Chinook Analysis

Prepared By:

Mariam Mostafa-Basmala Taha-Gehad Saeed

Supervised By:

Eng\Ibrahim Alnawasany

Eng\Mohammed Adel

Tools Used:

- MySQL
- Power BI

Date:

August 2025

1. Introduction

This project utilizes the Chinook database as a case study to perform data analysis and visualization, leveraging **MySQL** for querying and **Power BI** for building interactive dashboards.

The Chinook database simulates a digital media store, containing data about employees, customers, invoices, sales transactions, and music collections such as tracks, albums, artists, and genres.

The primary objective of this project is to extract business insights by writing SQL queries to answer key questions and designing professional dashboards that visualize sales performance, employee distribution, customer behavior, and product trends.

2-Data Description

The Chinook database consists of multiple relational tables connected through primary and foreign keys. Below is an overview of the key tables used in this project:

1- Employee Table

- **Employee ID** – Unique identifier for each employee
- **FirstName, LastName** – Employee full name
- **Title** – Job title
- **Reports To** – Manager ID (hierarchy)
- **Birth Date** – Date of birth
- **Hire Date** – Date of hire
- **Address, City, State, Country** – Location details
- **Postal Code** – Postal/ZIP code
- **Phone, Fax, Email** – Contact information

2- Customer Table

- **CustomerId** – Unique identifier for each customer
- **FirstName, LastName** – Customer full name
- **Company** – Company name (if applicable)
- **Address, City, State, Country** – Location details

- **Postal Code** – Postal/ZIP code
- **Phone, Fax, Email** – Contact information
- **Support Rep ID** – The employee (sales/support rep) assigned to this customer

---

3- Invoice Table

- **Invoice ID** – Unique identifier for each invoice
- **CustomerId** – Links invoice to the customer
- **Invoice Date** – Date of invoice
- **Billing Address, City, State, Country, Postal Code** – Billing details
- **Total** – Total amount of the invoice

4- Invoice Line Table

- **Invoice Line ID** – Unique identifier for each invoice line
- **Invoice ID** – Links to invoice
- **Track ID** – The purchased track
- **Unit Price** – Price per unit
- **Quantity** – Quantity purchased

5- Track Table

- **Track ID** – Unique identifier for each track
- **Name** – Track name
- **Album ID** – Links to album
- **MediaType ID** – Links to media type
- **Genre ID** – Links to genre
- **Composer** – Composer name
- **Milliseconds** – Length of track in milliseconds
- **Bytes** – File size

- **Unit Price** – Price of track

6-Album Table

- **Album ID** – Unique identifier for each album
- **Title** – Album title
- **Artist ID** – Links to artist

7- Artist Table

- **Artist ID** – Unique identifier for each artist
- **Name** – Artist name

8- Genre Table

- **Genre ID** – Unique identifier for each genre
- **Name** – Genre name (e.g., Rock, Jazz, Metal)

9- MediaType Table

- **MediaType ID** – Unique identifier for each media type
- **Name** – Media format (e.g., MPEG audio, AAC, Protected AAC)

10-**Playlist Table**

- **Playlist ID** – Unique identifier for each playlist
- **Name** – Playlist name

11- Playlist Track Table

- **Playlist ID** – Playlist identifier
- **Track ID** – Track identifier

3- Chinook SQL Queries Documentation

Q1 – Top Genres by Sales

- **Business Question**: Which music genres generate the highest sales revenue?
- **SQL Query:**

```
SELECT Genre. Name, SUM (Invoice Line. Unit Price * Invoice Line. Quantity) AS Total Revenue

FROM Invoice Line

JOIN Track ON Invoice Line. Track ID = Track. Track ID

JOIN Genre ON Track. Genre ID = Genre. GenreID

GROUP BY Genre. Name

ORDER BY Total Revenue DESC

LIMIT 5;
```

- **Explanation:** Joins invoice lines, tracks, and genres to calculate revenue per **genre.**
- **Expected Result:** Top 5 genres with the highest total sales.

Q2 – Best Selling Playlists

- **Business Question**: Which playlists generated the most revenue?
- **SQL Query:**

```
SELECT ROUND (SUM (Invoice Line. Unit Price * Invoice Line. Quantity),0) AS total Revenue,

    Playlist. Name

FROM Track

JOIN Playlist Track ON Track. Track ID = Playlist Track. Track ID

JOIN Playlist ON Playlist. Playlist ID = Playlist Track. Playlist ID

JOIN Invoice Line ON Invoice Line. Track ID = Track. Track ID

GROUP BY Playlist. Name
```

```
ORDER BY total Revenue DESC
```

- **Explanation**: Connects tracks, playlists, and invoice lines to calculate sales.
- **Expected Result:** Playlists ranked by total revenue.

