Database Table Analysis Report
RadioStation and RadioClips

1. Overview

This report explains the structure and relationships between the **RadioStation** and **RadioClips** tables. It describes what each table stores, the indexes they use, what issues were found, and how to improve them.

2. RadioStation Table

Purpose

The **RadioStation** table keeps information about each radio station, such as its name, location, and online stream link. It acts as a list of all radio stations.

Table Structure

Column Name	Data Type	Description
ID	INT (Primary Key, Auto Increment)	A unique number automatically assigned to each radio station.
StationName	VARCHAR(30)	The short name of the station (e.g., “KCFR”).
Country	TEXT	The country where the station operates.
StateProvince	TEXT	The state or province where the station is located.
City	TEXT	The city of the radio station.
Link	TEXT	The website or streaming link of the radio station.

Indexes

Index Name	Type	Purpose
PRIMARY	PRIMARY	The main index automatically created on the ID column. This makes each station's ID unique and easy to find.
StationName_UNIQUE	UNIQUE	Ensures that each station name is unique so the same station cannot be added twice by mistake.

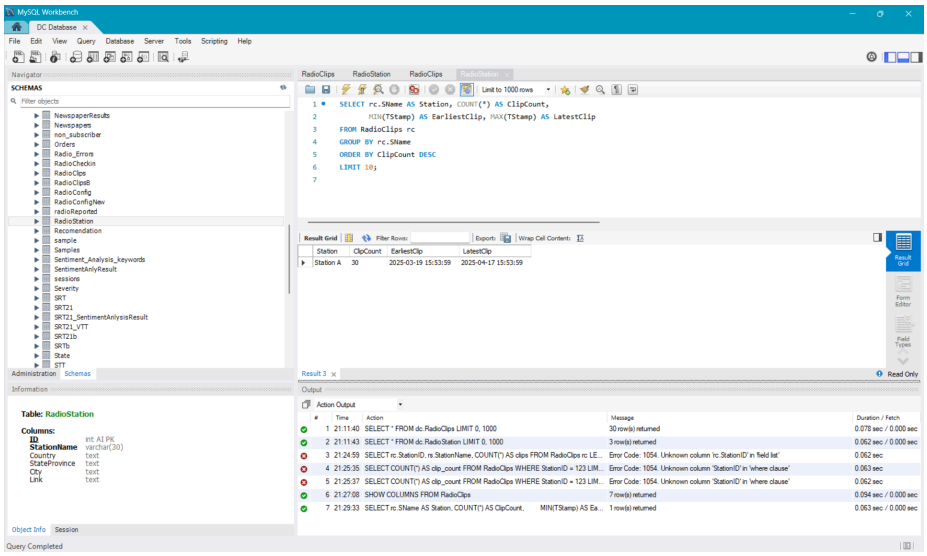
Observations / Issues

1. **Empty data is allowed (no NOT NULL rules).**
   The table allows blank fields. For example, a station could be added without a name or country. This could cause missing or incomplete records.

2. **TEXT type is too large for short fields.**
   The fields City, StateProvince, and Country only need to hold short words like "Denver" or "United States." TEXT is meant for long paragraphs, so it takes up more space and slows down searches. A smaller type such as VARCHAR(100) is more efficient.

3. **No date tracking columns.**
   The table does not record when each station was added or last updated.

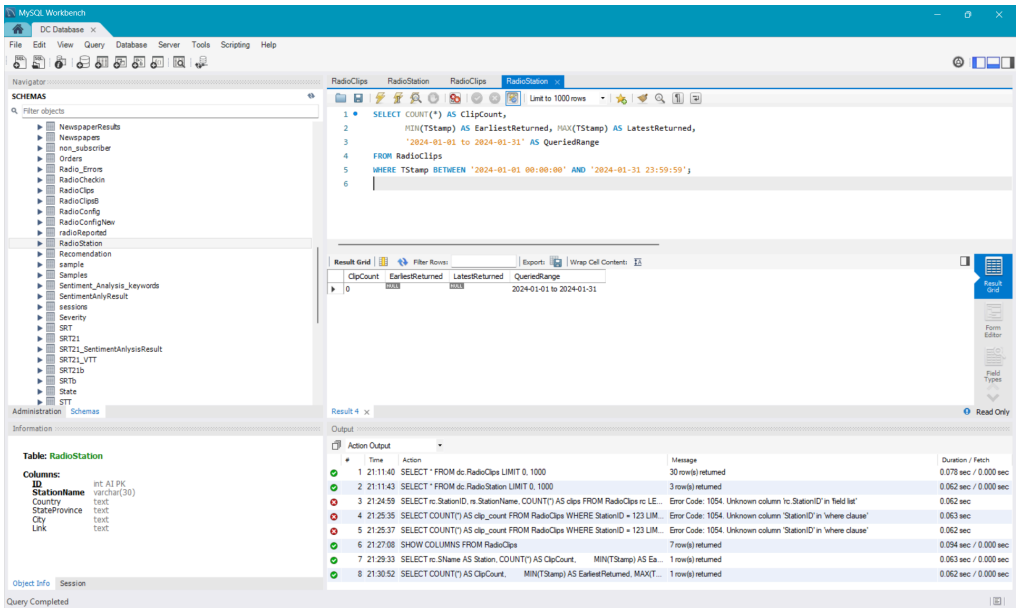
Station Lookup Test:



Observation: The query worked and shows there is only one station ("Station A") with 30 clips between March and April 2025.

