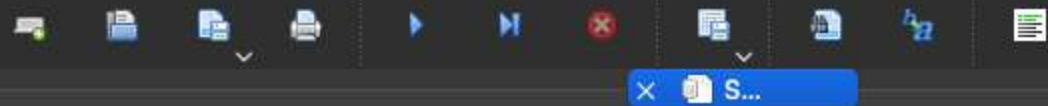
1  SELECT * FROM Album;
2
3  -- Q1. Who is the senior most employee based on joining date?
4
5  SELECT * FROM Employee ORDER BY HireDate LIMIT 1;
6
7  -- Q2. Which counties have the most Invoices?
8
9  SELECT Count(*) as Invoice_count, BillingCountry FROM Invoice GROUP BY BillingCountry
10 ORDER BY Invoice_count DESC;
11
12 -- Q3. What are top 3 values of total invoices?
13
14 SELECT InvoiceId, total FROM Invoice ORDER BY total DESC LIMIT 3;
15
16 -- Q4. Which city has the best customers?
17 --- We would like to throw a promotional Music Festival in the city we made the most money.
18 --- Write a query that returns one city that has the highest sum of invoice totals.
19 --- Return both the city name & sum of all invoice totals.
20
21 SELECT SUM(total) as invoice_total, BillingCity FROM invoice GROUP BY BillingCity
22 ORDER BY invoice_total DESC;
23
24 -- Q5. Who is the best customer?
25 --- The customer who has spent the most money will be declared the best customer.

```

SQL Project by  
Shiva Sagar H M

	Invoice_count	BillingCountry
1	91	USA
2	56	Canada
3	35	France
4	35	Brazil
5	28	Germany
6	21	United Kingdom
7	14	Portugal
8	14	Czech Republic
9	13	India
10	7	Sweden

Name	Type	Schema
Tables (11)		
Album	CREATE TABLE [Album] (	
Artist	CREATE TABLE [Artist] (	
Customer	CREATE TABLE [Customer]	
Employee	CREATE TABLE [Employee]	
Genre	CREATE TABLE [Genre] (	
Invoice	CREATE TABLE [Invoice] (	
InvoiceLine	CREATE TABLE [InvoiceLi	
MediaType	CREATE TABLE [MediaTyp	
Playlist	CREATE TABLE [Playlist] (	
PlaylistTrack	CREATE TABLE [PlaylistTr	
Track	CREATE TABLE [Track] (	
Indices (10)		
IFK_AlbumArtistId	CREATE INDEX [IFK_Albu	
IFK_CustomerSupportRepId	CREATE INDEX [IFK_Cust	
IFK_EmployeeReportsTo	CREATE INDEX [IFK_Empl	
IFK_InvoiceCustomerId	CREATE INDEX [IFK_Invo	
IFK_InvoiceLineInvoiceId	CREATE INDEX [IFK_Invo	
IFK_InvoiceLineTrackId	CREATE INDEX [IFK_Invo	
IFK_PlaylistTrackTrackId	CREATE INDEX [IFK_Playl	
IFK_TrackAlbumId	CREATE INDEX [IFK_Trac	
IFK_TrackGenreId	CREATE INDEX [IFK_Trac	
IFK_TrackMediaTypeId	CREATE INDEX [IFK_Trac	
Views (0)		
Triggers (0)		



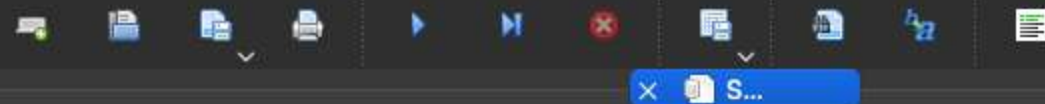
```
10 ORDER BY Invoice_count DESC;
11
12 -- Q3. What are top 3 values of total invoices?
13
14 SELECT InvoiceId, total FROM Invoice ORDER BY total DESC LIMIT 3;
15
16 -- Q4. Which city has the best customers?
17 --- We would like to throw a promotional Music Festival in the city we made the most money.
18 --- Write a query that returns one city that has the highest sum of invoice totals.
19 --- Return both the city name & sum of all invoice totals.
20
21 SELECT SUM(total) as invoice_total, BillingCity FROM invoice GROUP BY BillingCity
22 ORDER BY invoice_total DESC;
23
24 -- Q5. Who is the best customer?
25 --- The customer who has spent the most money will be declared the best customer.
26 --- Write a query that returns the person who has spent the most money.
27
28 SELECT c.CustomerId, c.FirstName, c.LastName, SUM(i.total) as Total
29 FROM Customer c JOIN Invoice i ON c.CustomerId = i.CustomerId
30 GROUP BY c.CustomerId ORDER BY Total DESC LIMIT 1;
31
32 -- Q6. Write a query to return the email, first name, last name, & Genre of all Rock Music listeners.
33 --- Return your list ordered alphabetically by email starting with A.
34
```

SQL Project by  
Shiva Sagar H M

	InvoiceId	Total
1	404	25.86
2	299	23.86
3	96	21.86

Name	Type	Schema
Tables (11)		
Album	CREATE TABLE [Album] (	
Artist	CREATE TABLE [Artist] (	
Customer	CREATE TABLE [Customer]	
Employee	CREATE TABLE [Employee]	
Genre	CREATE TABLE [Genre] (	
Invoice	CREATE TABLE [Invoice] (	
InvoiceLine	CREATE TABLE [InvoiceLi	
MediaType	CREATE TABLE [MediaTyp	
Playlist	CREATE TABLE [Playlist] (	
PlaylistTrack	CREATE TABLE [PlaylistTr	
Track	CREATE TABLE [Track] (	
Indices (10)		
IFK_AlbumArtistId	CREATE INDEX [IFK_Albu	
IFK_CustomerSupportRepId	CREATE INDEX [IFK_Cust	
IFK_EmployeeReportsTo	CREATE INDEX [IFK_Empl	
IFK_InvoiceCustomerId	CREATE INDEX [IFK_Invo	
IFK_InvoiceLineInvoiceId	CREATE INDEX [IFK_Invo	
IFK_InvoiceLineTrackId	CREATE INDEX [IFK_Invo	
IFK_PlaylistTrackTrackId	CREATE INDEX [IFK_Play	
IFK_TrackAlbumId	CREATE INDEX [IFK_Trac	
IFK_TrackGenreId	CREATE INDEX [IFK_Trac	
IFK_TrackMediaTypeId	CREATE INDEX [IFK_Trac	
Views (0)		
Triggers (0)		