Q3 – Total Revenue by MediaType

- **Business Question**: What is the total revenue for each media type?
- **SQL Query:**

```
SELECT MediaType. Name AS MediaType,
    ROUND (SUM (Invoice Line. Unit Price * Invoice Line. Quantity),0) AS total Revenue
FROM Invoice Line
JOIN Track ON Invoice Line. Track ID = Track. Track ID
JOIN MediaType ON Track. MediaType ID = Media Type. Media Type ID
GROUP BY Media Type
ORDER BY total Revenue DESC.
```

- **Explanation**: Aggregates invoice line sales grouped by media type.
- **Expected Result**: Table of media types with revenue.

Q4 – Monthly Revenue

- **Business Question**: How much revenue is generated each month?
- **SQL Query:**

```
SELECT YEAR (Invoice Date) AS Year, MONTH (Invoice Date) AS Month,
    ROUND(SUM(Total),0) AS Monthly Revenue
FROM Invoice
GROUP BY YEAR (Invoice Date), MONTH (Invoice Date)
ORDER BY Year, Month;
```

- **Explanation:** Uses invoice dates to group revenue monthly.
- **Expected Result**: Timeline of monthly sales totals.

Q5 – Customer Distribution by Country

- **Business Question:** How many customers are there in each country?
- SQL Query:

```
SELECT Country, COUNT(CustomerId) AS Customer Count
FROM Customer
GROUP BY Country
ORDER BY Customer Count DESC.
```

- **Explanation:** Counts customers per country.
- **Expected Result:** Countries with the number of customers.

Q6 – Top 5 Customers by Total Sales

- **Business Question:** Who are the top 5 customers by purchases?
- **SQL Query:**

```
SELECT CONCAT (Customer. First Name," ", Customer. Last Name) AS full name,
    ROUND (SUM (Invoice Line. Unit Price * Invoice Line. Quantity),0) AS total sales,
    DENSE_RANK () OVER (ORDER BY SUM (Invoice Line. Unit Price * Invoice Line. Quantity) DESC) AS ranking
FROM Invoice
JOIN Customer ON Customer. Customer ID = Invoice. Customer ID
JOIN Invoice Line ON Invoice. Invoice ID = Invoice Line. Invoice ID
GROUP BY full name
ORDER BY total sales DESC
LIMIT 5;
```

- **Explanation:** Calculates and ranks customer revenue using window functions.
- **Expected Result:** List of top 5 customers with totals.

Q7 – Customers by Albums Purchased

- **Business Question:** Which customers purchased the most albums?
- **SQL Query:**

```
SELECT CONCAT (c. First Name,' ', c. Last Name) AS full name,
       COUNT (DISTINCT a. Album ID) AS Albums Purchased
FROM Customer c
JOIN Invoice I ON c.CustomerId = i.CustomerId
JOIN InvoiceLine il ON i.InvoiceId = il.InvoiceId
JOIN Track t ON il.TrackId = t.TrackId
JOIN Album a ON t.AlbumId = a.AlbumId
GROUP BY fullname
ORDER BY AlbumsPurchased DESC;
```

- **Explanation:** Counts unique albums bought per customer.
- **Expected Result:** Customers sorted by albums purchased.

Q8 – Active Customers Over Time

- **Business Question**: How many customers were active each year?
- **SQL Query:**

```
SELECT YEAR(InvoiceDate) AS Year,
       COUNT(DISTINCT CustomerId) AS ActiveCustomers
FROM Invoice
GROUP BY YEAR(InvoiceDate)
ORDER BY Year;
```

- **Explanation:** Groups invoices by year and counts distinct customers.
- **Expected Result:** Customers active per year.

Q9 – Top 10 Artists by Number of Tracks

- **Business Question**: Which artists have the most tracks?

- **SQL Query:**

```
SELECT Artist.Name, COUNT(Track.TrackId) AS countTracks,
    DENSE_RANK() OVER(ORDER BY COUNT(Track.TrackId) DESC) AS ranking
FROM Track
JOIN Album ON Album.AlbumId = Track.AlbumId
JOIN Artist ON Artist.ArtistId = Album.ArtistId
GROUP BY Artist.Name
ORDER BY countTracks DESC
LIMIT 10;
```

- **Explanation:** Counts tracks per artist and ranks them**.**
- **Expected Result**: Top 10 artists by number of tracks.

Q10 – Top 10 Artists by Sales

- **Business Question:** Which artists generated the most sales?
- **SQL Query:**

```
SELECT Artist.Name,
    ROUND(SUM(InvoiceLine.UnitPrice * InvoiceLine.Quantity),0) AS TotalSales
FROM InvoiceLine
JOIN Track ON InvoiceLine.TrackId = Track.TrackId
JOIN Album ON Track.AlbumId = Album.AlbumId
JOIN Artist ON Album.ArtistId = Artist.ArtistId
GROUP BY Artist.Name
ORDER BY TotalSales DESC
LIMIT 10;
```

- **Explanation:** Joins invoice lines with artists to compute sales totals.
- **Expected Result:** Top 10 artists by revenue.

