

```

1  -- Basic Requirements
2  -----
3  -- Which tracks appeared in the most playlists? how many playlist did they appear in?
4  SELECT * FROM playlist_track;
5  SELECT * FROM tracks;
6
7  SELECT playlist_track.TrackId AS 'Track ID', tracks.Name AS 'Track Name',
8  count(playlist_track.PlaylistId) AS 'Count of Playlist'
9  FROM playlist_track
10 INNER JOIN tracks
11     ON playlist_track.TrackId = tracks.TrackId
12 GROUP BY playlist_track.TrackId
13 ORDER BY 3 DESC;
14
15 -- Which track generated the most revenue? which album? which genre?
16
17 SELECT * FROM invoice_items;
18
19 /* Revenue per Track */
20 SELECT invoice_items.TrackId AS 'Track ID', tracks.Name AS 'Track Name', tracks.UnitPrice AS 'Unit
21 Price', count(invoice_items.UnitPrice) AS 'Count of Sales per Track'
22     , sum(invoice_items.UnitPrice) AS 'Revenue per Track'
23 FROM invoice_items
24 INNER JOIN tracks
25     ON invoice_items.TrackId = tracks.TrackId
26 GROUP BY invoice_items.TrackId, tracks.Name
27 ORDER BY 5 DESC, 1;
28
29 /* Revenue per Album */
30 SELECT tracks.AlbumId AS 'Album ID', albums.Title AS "Album Title", tracks.UnitPrice AS 'Track Unit
31 Price', count(invoice_items.UnitPrice) AS 'Number of Tracks sold per Album'
32     , sum(invoice_items.UnitPrice) AS 'Revenue per Album'
33 FROM invoice_items
34 INNER JOIN tracks
35     ON invoice_items.TrackId = tracks.TrackId
36 INNER JOIN albums
37     ON tracks.AlbumId = albums.AlbumId
38 GROUP BY albums.Title
39 ORDER BY 4 DESC, 1;
40
41 /* Revenue per Album - SPOT VALIDATION of total sales of tracks sold for the album */
42 SELECT tracks.AlbumId AS 'Album ID', albums.Title AS "Album Title", tracks.Name, tracks.UnitPrice AS
43 'Track Unit Price', count(invoice_items.UnitPrice) AS 'Count of Sales per Track'
44     , sum(invoice_items.UnitPrice) AS 'Revenue per Track'
45 FROM invoice_items
46 INNER JOIN tracks
47     ON invoice_items.TrackId = tracks.TrackId
48 INNER JOIN albums
49     ON tracks.AlbumId = albums.AlbumId
50 WHERE albums.Title = 'Minha Historia'
51 GROUP BY tracks.Name
52 ORDER BY 5 DESC;
53
54 /* Revenue per Album - SPOT VALIDATION of total number of tracks sold for the album*/
55 SELECT tracks.AlbumId AS 'Album ID', albums.Title AS "Album Title", tracks.Name AS "Track Name",
56 count(invoice_items.TrackId) AS 'Number of Tracks Sold'
57 FROM tracks
58 INNER JOIN albums
59     ON albums.AlbumId = tracks.AlbumId
60 LEFT JOIN invoice_items
61     ON albums.AlbumId = tracks.AlbumId AND invoice_items.TrackId = tracks.TrackId
62 WHERE albums.Title = 'Minha Historia'
63 GROUP BY tracks.TrackId
64 ORDER BY 4 DESC;
65
66 /* Revenue per Genre */
67 SELECT genres.GenreId AS 'Genre ID', genres.Name AS "Genre Name", tracks.UnitPrice AS 'Track Unit
68 Price', count(invoice_items.UnitPrice) AS 'Number of Tracks sold per Genre'
69     , round(sum(invoice_items.UnitPrice), 2) AS 'Revenue per Genre'
70 FROM invoice_items
71 INNER JOIN tracks
72     ON invoice_items.TrackId = tracks.TrackId
73 INNER JOIN genres
74     ON tracks.GenreId = genres.GenreId