```
10 ORDER BY Invoice_count DESC;
11
12 -- Q3. What are top 3 values of total invoices?
13
14 SELECT InvoiceId, total FROM Invoice ORDER BY total DESC LIMIT 3;
15
16 -- Q4. Which city has the best customers?
17 --- We would like to throw a promotional Music Festival in the city we made the most money.
18 --- Write a query that returns one city that has the highest sum of invoice totals.
19 --- Return both the city name & sum of all invoice totals.
20
21 SELECT SUM(total) as invoice_total, BillingCity FROM invoice GROUP BY BillingCity
22 ORDER BY invoice_total DESC;
23
24 -- Q5. Who is the best customer?
25 --- The customer who has spent the most money will be declared the best customer.
26 --- Write a query that returns the person who has spent the most money.
27
28 SELECT c.CustomerId, c.FirstName, c.LastName, SUM(i.total) as Total
29 FROM Customer c JOIN Invoice i ON c.CustomerId = i.CustomerId
30 GROUP BY c.CustomerId ORDER BY Total DESC LIMIT 1;
31
32 -- Q6. Write a query to return the email, first name, last name, & Genre of all Rock Music listeners.
33 --- Return your list ordered alphabetically by email starting with A.
34
```

SQL Project by  
Shiva Sagar H M

	Invoice_total	BillingCity
1	90.24	Prague
2	77.24	Paris
3	77.24	Mountain View
4	75.24	Berlin
5	75.24	São Paulo
6	75.24	London
7	47.62	Fort Worth
8	46.62	Santiago
9	45.62	Dublin
10	45.62	Budapest

Name	Type	Schema
Tables (11)		
Album	CREATE TABLE [Album] (	
Artist	CREATE TABLE [Artist] ( [	
Customer	CREATE TABLE [Customer]	
Employee	CREATE TABLE [Employee]	
Genre	CREATE TABLE [Genre] (	
Invoice	CREATE TABLE [Invoice] (	
InvoiceLine	CREATE TABLE [InvoiceLi	
MediaType	CREATE TABLE [MediaTyp	
Playlist	CREATE TABLE [Playlist] (	
PlaylistTrack	CREATE TABLE [PlaylistTr	
Track	CREATE TABLE [Track] ( [	
Indices (10)		
IFK_AlbumArtistId	CREATE INDEX [IFK_Albu	
IFK_CustomerSupportRepId	CREATE INDEX [IFK_Cust	
IFK_EmployeeReportsTo	CREATE INDEX [IFK_Empl	
IFK_InvoiceCustomerId	CREATE INDEX [IFK_Invol	
IFK_InvoiceLineInvoiceId	CREATE INDEX [IFK_Invol	
IFK_InvoiceLineTrackId	CREATE INDEX [IFK_Invol	
IFK_PlaylistTrackTrackId	CREATE INDEX [IFK_Playl	
IFK_TrackAlbumId	CREATE INDEX [IFK_Trac	
IFK_TrackGenreId	CREATE INDEX [IFK_Trac	
IFK_TrackMediaTypeId	CREATE INDEX [IFK_Trac	
Views (0)		
Triggers (0)		



S...

