Chinook | Data dictionary

The Chinook database is a **sample database** used for teaching and practising database management concepts and SQL queries. It models a **digital media store**, containing tables related to **music albums**, **artists**, **customers**, **invoices**, and more.

Data model

The Chinook database follows a relational data model, with structured tables representing various entities related to a digital music store, as seen in Figure1. Here are some key characteristics and observations about the data model of the Chinook database:

- **Tables**: The database is organised into multiple tables, each representing a specific entity or concept, such as albums, artists, customers, invoices, and tracks. This conforms to the principles of a relational data model, where data is stored in structured tables.
- **Primary keys**: Each table typically has a primary key column (e.g., AlbumId, ArtistId, CustomerId) that uniquely identifies each record within that table. Primary keys are used to establish relationships between tables.
- **Foreign keys**: Relationships between tables are established using foreign key columns (e.g., ArtistId in the Albums table), which reference the primary key columns in related tables. These foreign keys create associations between records in different tables, allowing for data retrieval and analysis across tables.

The data model, including the column names, data types, feature descriptions, and their constraints, is included in Table1.

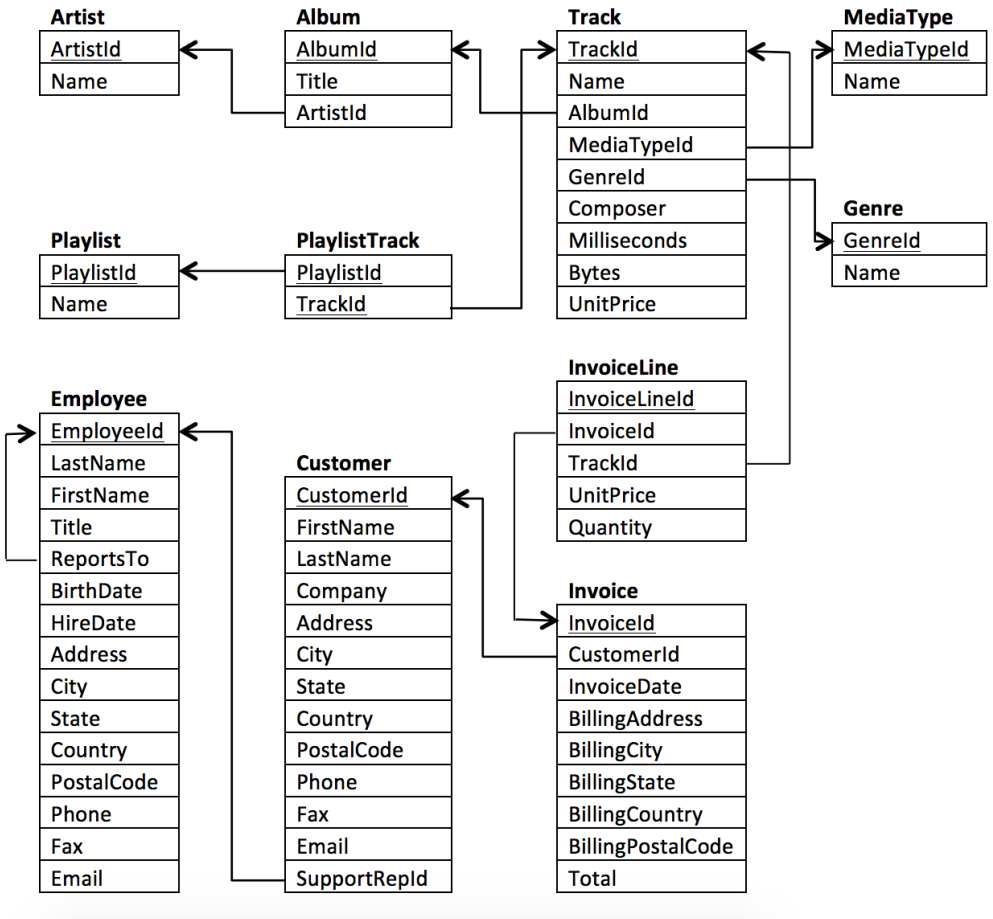

Figure1 The ERD for the Chinook database.

Table1: The data model for the Chinook database.