```

```

68 GROUP BY genres.Name, tracks.UnitPrice
69 ORDER BY 4 DESC, 1;
70
71 /* Revenue per Genre - SPOT VALIDAATION of total sales of tracks sold for the genre */
72 SELECT genres.GenreId AS 'Genre ID', genres.Name AS "Genre", tracks.Name AS 'Track Name',
73        tracks.UnitPrice AS 'Track Unit Price'
74        ,count(invoice_items.UnitPrice) AS 'Count of Sales per Genre', sum(invoice_items.UnitPrice) AS
75        'Revenue per Track'
76 FROM invoice_items
77 INNER JOIN tracks
78        ON invoice_items.TrackId = tracks.TrackId
79 INNER JOIN genres
80        ON tracks.GenreId = genres.GenreId
81 WHERE genres.Name = 'Rock'
82 GROUP BY tracks.Name
83 ORDER BY 5 DESC;
84
85 /* Revenue per Genre - SPOT VALIDATION of total number of tracks sold for the genre*/
86 SELECT genres.GenreId AS 'Genre ID', genres.Name AS "Genre", tracks.Name AS "Track Name",
87        count(invoice_items.TrackId) AS 'Number of Tracks Sold'
88 FROM tracks
89 INNER JOIN genres
90        ON tracks.GenreId = genres.GenreId
91 LEFT JOIN invoice_items
92        ON tracks.GenreId = genres.GenreId AND invoice_items.TrackId = tracks.TrackId
93 WHERE genres.Name = 'Rock'
94 GROUP BY tracks.Name
95 ORDER BY 4 DESC;
96
97 -- Which countries have the highest sales revenue? What percent of total revenue does each country
98 make up?
99 ---- Which countries have the highest sales revenue?
100 SELECT BillingCountry, sum(Total) AS 'Total Revenue per Country'
101 FROM invoices
102 GROUP BY BillingCountry
103 ORDER BY 2 DESC;
104
105 ---- What percent of total revenue does each country make up?
106 WITH TOTAL_REVENUE AS (
107     SELECT sum(Total) AS all_total_revenue
108     FROM invoices
109 ),
110 COUNTRY_REVENUE AS (
111     SELECT BillingCountry, sum(Total) AS country_revenue
112     FROM invoices
113     GROUP BY BillingCountry
114 )
115 SELECT invoices.BillingCountry, COUNTRY_REVENUE.country_revenue AS 'Total Revenue per Country',
116        TOTAL_REVENUE.all_total_revenue AS 'Overall Total Revenue',
117        round((COUNTRY_REVENUE.country_revenue / TOTAL_REVENUE.all_total_revenue) * 100, 2) AS
118        'Percent of Total Revenue'
119 FROM invoices
120 INNER JOIN COUNTRY_REVENUE ON invoices.BillingCountry = COUNTRY_REVENUE.BillingCountry
121 CROSS JOIN TOTAL_REVENUE
122 GROUP BY invoices.BillingCountry
123 ORDER BY 2 DESC;
124
125 -- How many customers did each employee support, what is the average revenue for each sale, and what
126 is their total sale?
127 SELECT employees.EmployeeId, employees.FirstName || " " || employees.LastName AS 'Employee Name',
128        count(DISTINCT customers.CustomerId) AS 'Count of Customers per Employee',
129        round(avg(invoices.Total), 2) AS 'Average Revenue per Sale', round(sum(invoices.Total), 2) AS
130        'Total Sale per Employee'
131 FROM employees
132 LEFT JOIN customers ON employees.EmployeeId = customers.SupportRepId
133 LEFT JOIN invoices ON customers.CustomerId = invoices.CustomerId
134 GROUP BY 1, 2
135 ORDER BY EmployeeId;

```