Chinook数据分析

1. 哪位员工的顾客总数最多?

- 将员工的姓氏和名字连接起来。
- COUNT 客户 ID 以获取客户总数。

代码块

```
SELECT
      e.LastName || ' ' || e.FirstName AS Employee,
      COUNT(c.customerid) AS Total_Customer
FROM Employee AS e
INNER JOIN Customer AS c
ON e.EmployeeId = c.SupportRepId
GROUP BY 1
ORDER BY 1 DESC;
```

	Employee	Total_Customer
1	Peacock Jane	21
2	Park Margaret	20
3	Johnson Steve	18

2. 根据发票，我们的主要客户是谁?

- 选择客户的名字和姓氏并计算其发票总额。

代码块

```
SELECT
               C.FirstName || ' ' || C.LastName AS Customer_Name,
        SUM(I.Total) AS Total_spent
FROM Invoice AS I
INNER JOIN Customer  AS C
ON C.CustomerId = I.CustomerId
GROUP BY 1
ORDER BY 2 DESC
LIMIT 5;
```

	Customer_Name	Total_spent
1	Helena Holý	49.62
2	Richard Cunningham	47.62
3	Luis Rojas	46.62
4	Ladislav Kovács	45.62
5	Hugh O'Reilly	45.62

3. 摇滚乐听众是谁？我们想知道所有摇滚乐听众的电子邮件地址、名字、姓氏和音乐类型。

- 选择客户的电子邮件、名字和姓氏以及流派，并将流派过滤为摇滚音乐。

代码块

```
SELECT
      C.Email, C.FirstName, C.LastName, G.Name AS Genre
FROM Customer AS C
INNER JOIN Invoice AS I
ON I.CustomerId = C.CustomerId
INNER JOIN InvoiceLine AS Il
ON Il.InvoiceId = I.InvoiceId
INNER JOIN Track AS T
ON T.TrackId = Il.TrackId
INNER JOIN Genre AS G
ON G.GenreId = T.GenreId
WHERE Genre = 'Rock'
GROUP BY 1,2,3,4
ORDER BY 1;
```

	Email	FirstName	LastName	Genre
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

4. 谁在创作摇滚乐?

- 选择艺术家的姓名并计算他们创作的摇滚音乐的数量。

代码块

```
SELECT
        Ar.Name As Artist,
        COUNT(G.name) AS Total_rock
FROM Artist AS Ar
INNER JOIN Album AS Al
ON Al.ArtistId = Ar.ArtistId
INNER JOIN Track AS T
ON T.AlbumId = Al.AlbumId
INNER JOIN Genre AS G
ON G.GenreId = T.GenreId
WHERE G.Name = 'Rock'
GROUP BY 1
ORDER BY 2 DESC
LIMIT 10;
```

	Artist	Total_rock
1	Led Zeppelin	114
2	U2	112
3	Deep Purple	92
4	Iron Maiden	81
5	Pearl Jam	54
6	Van Halen	52
7	Queen	45
8	The Rolling Stones	41
9	Creedence Clearwater Revival	40
10	Kiss	35

5. 根据发票明细，哪位艺术家的收入最高？使用这位艺术家来查找哪位客户在这位艺术家身上花费最多。

收入最高的艺术家

- 选择艺术家的姓名和
- 将单价乘以数量来计算总收入。

代码块

```
SELECT
        Ar.Name AS Artist,
        ROUND(SUM(T.UnitPrice *Il.Quantity),2) AS Total_earned
FROM Artist AS Ar
INNER JOIN Album AS Al
ON Al.ArtistId = Ar.ArtistId
INNER JOIN Track AS T
ON T.AlbumId = Al.AlbumId
INNER JOIN InvoiceLine AS Il
ON Il.TrackId = T.TrackId
INNER JOIN Invoice AS I
ON I.InvoiceId = Il.InvoiceId
GROUP BY 1
ORDER BY 2 DESC
LIMIT 5;
```

	Artist	Total_earned
1	Iron Maiden	138.6
2	U2	105.93
3	Metallica	90.09
4	Led Zeppelin	86.13
5	Lost	81.59

在 Iron Maiden 上花费最多的顾客

- 使用 CTE 获取艺术家姓名、客户 ID、客户姓名以及将单价乘以数量得出的花费金额。
- 从 CTE 中选择艺术家、客户 ID、客户姓名和消费金额。
- 将 A 部分中编写的查询作为子查询插入到 WHERE 子句中，以将结果筛选为仅收入最高的艺术家。

代码块

```
WITH Customer_spending AS(
                        SELECT  Ar.Name As Artist,
                                C.CustomerId AS Customer_Id,
                                C.FirstName AS First_Name,
                                C.LastName AS Last_Name,
                                T.UnitPrice* Il.Quantity AS Amount_spent
                        FROM Artist AS Ar
                        INNER JOIN Album AS Al
                        ON Al.ArtistId = Ar.ArtistId
                        INNER JOIN Track AS T
                        ON T.AlbumId = Al.AlbumId
                        INNER JOIN InvoiceLine AS Il
                        ON Il.TrackId = T.TrackId
                        INNER JOIN Invoice AS I
                        ON I.InvoiceId = Il.InvoiceId
                        INNER JOIN Customer AS C
                        ON C.CustomerId = I.CustomerId
                        ORDER BY 5 DESC)
SELECT          Artist,
                Customer_Id,
                First_Name,
                Last_Name,
                SUM(Amount_spent) AS Amount_spent
FROM Customer_spending
WHERE Artist =  (SELECT Artist
                FROM(SELECT         Ar.Name As Artist,
                                ROUND(SUM(T.UnitPrice *Il.Quantity),2) AS
Total_earned
```

```
                        FROM Artist AS Ar
                        INNER JOIN Album AS Al
                        ON Al.ArtistId = Ar.ArtistId
                        INNER JOIN Track AS T
                        ON T.AlbumId = Al.AlbumId
                        INNER JOIN InvoiceLine AS Il
                        ON Il.TrackId = T.TrackId
                        INNER JOIN Invoice AS I
                        ON I.InvoiceId = Il.InvoiceId
                        GROUP BY 1
                        ORDER BY 2 DESC
                        LIMIT 1) t1)
GROUP BY 1,2,3,4
ORDER BY 5 DESC
LIMIT 6;
```