Q11 – Albums per Year

- **Business Question:** How many albums were purchased per year?
- **SQL Query:**

```
SELECT YEAR(InvoiceDate) AS Year, COUNT(DISTINCT Album.AlbumId) AS AlbumCount
FROM Invoice I
JOIN InvoiceLine il ON i.InvoiceId = il.InvoiceId
JOIN Track t ON il.TrackId = t.TrackId
JOIN Album ON t.AlbumId = Album.AlbumId
GROUP BY YEAR(InvoiceDate)
ORDER BY Year;
```

- **Explanation:** Counts distinct albums purchased by year.
- **Expected Result:** Albums purchased trend per year.

Q12 – Top Artists by Genre

- **Business Question:** Who are the top artists within each genre?
- **SQL Query:**

```
SELECT Genre. Name AS GenreName, Artist. Name AS ArtistName,
    COUNT(Track.TrackId) AS TrackCount
FROM Track
JOIN Album ON Track.AlbumId = Album.AlbumId
JOIN Artist ON Album.ArtistId = Artist.ArtistId
JOIN Genre ON Track.GenreId = Genre.GenreId
GROUP BY Genre.Name, Artist.Name
ORDER BY GenreName, TrackCount DESC;
```

- **Explanation:** Groups tracks by genre and artist to find leaders.
- **Expected Result:** List of artists ranked within each genre.

Q13 – Top 3 Genres by Number of Tracks (Subquery)

- **Business Question:** What are the top 3 genres by number of tracks?
- **SQL Query:**

```
SELECT GenreName, NumberOfTracks
FROM (
    SELECT Genre. Name AS GenreName,
           COUNT(Track.TrackId) AS NumberOfTracks
    FROM Genre
    JOIN Track ON Genre.GenreId = Track.GenreId
    GROUP BY Genre.GenreId, Genre.Name
    ORDER BY NumberOfTracks DESC
    LIMIT 3
) AS TopGenres;
```

- **Explanation:** Uses a subquery to select only the top 3 genres.
- **Expected Result:** 3 genres with the highest track counts.

Q14 – Tracks per Year (Subquery)

- **Business Question:** How many unique tracks were sold each year?
- **SQL Query:**

```
WITH years AS (
    SELECT DISTINCT YEAR(Invoice.InvoiceDate) AS Year
    FROM Invoice
)
SELECT years.Year, COUNT(DISTINCT InvoiceLine.TrackId) AS noTrack
FROM years
JOIN Invoice ON years.Year = YEAR(Invoice.InvoiceDate)
JOIN InvoiceLine ON Invoice.InvoiceId = InvoiceLine.InvoiceId
```

```
JOIN Track ON InvoiceLine.TrackId = Track.TrackId
GROUP BY years.Year
ORDER BY years.Year;
```

- **Explanation:** Counts unique tracks sold per year using a subquery.
- **Expected Result:** Yearly track purchase count.

Q15 – Top Albums by Number of Tracks

- **Business Question:** Which albums contain the most tracks?
- **SQL Query:**

```
SELECT Album.Title, COUNT(Track.TrackId) AS TrackCount
FROM Album
JOIN Track ON Album.AlbumId = Track.AlbumId
GROUP BY Album.Title
ORDER BY TrackCount DESC
LIMIT 10;
```

- **Explanation:** Counts the number of tracks grouped by album.
- **Expected Result:** Top 10 albums by track count.

Q16 – Top Track, Album, and Media by Sales

- **Business Question:** What are the top-selling tracks, albums, and media types?
- **SQL Query:**

```
--top track
SELECT Track.Name AS TrackName, SUM(InvoiceLine.UnitPrice * InvoiceLine.Quantity) AS TotalSales
FROM InvoiceLine
JOIN Track ON InvoiceLine.TrackId = Track.TrackId
GROUP BY Track.TrackId
ORDER BY TotalSales DESC
```

```
LIMIT 1;


-- Top Album
SELECT Album.Title AS AlbumTitle, SUM(InvoiceLine.UnitPrice * InvoiceLine.Quantity) AS TotalSales
FROM InvoiceLine
JOIN Track ON InvoiceLine.TrackId = Track.TrackId
JOIN Album ON Track.AlbumId = Album.AlbumId
GROUP BY Album.AlbumId
ORDER BY TotalSales DESC
LIMIT 1;


-- Top MediaType
SELECT MediaType.Name AS MediaType, SUM(InvoiceLine.UnitPrice * InvoiceLine.Quantity) AS TotalSales
FROM InvoiceLine
JOIN Track ON InvoiceLine.TrackId = Track.TrackId
JOIN MediaType ON Track.MediaTypeId = MediaType.MediaTypeId
GROUP BY MediaType.Name
ORDER BY TotalSales DESC
LIMIT 1;
```

- **Explanation:** Runs three queries to get the top-selling track, album, and media type.
- **Expected Result:** One record for each: best track, album, and media type.