```
18 --- Write a query that returns one city that has the highest sum of invoice totals.
19 --- Return both the city name & sum of all invoice totals.
20
21 SELECT SUM(total) as invoice_total, BillingCity FROM invoice GROUP BY BillingCity
22 ORDER BY invoice_total DESC;
```

```
23
24 -- Q5. Who is the best customer?
25 --- The customer who has spent the most money will be declared the best customer.
26 --- Write a query that returns the person who has spent the most money.
```

```
27
28 SELECT c.CustomerId, c.FirstName, c.LastName, SUM(i.total) as Total
29 FROM Customer c JOIN Invoice i ON c.CustomerId = i.CustomerId
30 GROUP BY c.CustomerId ORDER BY Total DESC LIMIT 1;
```

```
31
32 -- Q6. Write a query to return the email, first name, last name, & Genre of all Rock Music listeners.
33 --- Return your list ordered alphabetically by email starting with A.
```

```
34
35 SELECT DISTINCT c.Email, c.FirstName, c.LastName, g.Name
36 FROM Customer c
37 JOIN Invoice i ON i.CustomerId = c.CustomerId
38 JOIN InvoiceLine il ON il.InvoiceId = i.InvoiceId
39 JOIN Track t ON t.TrackId = il.TrackId
40 JOIN Genre g ON g.GenreId = t.GenreId
41 WHERE g.Name LIKE 'Rock'
42 ORDER BY Email;
```

SQL Project by  
Shiva Sagar H M

	CustomerId	FirstName	LastName	Total
1	6	Helena	Holý	49.62

Name	Type	Schema
Tables (11)		
Album	CREATE TABLE [Album] (	
Artist	CREATE TABLE [Artist] (	
Customer	CREATE TABLE [Customer]	
Employee	CREATE TABLE [Employee]	
Genre	CREATE TABLE [Genre] (	
Invoice	CREATE TABLE [Invoice] (	
InvoiceLine	CREATE TABLE [InvoiceLi	
MediaType	CREATE TABLE [MediaTyp	
Playlist	CREATE TABLE [Playlist] (	
PlaylistTrack	CREATE TABLE [PlaylistTr	
Track	CREATE TABLE [Track] (	
Indices (10)		
IFK_AlbumArtistId	CREATE INDEX [IFK_Albu	
IFK_CustomerSupportRepId	CREATE INDEX [IFK_Cust	
IFK_EmployeeReportsTo	CREATE INDEX [IFK_Empl	
IFK_InvoiceCustomerId	CREATE INDEX [IFK_Invo	
IFK_InvoiceLineInvoiceId	CREATE INDEX [IFK_Invo	
IFK_InvoiceLineTrackId	CREATE INDEX [IFK_Invo	
IFK_PlaylistTrackTrackId	CREATE INDEX [IFK_Play	
IFK_TrackAlbumId	CREATE INDEX [IFK_Trac	
IFK_TrackGenreId	CREATE INDEX [IFK_Trac	
IFK_TrackMediaTypeId	CREATE INDEX [IFK_Trac	
Views (0)		
Triggers (0)		





x S...