	Artist	Customer_Id	First_Name	Last_Name	Amount_spent
1	Iron Maiden	55	Mark	Taylor	17.82
2	Iron Maiden	35	Madalena	Sampaio	15.84
3	Iron Maiden	16	Frank	Harris	13.86
4	Iron Maiden	36	Hannah	Schneider	13.86
5	Iron Maiden	5	František	Wichterlová	8.91
6	Iron Maiden	27	Patrick	Gray	8.91

6. 列出歌曲长度大于平均歌曲长度的曲目。

- 选择名称和毫秒。
- 使用子查询查找平均歌曲长度。
- 在 WHERE 子句中插入子查询来过滤结果。

代码块

```
SELECT
        Name,
        Milliseconds AS Song_length_ms
FROM Track
WHERE Milliseconds > (SELECT ROUND(AVG(Milliseconds),2) AS Avg_Song_length FROM
 Track)
ORDER BY 2 DESC
```

```
7   LIMIT 10;
```

	Name	Song_length_ms
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

7. 找出每个国家最受欢迎的流派

- 使用子查询获取国家、类型 ID、类型名称和购买次数。
- 编写另一个子查询来从第一个子查询中选择最大购买额。
- 使用第一个子查询作为查询 3 来连接第二个子查询。
- 从查询 2 和查询 3 中选择国家、类型 ID、类型名称和最大购买量。

代码块

```
1   SELECT
2           sub2.Country,
3           sub2.Purchases,
4           sub3.Genre_Id,
5           sub3.Genre_Name
6   FROM        (SELECT Country, MAX(purchases) AS Purchases
7            FROM        (SELECT C.Country AS Country,
8                               G.GenreId AS Genre_Id,
9                               G.Name AS Genre_Name,
10                              COUNT(*) AS purchases
11                      FROM Customer AS C
12                      INNER JOIN Invoice AS I
13                      ON C.CustomerId = I.CustomerId
14                      INNER JOIN InvoiceLine AS Il
15                      ON I.InvoiceId = Il.InvoiceId
16                      INNER JOIN Track AS T
17                      ON Il.TrackId = T.TrackId
18                      INNER JOIN Genre AS G
19                      ON G.GenreId = T.GenreId
20                      GROUP BY 1,2,3
```

```
21                  ORDER BY 1) sub1 -- first sub query
22          GROUP BY 1
23          ORDER BY 2) sub2       --- second sub query
24  JOIN        (SELECT C.Country AS Country,
25                  G.GenreId AS Genre_Id,
26                  G.Name AS Genre_Name,
27                  COUNT(*) AS purchases
28                FROM Customer AS C
29                INNER JOIN Invoice AS I
30                ON C.CustomerId = I.CustomerId
31                INNER JOIN InvoiceLine AS Il
32                ON I.InvoiceId = Il.InvoiceId
33                INNER JOIN Track AS T
34                ON Il.TrackId = T.TrackId
35                INNER JOIN Genre AS G
36                ON G.GenreId = T.GenreId
37                GROUP BY 1,2,3
38                ORDER BY 1) sub3 --- third query
39  WHERE sub2.Country = sub3.Country AND sub2.Purchases = sub3.purchases

40  ORDER BY 1,4;
```

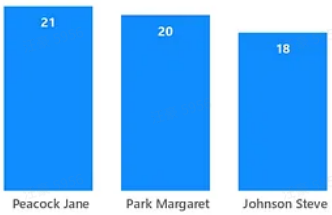
	Country	Purchases	Genre_Id	Genre_Name
1	Argentina	9	4	Alternative & Punk
2	Argentina	9	1	Rock
3	Australia	22	1	Rock
4	Austria	15	1	Rock
5	Belgium	21	1	Rock
6	Brazil	81	1	Rock
7	Canada	107	1	Rock
8	Chile	9	1	Rock
9	Czech Republic	25	1	Rock
10	Denmark	21	1	Rock

附录

CHINOOK DATABASE

Tracks	Genre	Artists	Customers
3503	25	275	59

Number of Customers per Employee



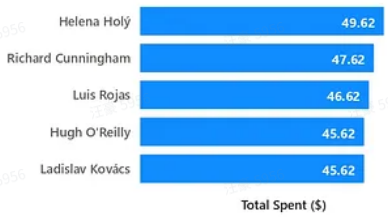
Highest Earning Artist

Iron Maiden

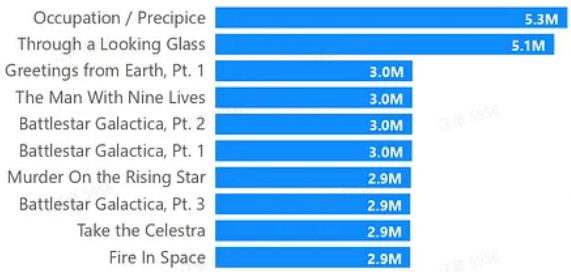
Customer That Spent Most on Iron Maiden

Mark Taylor

Top 5 Customers with amount spent on Invoices



Top 10 songs with Length higher than Average



Artists who have written the most Rock Music