Q17 – Employee Invoice Distribution

- **Business Question:** How many invoices were handled by each employee?
- **SQL Query:**

```
SELECT CONCAT(e.FirstName, ' ', e.LastName) AS SupportEmployee,
```

```
    COUNT(i.InvoiceId) AS InvoiceCount
FROM Employee e
JOIN Customer c ON e.EmployeeId = c.SupportRepId
JOIN Invoice i ON c.CustomerId = i.CustomerId
GROUP BY e.EmployeeId
ORDER BY InvoiceCount DESC;
```

- **Explanation:** Joins employees, customers, and invoices to count invoices.
- **Expected Result:** Employees ranked by invoice volume.

Q18 – Employee Count by Job Title

- **Business Question:** How many employees are in each job title?
- **SQL Query:**

```
SELECT Title, COUNT(EmployeeId) AS EmployeeCount
FROM Employee
GROUP BY Title
ORDER BY EmployeeCount DESC;
```

- **Explanation:** Groups employees by job title.
- **Expected Result:** Job titles with employee counts.

Q19 – Average Employee Experience

- **Business Question:** What is the average employee experience in years?
- **SQL Query:**

```
SELECT ROUND(AVG(TIMESTAMPDIFF(YEAR,HireDate,CURDATE())),1) AS
AvgExperience
FROM Employee;
```

- **Explanation:** Computes average years of experience from hire date.
- **Expected Result:** Single value of average employee tenure.

Q20 – Customers per Employee

- **Business Question:** How many customers does each employee support?
- **SQL Query:**

```
SELECT CONCAT(e.FirstName,' ',e.LastName) AS EmployeeName,
    COUNT(c.CustomerId) AS CustomerCount
FROM Employee e
LEFT JOIN Customer c ON e.EmployeeId = c.SupportRepId
GROUP BY e.EmployeeId
ORDER BY CustomerCount DESC;
```

- **Explanation:** Counts customers linked to each employee.
- **Expected Result:** Employees with their supported customer counts.

Q21 – Total Sales by Country (Subquery)

- **Business Question:** Which country generates the most sales?
- **SQL Query:**

```
WITH total AS (
  SELECT InvoiceLineId, SUM(UnitPrice * Quantity) AS totalsales
  FROM InvoiceLine
  GROUP BY InvoiceLineId
)
SELECT Customer. Country,
    ROUND(SUM(total.totalsales),0) AS totalsales
FROM InvoiceLine
JOIN total ON total.InvoiceLineId = InvoiceLine.InvoiceLineId
JOIN Invoice ON Invoice.InvoiceId = InvoiceLine.InvoiceId
JOIN Customer ON Customer.CustomerId = Invoice.CustomerId
GROUP BY Customer.Country
```

```
ORDER BY totalsales DESC;
```

- **Explanation:** Uses a subquery to calculate total sales per country.
- **Expected Result:** Countries ranked by sales totals.

Q22 – Customers by Country

- **Business Question:** How many customers are in each country?
- **SQL Query:**

```
SELECT Country, COUNT(CustomerId) AS CustomerCount
FROM Customer
GROUP BY Country
ORDER BY CustomerCount DESC;
```

- **Explanation:** Simple grouping of customers by country.
- **Expected Result:** Table of customer counts per country.

Q23 – Invoice Distribution by Country

- **Business Question:** How many invoices were generated in each country?
- **SQL Query:**

```
SELECT BillingCountry, COUNT(InvoiceId) AS InvoiceCount
FROM Invoice
GROUP BY BillingCountry
ORDER BY InvoiceCount DESC;
```

- **Explanation:** Groups invoices by billing country.
- **Expected Result:** Invoice counts per country.

Q24 – Employee Distribution by City

- **Business Question:** How are employees distributed across cities?
- **SQL Query:**