```
29 FROM Customer c JOIN Invoice i ON c.CustomerId = i.CustomerId
30 GROUP BY c.CustomerId ORDER BY Total DESC LIMIT 1;
```

31

32 -- Q8. Write a query to return the email, first name, last name, & Genre of all Rock Music listeners.

33 -- Return your list ordered alphabetically by email starting with A.

```
34
35 SELECT DISTINCT c.Email, c.FirstName, c.LastName, g.Name
36 FROM Customer c
37 JOIN Invoice i ON i.CustomerId = c.CustomerId
38 JOIN InvoiceLine il ON il.InvoiceId = i.InvoiceId
39 JOIN Track t ON t.TrackId = il.TrackId
40 JOIN Genre g ON g.GenreId = t.GenreId
41 WHERE g.Name LIKE 'Rock'
42 ORDER BY Email;
```

43

44 -- Q7. Let's invite the artist who have written the most rock music in our dataset.

45 -- Write a query that returns the Artist name and total track count of the top 10 rock bands.

```
46
47 SELECT a.ArtistId, a.Name, COUNT(a.ArtistId) AS Total_Number_Of_Songs
48 FROM Track t
49 JOIN Album al ON al.AlbumId = t.AlbumId
50 JOIN Artist a ON a.ArtistId = al.ArtistId
51 JOIN Genre g ON g.GenreId = t.GenreId
52 WHERE g.Name LIKE 'Rock'
53 GROUP BY a.ArtistId
```

SQL Project by  
Shiva Sagar H M

	Email	FirstName	LastName	Name
1	aaronmitchell@yahoo.ca	Aaron	Mitchell	Rock
2	alero@uol.com.br	Alexandre	Rocha	Rock
3	astrid.gruber@apple.at	Astrid	Gruber	Rock
4	bjorn.hansen@yahoo.no	Bjørn	Hansen	Rock
5	camille.bernard@yahoo.fr	Camille	Bernard	Rock
6	daan_peeters@apple.be	Daan	Peeters	Rock
7	diego.gutierrez@yahoo.ar	Diego	Gutiérrez	Rock
8	dmiller@comcast.com	Dan	Miller	Rock
9	dominiquelefebvre@gmail.com	Dominique	Lefebvre	Rock
10	edfrancis@yachoo.ca	Edward	Francis	Rock

Name	Type	Schema
Tables (11)		
Album	CREATE TABLE [Album] (	
Artist	CREATE TABLE [Artist] ( [	
Customer	CREATE TABLE [Customer]	
Employee	CREATE TABLE [Employee]	
Genre	CREATE TABLE [Genre] (	
Invoice	CREATE TABLE [Invoice] (	
InvoiceLine	CREATE TABLE [InvoiceLi	
MediaType	CREATE TABLE [MediaTyp	
Playlist	CREATE TABLE [Playlist] (	
PlaylistTrack	CREATE TABLE [PlaylistTr	
Track	CREATE TABLE [Track] ( [	
Indices (10)		
IFK_AlbumArtistId	CREATE INDEX [IFK_Albu	
IFK_CustomerSupportRepId	CREATE INDEX [IFK_Custu	
IFK_EmployeeReportsTo	CREATE INDEX [IFK_Empl	
IFK_InvoiceCustomerId	CREATE INDEX [IFK_Invol	
IFK_InvoiceLineInvoiceId	CREATE INDEX [IFK_Invol	
IFK_InvoiceLineTrackId	CREATE INDEX [IFK_Invol	
IFK_PlaylistTrackTrackId	CREATE INDEX [IFK_Playl	
IFK_TrackAlbumId	CREATE INDEX [IFK_Track	
IFK_TrackGenreId	CREATE INDEX [IFK_Track	
IFK_TrackMediaTypeId	CREATE INDEX [IFK_Track	
Views (0)		
Triggers (0)		



X S...

