✅ Task 4 – SQL Joins (INNER, LEFT) with Business Questions

Dataset: Chinook Database

1️⃣ List all customers and their invoices (INNER JOIN)

👉 Show only customers who have made purchases.

Copy code

Sql

```
SELECT
    c.CustomerId,
    c.FirstName,
    c.LastName,
    i.InvoiceId,
    i.InvoiceDate,
    i.Total
FROM Customers c
INNER JOIN Invoices i
ON c.CustomerId = i.CustomerId;
```

Business Insight:

Only active (paying) customers are included.

2️⃣ List all customers including those with NO invoices (LEFT JOIN)

👉 Identify customers who never purchased.

Copy code

Sql

```
SELECT
    c.CustomerId,
    c.FirstName,
    c.LastName,
    i.InvoiceId,
    i.Total
FROM Customers c
LEFT JOIN Invoices i
ON c.CustomerId = i.CustomerId;
```

Business Insight:

Helps find inactive customers for marketing campaigns.

3️⃣ Find customers who have never made a purchase

👉 Uses LEFT JOIN + NULL condition.

Copy code

Sql

```
SELECT
    c.CustomerId,
    c.FirstName,
    c.LastName
FROM Customers c
LEFT JOIN Invoices i
ON c.CustomerId = i.CustomerId
WHERE i.InvoiceId IS NULL;
```

Business Insight:

Target these customers with discounts or promotions.

4️⃣ Show invoice details with customer name

👉 Combine customer and invoice data.

Copy code

Sql

```
SELECT
   i.InvoiceId,
   i.InvoiceDate,
   c.FirstName || ' ' || c.LastName AS CustomerName,
   i.Total
FROM Invoices i
INNER JOIN Customers c
ON i.CustomerId = c.CustomerId;
```

Business Insight:

Useful for billing and customer support.

5️⃣ Total sales amount by each customer

👉 Revenue contribution per customer.

Copy code

Sql

```
SELECT
   c.CustomerId,
   c.FirstName,
   c.LastName,
   SUM(i.Total) AS TotalSpent
FROM Customers c
INNER JOIN Invoices i
ON c.CustomerId = i.CustomerId
GROUP BY c.CustomerId;
```

Business Insight:

Identify high-value customers.

6️⃣ Top 5 customers by total spending

👉 Ranking customers.

Copy code

Sql

```
SELECT
   c.FirstName,
   c.LastName,
   SUM(i.Total) AS TotalSpent
FROM Customers c
INNER JOIN Invoices i
ON c.CustomerId = i.CustomerId
GROUP BY c.CustomerId
ORDER BY TotalSpent DESC
LIMIT 5;
```

Business Insight:

Focus loyalty programs on top customers.

7️⃣ List all tracks with their album and artist name

👉 Multiple INNER JOINs.

Copy code

Sql

```
SELECT
   t.Name AS TrackName,
   al.Title AS AlbumTitle,
   ar.Name AS ArtistName
FROM Tracks t
INNER JOIN Albums al ON t.AlbumId = al.AlbumId
INNER JOIN Artists ar ON al.ArtistId = ar.ArtistId;
```

Business Insight:

Content catalog understanding.

8️⃣ Find employees and the customers they support

👉 Employee–Customer relationship.

Copy code

Sql

```
SELECT
    e.FirstName || ' ' || e.LastName AS EmployeeName,
    c.FirstName || ' ' || c.LastName AS CustomerName
FROM Employees e
LEFT JOIN Customers c
ON e.EmployeeId = c.SupportRepId;
```

Business Insight:

Evaluate employee workload.

9️⃣ Total sales by country

👉 Regional performance.

Copy code

Sql

```
SELECT
    BillingCountry,
    SUM(Total) AS CountrySales
FROM Invoices
GROUP BY BillingCountry
ORDER BY CountrySales DESC;
```

Business Insight:

Identify top-performing countries.

🔟 Difference between INNER JOIN and LEFT JOIN (Theory – Exam Ready)

INNER JOIN

LEFT JOIN

Returns only matching rows

Returns all rows from left table

Excludes NULL matches

Includes NULLs

Used for active data

Used to find missing data

✅ Conclusion

This task demonstrates:

Correct use of INNER JOIN

Correct use of LEFT JOIN

Real business insights from SQL queries