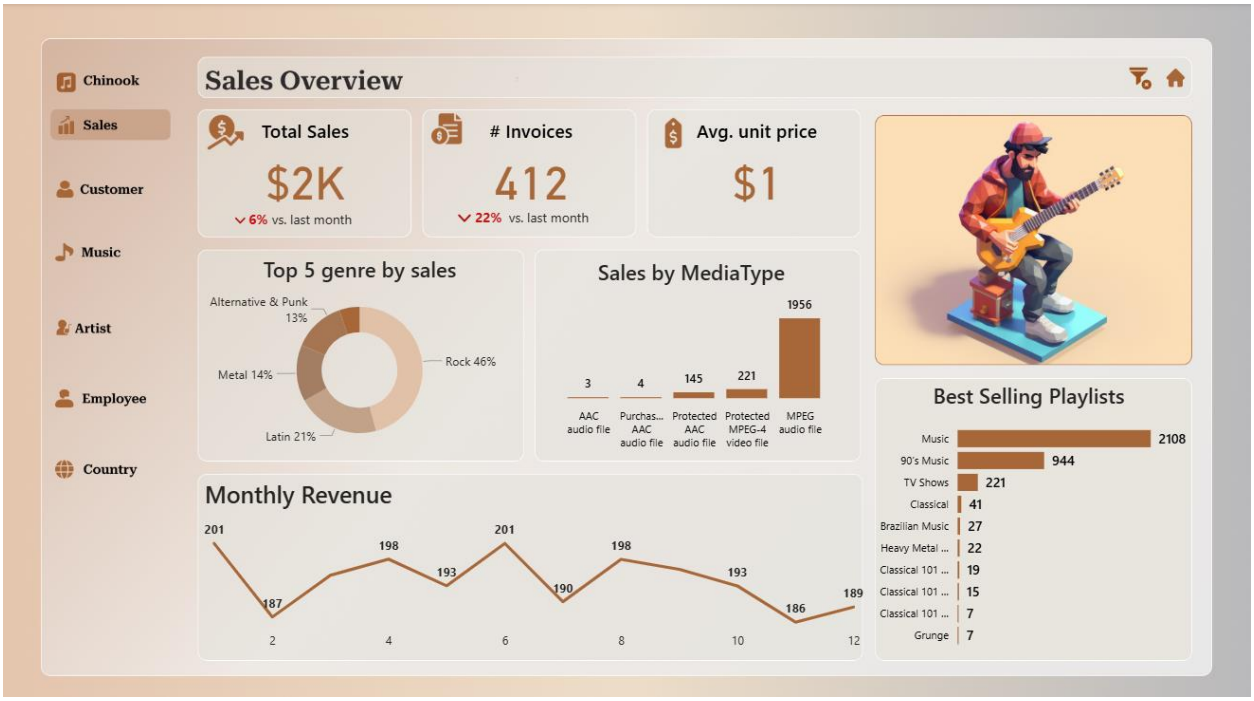
```
SELECT City, COUNT(EmployeeId) AS EmployeeCount
FROM Employee
GROUP BY City
ORDER BY EmployeeCount DESC;
```

- **Explanation:** Groups employees based on city.
- **Expected Result:** Employee counts per city.

4- Dashboards and Visualization

We developed six interactive dashboards covering key business areas. In addition, we implemented two drill-through features to allow deep exploration of the data.

1-Sales Dashboard



Visualization:

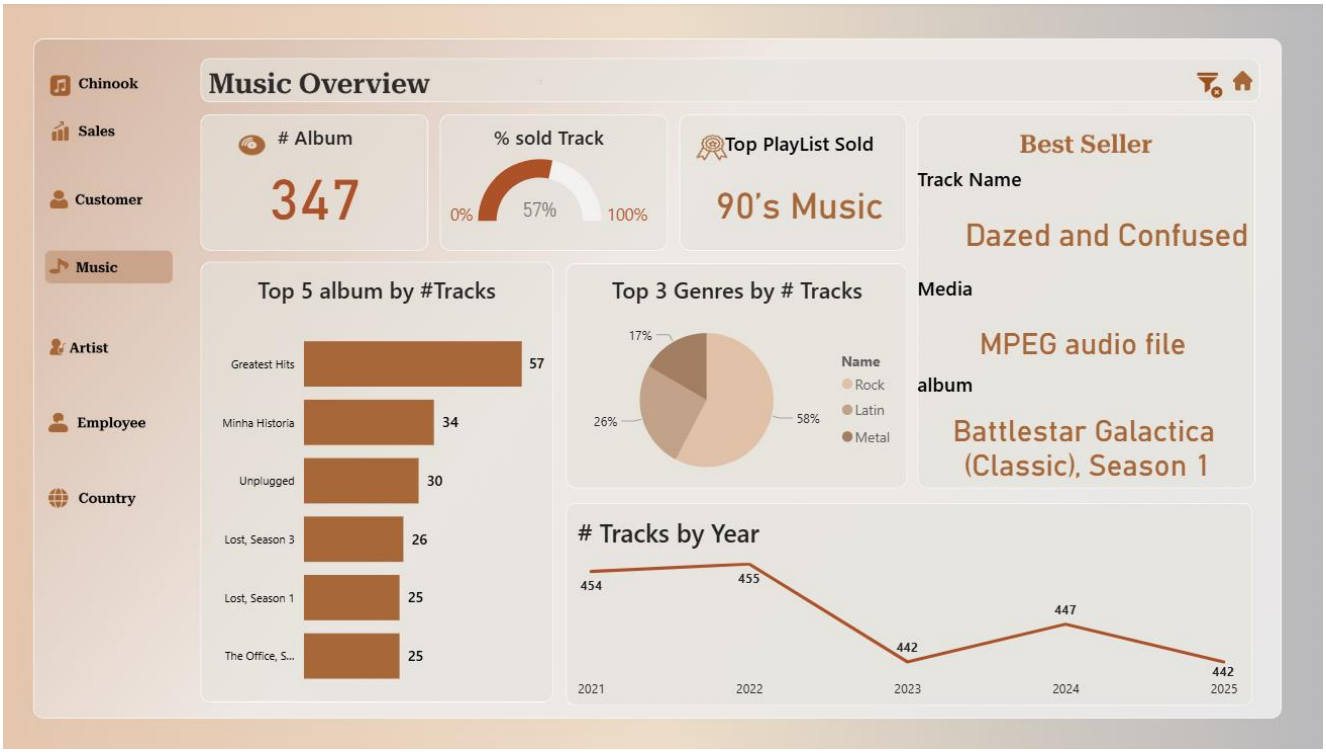
- **KPI: Total Sales, invoices, Avg. unit price**
- **Donut Chart: Top 5 genres by sales**
- **Column chart: Sales by Media type**
- **Line chart: Monthly Revenue**
- **Bar chart: Best Selling Playlist**
- **Bookmark: to open and close the filter**
- **Slicer: filter with Country, Year, Genre**

2-Customer Dashboard

Visualizations:.