```

36 FROM Customer c
37 JOIN Invoice i ON i.CustomerId = c.CustomerId
38 JOIN InvoiceLine il ON il.InvoiceId = i.InvoiceId
39 JOIN Track t ON t.TrackId = il.TrackId
40 JOIN Genre g ON g.GenreId = t.GenreId
41 WHERE g.Name LIKE 'Rock'
42 ORDER BY Email;

```

-- Q7. Let's invite the artist 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.

```

47 SELECT a.ArtistId, a.Name, COUNT(a.ArtistId) AS Total_Number_Of_Songs
48 FROM Track t
49 JOIN Album al ON al.AlbumId = t.AlbumId
50 JOIN Artist a ON a.ArtistId = al.ArtistId
51 JOIN Genre g ON g.GenreId = t.GenreId
52 WHERE g.Name LIKE 'Rock'
53 GROUP BY a.ArtistId
54 ORDER BY Total_Number_Of_Songs DESC
55 LIMIT 10;

```

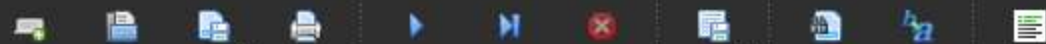
SQL Project by  
Shiva Sagar H M

-- Q8. 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.

	ArtistId	Name	Total_Number_Of_Songs
1	22	Led Zeppelin	114
2	150	U2	112
3	58	Deep Purple	92
4	90	Iron Maiden	81
5	118	Pearl Jam	54
6	152	Van Halen	52
7	51	Queen	45
8	142	The Rolling Stones	41
9	76	Creedence Clearwater Revival	40
10	52	Kiss	35

Name	Type	Schema
Tables (11)		
Album	CREATE TABLE [Album] (	
Artist	CREATE TABLE [Artist] (	
Customer	CREATE TABLE [Customer]	
Employee	CREATE TABLE [Employee]	
Genre	CREATE TABLE [Genre] (	
Invoice	CREATE TABLE [Invoice] (	
InvoiceLine	CREATE TABLE [InvoiceLi]	
MediaType	CREATE TABLE [MediaTyp	
Playlist	CREATE TABLE [Playlist] (	
PlaylistTrack	CREATE TABLE [PlaylistTr	
Track	CREATE TABLE [Track] (	
Indices (10)		
IFK_AlbumArtistId	CREATE INDEX [IFK_Albu	
IFK_CustomerSupportRepId	CREATE INDEX [IFK_Cust	
IFK_EmployeeReportsTo	CREATE INDEX [IFK_Empl	
IFK_InvoiceCustomerId	CREATE INDEX [IFK_Invo	
IFK_InvoiceLineInvoiceId	CREATE INDEX [IFK_Invo	
IFK_InvoiceLineTrackId	CREATE INDEX [IFK_Invo	
IFK_PlaylistTrackTrackId	CREATE INDEX [IFK_Playl	
IFK_TrackAlbumId	CREATE INDEX [IFK_Trac	
IFK_TrackGenreId	CREATE INDEX [IFK_Trac	
IFK_TrackMediaTypeId	CREATE INDEX [IFK_Trac	
Views (0)		
Triggers (0)		





× S...

```
39 JOIN Track t ON t.TrackId = i1.TrackId
40 JOIN Genre g ON g.GenreId = t.GenreId
41 WHERE g.Name LIKE 'Rock'
42 ORDER BY Email;
```

```
44 -- Q7. Let's invite the artist who have written the most rock music in our dataset.
45 -- Write a query that returns the Artist name and total track count of the top 10 rock bands.
```

```
47 SELECT a.ArtistId, a.Name, COUNT(a.ArtistId) AS Total_Number_Of_Songs
48 FROM Track t
49 JOIN Album al ON al.AlbumId = t.AlbumId
50 JOIN Artist a ON a.ArtistId = al.ArtistId
51 JOIN Genre g ON g.GenreId = t.GenreId
52 WHERE g.Name LIKE 'Rock'
53 GROUP BY a.ArtistId
54 ORDER BY Total_Number_Of_Songs DESC
55 LIMIT 10;
```

SQL Project by  
Shiva Sagar H M