Albums table			
This table stores information about music albums, including their titles and associated artists.			
Column name	**Data type**	**Feature description**	**Constraints**
AlbumId	INT	Unique identifier for each album.	Primary key
Title	VARCHAR(160)	The title of the album.	NOT NULL
ArtistId	INT	Identifier linking the album to a specific artist.	Foreign key (Artists)

Artists table			
This table contains details about music artists or performers.			
Column name	**Data type**	**Feature description**	**Constraints**
ArtistId	INT	Unique identifier for each artist.	Primary key
Name	VARCHAR(120)	The name of the artist.	NOT NULL

Customers table			
This table stores information about customers who purchase music from the store.			
Column name	**Data type**	**Feature description**	**Constraints**
CustomerId	INT	Unique identifier for each customer.	Primary key
FirstName	VARCHAR(40)	The customer's first name.	NOT NULL
LastName	VARCHAR(20)	The customer's last name.	NOT NULL
Company	VARCHAR(80)	The name of the customer's company (if applicable).	
Address	VARCHAR(70)	The customer's street address.	
City	VARCHAR(40)	The city where the customer is located.	
State	VARCHAR(40)	The state or region where the customer is located.	

Country	VARCHAR(40)	The country where the customer is located.	
PostalCode	VARCHAR(10)	The postal or ZIP code associated with the customer's address.	
Phone	VARCHAR(24)	The customer's contact phone number.	
Fax	VARCHAR(24)	The facsimile (fax) number for the customer.	
Email	VARCHAR(60)	The customer's email address.	
SupportRepId	INT	Identifier linking the customer to a support representative.	Foreign key (Employees)

Invoices table

This table contains records of customer invoices, including details about purchases made.

Column name	Data type	Feature description	Constraints
InvoiceId	INT	Unique identifier for each invoice.	Primary key
CustomerId	INT	Identifier linking the invoice to a specific customer.	Foreign key (Customers)
InvoiceDate	DATETIME	The date when the invoice was issued.	NOT NULL
BillingAddress	VARCHAR(70)	The billing address for the invoice.	
BillingCity	VARCHAR(40)	The billing city for the invoice.	
BillingState	VARCHAR(40)	The billing state or region for the invoice.	
BillingCountry	VARCHAR(40)	The billing country for the invoice.	
BillingPostalCode	VARCHAR(10)	The billing postal or ZIP code for the invoice.	
Total	NUMERIC(10, 2)	The total amount of the invoice.	

InvoiceItemstable

This table stores details about individual items (tracks) within each invoice.

Column name	Data type	Feature description	Constraints
InvoiceLineId	INT	Unique identifier for each invoice item.	Primary key
InvoiceId	INT	Identifier linking the invoice item to a specific invoice.	Foreign key (Invoices)
TrackId	INT	Identifier linking the invoice item to a specific music track.	Foreign key (Tracks)
UnitPrice	NUMERIC(10, 2)	The unit price of the track at the time of the purchase.	NOT NULL
Quantity	INT	The quantity of the track purchased.	NOT NULL

MediaTypes table

This table contains information about different media types used for music tracks.

Column name	Data type	Feature description	Constraints
MediaTypeId	INT	Unique identifier for each media type.	Primary key
Name	VARCHAR(120)	The name of the media type.	NOT NULL

Tracks table

This table stores information about individual music tracks, including their titles, associated albums, media types, genres, composers, durations, sizes, and prices.

Column name	Data type	Feature description	Constraints
TrackId	INT	Unique identifier for each track.	Primary key
Name	VARCHAR(200)	The name of the track.	NOT NULL
AlbumId	INT	Identifier linking the track to a specific album.	Foreign key (Albums)
MediaTypeId	INT	Identifier linking the track to a specific media type.	Foreign key (MediaTypes)
GenreId	INT	Identifier linking the track to a specific genre.	Foreign key (Genres)
Composer	VARCHAR(220)	The composer or songwriter of the track.	
Milliseconds	INT	The duration of the track in	

		milliseconds.	
Bytes	INT	The size of the track in bytes.	
UnitPrice	NUMERIC(10, 2)	The price of the track.	NOT NULL

Playlists table

This table contains records of playlists created by users, storing the names of the playlists for organising and grouping music tracks.

Column name	Data type	Feature description	Constraints
PlaylistId	INT	Unique identifier for each playlist.	Primary key
Name	VARCHAR(120)	The name of the playlist.	NOT NULL

Playlist_track table

This table serves as a junction between playlists and tracks, storing associations between playlists and the specific tracks they contain.

Column name	Data type	Feature description	Constraints
PlaylistId	INT	Identifier linking the playlist to a specific playlist.	Foreign key (Playlists)
TrackId	INT	Identifier linking the playlist to a specific track.	Foreign key (Tracks)

Genres table

This table holds information about music genres, providing a list of distinct musical categories for categorising and organising tracks based on their genre.

Column name	Data type	Feature description	Constraints
GenreId	INT	Unique identifier for each genre.	Primary key
Name	VARCHAR(120)	The name of the genre.	NOT NULL