- **KPI: # Customer, # Companies, Top Purchasing Customer, AOV Per Customer**
- **Column Chart: Top 5 Customers by Sales**
- **Tree Map: Top 5 Customers by # Albums**
- **Line chart: Active Customer Over Time**
- **Map chart: Customer Distribution**
- **Bookmark: to open and close the filter**
- **Slicer: Filter by Company, Year, State**

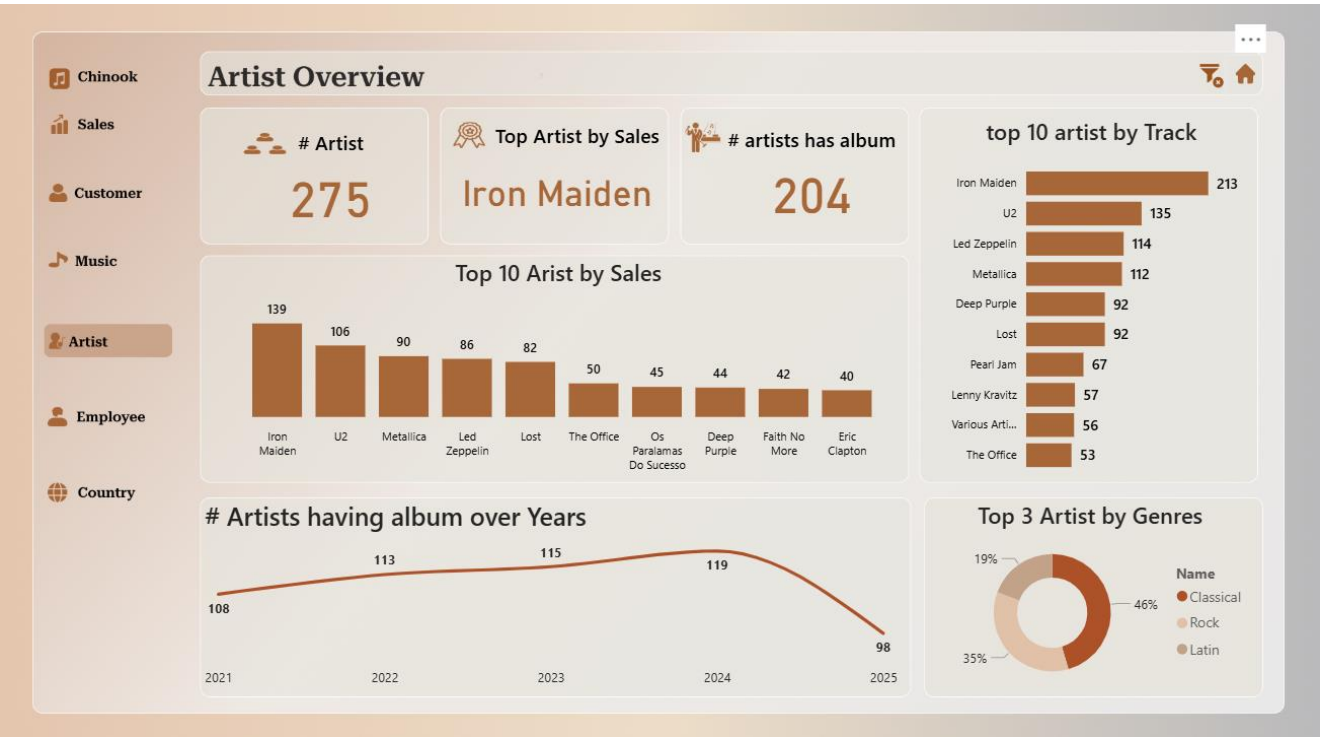
3-Music Dashboard



Visualizations:

- **KPI: #Album, % Sold Track, Top Playlist Sold, Best Seller of (Track Name, Media, Album)**
- **Bar Chart: Top 5 Albums by # Track**
- **Pie Chart: Top 3 Genres by # Tracks**
- **Line chart: #Track by Year**
- **Bookmark: to open and close the filter**
- **Slicer: Filter by Genre, Tracks**

4-Artist Dashboard



Visualizations:

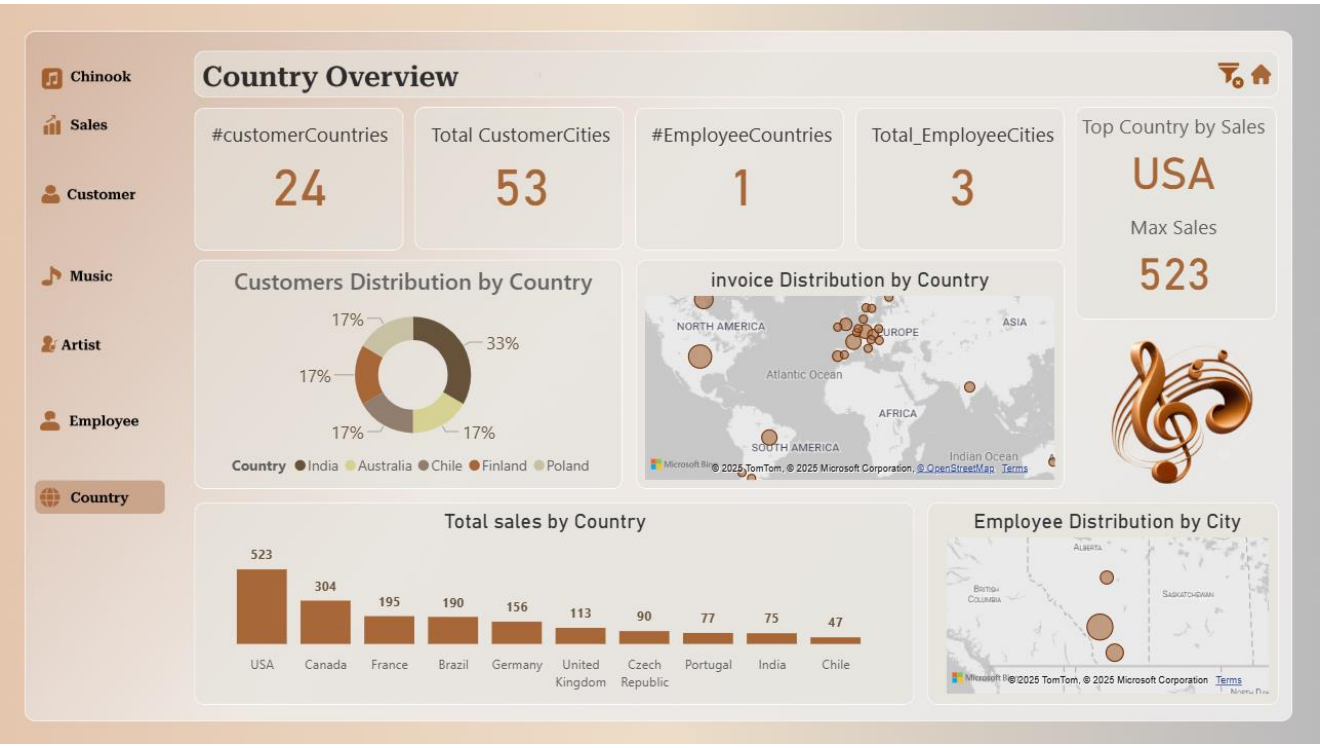
- **KPI:** #Artist, Top Artist by Sales, # Artist has an album
- **Column Chart:** Top 10 Artists by Sales
- **Bar Chart:** Top 10 Artists by Track
- **Line chart:** #Artist Having album over Year
- **Donut Chart:** Top 3 Artists by Genre
- **Bookmark:** to open and close the filter
- **Slicer:** Filter by Playlist, Year

5-Employee Dashboard

Visualizations:

- **KPI: #Employee, No of Job Title, Avg Tenure, support Employee Count, Most Experienced Employee**
- **Donut Chart: Customer To Support Employee Ratio**
- **Column Chart: Employee Invoice Distribution**
- **Tree Map chart: Employee Count by Job Title**
- **Matrix Chart: Workforce Experience Level**
- **Bookmark: to open and close the filter**
- **Slicer: Filter by Year, Tenure Year**

6-Country Dashboard



Visualizations:

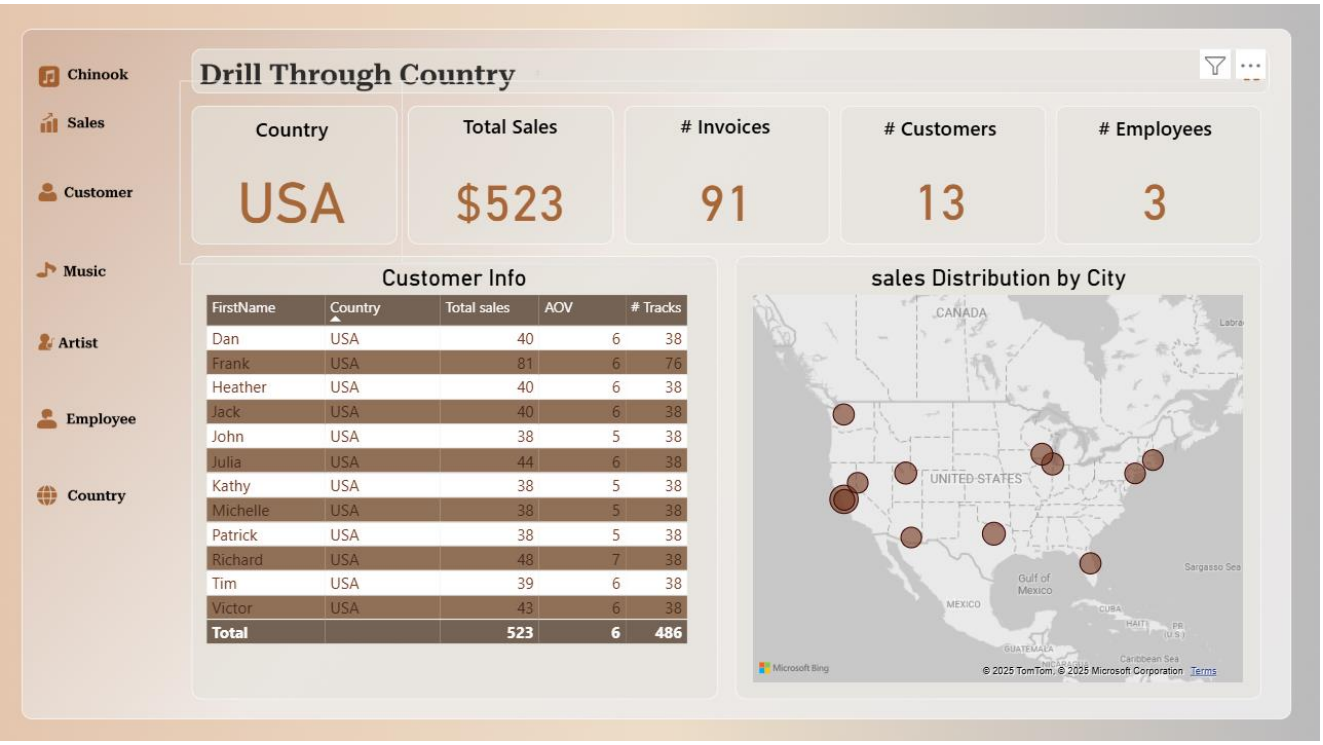
- **KPI:** #Customer Countries, Total Customer Cities, #Employee, Countries, Total Employee Cities, Top Country by Sales, Max Sales
- **Donut Chart:** Customer Distribution by Country
- **Column Chart:** Total Sales by Country
- **Map Chart:** Employee Distribution by City, Invoice Distribution by Country
- **Bookmark:** to open and close the filter
- **Slicer:** Filter by Year, City

7-Drill Through Customer Dashboard

Visualizations:

- **KPI:** **Top Purchasing Customer, Total Sales, #invoices, Country, AOV**
- **Map Chart:** **Customer Distribution**
- **Matrix Chart:** **Customer Info.**

8-Drill Through Country Dashboard



Visualizations:

- **Kpis:** Country, Total Sales, #Invoices, #Customers, #Employees
- **Map Chart:** Sales Distribution by City
- **Matrix Chart:** Customer Info.

5- Insights& Results

📊 Key Insights

- 💰 **Total Sales: 2,000**
- 🧾 **Number of Invoices: 412**
- 👥 **Number of Customers: 59**
- 🏆 **Top Purchasing Customer: Frank**
- 🎶 **Total Albums: 347**
- 🎧 **Top Playlist: 90’s Music**
- 🎵 **Best Track: Dazed and**
- 🎤 **Top Artist by Sales**: Iron Maiden
- 👦🎤 **Total Artists**: 275
- 👩💼 **Employees**: 8
- 🤝 **Support Employees**: 3
- 📈 **Average Experience (Years)**: 22
- 🌍 **Customer Countries**: 24
- 🏢 **Employee Countries**: 1 (3 Cities)
- 🥇 **Top Country by Sales**: USA

6- **Recommendation**

---

1. **Focus on the USA Market:** Since the USA is the top country by sales, strengthen marketing
   Campaigns and partnerships there.
2. **Expand Customer Base:** Current customers are only 59; launch loyalty programs and
   Promotions to attract new customers.
3. **Leverage Top Customer (Frank):** Use personalized offers and VIP programs to retain and
   Increase spending from top customers.
4. **Promote Popular Content:** Highlight best-selling albums (Battle Star), playlists (90's Music),
   And artists (Iron Maiden) in marketing campaigns.
5. **Increase Employee Diversity**: Employees come from only one country; consider hiring from
   Other regions can bring wider perspectives.
6. **Enhance Support Team**: Only 3 support employees — expanding this team can improve
   Customer satisfaction.
7. **Utilize Experienced Staff:** With an average of 22 years of experience, invest in knowledge-
   Sharing programs to train new hires.
8. **Target New Countries:** Since customers already span 24 countries, explore localized
   Marketing to grow in underperforming regions.