```
57 -- Q8. Return all the track names that have a song length longer than the average song length.
58 -- Return the Name and Milliseconds for each track.
59 -- Order by the song length with the longest songs listed first.
```

```
61 SELECT Name, Milliseconds FROM Track
62 WHERE Milliseconds > ( SELECT AVG(Milliseconds) AS Average_Track_Length FROM Track )
63 ORDER BY Milliseconds DESC;
```

	Name	Milliseconds
1	Occupation / Precipice	5286953
2	Through a Looking Glass	5088838
3	Greetings from Earth, Pt. 1	2960293
4	The Man With Nine Lives	2956998
5	Battlestar Galactica, Pt. 2	2956081
6	Battlestar Galactica, Pt. 1	2952702
7	Murder On the Rising Star	2935894
8	Battlestar Galactica, Pt. 3	2927802
9	Take the Celestra	2927677
10	Fire In Space	2926593

Name	Type	Schema
Tables (11)		
Album		CREATE TABLE [Album] (
Artist		CREATE TABLE [Artist] (
Customer		CREATE TABLE [Customer]
Employee		CREATE TABLE [Employee]
Genre		CREATE TABLE [Genre] (
Invoice		CREATE TABLE [Invoice] (
InvoiceLine		CREATE TABLE [InvoiceLi
MediaType		CREATE TABLE [MediaTyp
Playlist		CREATE TABLE [Playlist] (
PlaylistTrack		CREATE TABLE [PlaylistTr
Track		CREATE TABLE [Track] (
Indices (10)		
IFK_AlbumArtistId		CREATE INDEX [IFK_Albu
IFK_CustomerSupportRepId		CREATE INDEX [IFK_Cust
IFK_EmployeeReportsTo		CREATE INDEX [IFK_Empl
IFK_InvoiceCustomerId		CREATE INDEX [IFK_Invo
IFK_InvoiceLineInvoiceId		CREATE INDEX [IFK_Invo
IFK_InvoiceLineTrackId		CREATE INDEX [IFK_Invo
IFK_PlaylistTrackTrackId		CREATE INDEX [IFK_Play
IFK_TrackAlbumId		CREATE INDEX [IFK_Trac
IFK_TrackGenreId		CREATE INDEX [IFK_Trac
IFK_TrackMediaTypeId		CREATE INDEX [IFK_Trac
Views (0)		
Triggers (0)		



x S...

67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89  
90Thank You :DSQL Project by  
Shiva Sagar H M

Name	Type	Schema
Tables (11)		
Album		CREATE TABLE [Album] (
Artist		CREATE TABLE [Artist] (
Customer		CREATE TABLE [Customer]
Employee		CREATE TABLE [Employee]
Genre		CREATE TABLE [Genre] (
Invoice		CREATE TABLE [Invoice] (
InvoiceLine		CREATE TABLE [InvoiceLi
MediaType		CREATE TABLE [MediaTyp
Playlist		CREATE TABLE [Playlist] (
PlaylistTrack		CREATE TABLE [PlaylistTr
Track		CREATE TABLE [Track] (
Indices (10)		
IFK_AlbumArtistId		CREATE INDEX [IFK_Albu
IFK_CustomerSupportRepId		CREATE INDEX [IFK_Cust
IFK_EmployeeReportsTo		CREATE INDEX [IFK_Empl
IFK_InvoiceCustomerId		CREATE INDEX [IFK_Invoi
IFK_InvoiceLineInvoiceId		CREATE INDEX [IFK_Invoi
IFK_InvoiceLineTrackId		CREATE INDEX [IFK_Invoi
IFK_PlaylistTrackTrackId		CREATE INDEX [IFK_Playl
IFK_TrackAlbumId		CREATE INDEX [IFK_Track
IFK_TrackGenreId		CREATE INDEX [IFK_Track
IFK_TrackMediaTypeId		CREATE INDEX [IFK_Track
Views (0)		
Triggers (0)		