Issue: There's only one station in the data, so we can't check performance for multiple stations or confirm the 500+ clips requirement.

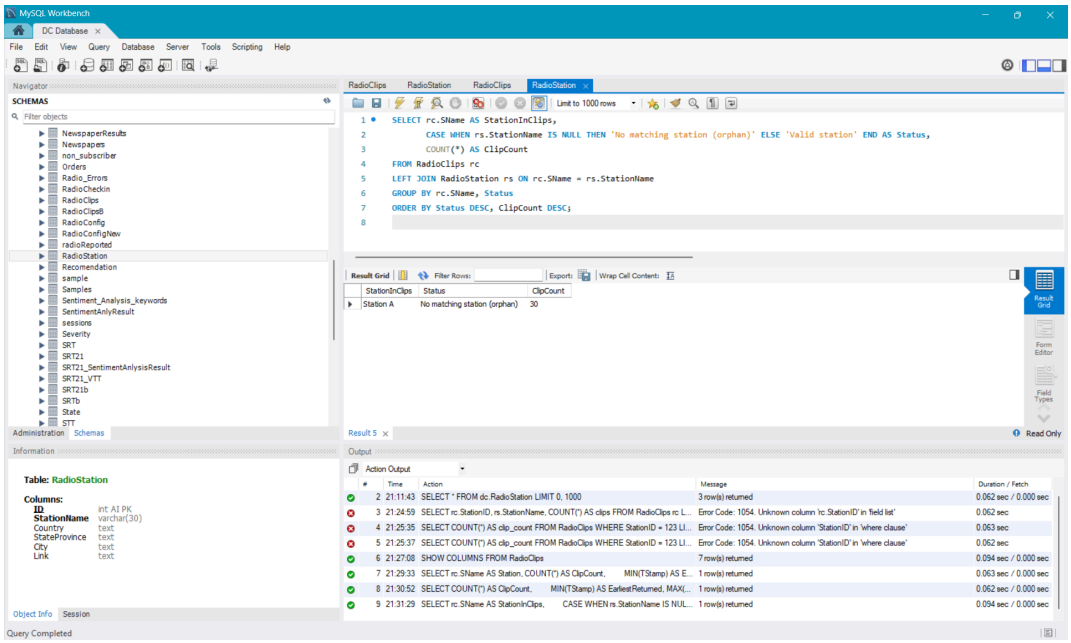
Time Range Query:



Observation: The query ran correctly but found 0 clips in the January 2024 date range.

Issue: There is no data for that time period, so nothing was returned. The timestamp column likely works but just doesn't have entries in that range.

Invalid Foreign Key Check:



Observation: All 30 clips belong to “Station A,” which has no matching record in the RadioStation table.

Issue: This shows there’s no foreign key link between the tables, allowing orphaned records (clips that don’t match any real station).

Recommendations

- Add **NOT NULL** to important columns like StationName and Country to make sure those values are always filled in.
- Change **TEXT** fields (City, StateProvince, Country) to **VARCHAR(100)** for faster searches and less space usage.
- Add **created_at** and **updated_at** columns to track when records are added or changed.
- Keep the **UNIQUE** index on StationName because it helps prevent duplicates.

3. RadioClips Table

Purpose

The **RadioClips** table stores audio clips or recordings from different stations. Each row represents one recorded clip.

Table Structure

Column Name	Data Type	Description
ID	INT (Primary Key, Auto Increment)	A unique number for each clip.
FName	VARCHAR(100)	The file name of the clip.
TStamp	VARCHAR(100)	The date and time when the clip was recorded, stored as text.
SName	VARCHAR(100)	The station name for the clip (written as text, not linked to RadioStation).

TEXTS	TEXT	The transcript or summary of what was recorded.
Categories	TEXT	The topics or categories for the clip.
DownloadLink	TEXT	The link where the clip can be downloaded.

Indexes

Index Name	Type	Purpose
PRIMARY	PRIMARY	The main index on the ID column. This identifies each clip uniquely.
TEXTS	FULLTEXT	Allows word searches inside the TEXTS column (used for text or transcript searching).
TEXTS_2	FULLTEXT	Duplicate or experimental full-text index for the same column.
TEXTS_3	FULLTEXT	Another full-text index, likely a duplicate.

Observations / Issues

1. **Not linked to RadioStation table.**
   The SName field only stores the station name as text. There is no official connection to the RadioStation table. This means the database doesn't "know" which station a clip belongs to, and errors can happen if names don't match exactly.

2. **TStamp stored as text.**
   The TStamp field is a VARCHAR(100), which means it is saved as text instead of a proper date/time format. This makes it hard for the database to sort clips by time or filter by date ranges.

3. **Duplicate full-text indexes.**
   The three full-text indexes (TEXTS, TEXTS_2, TEXTS_3) serve the same purpose and only one is needed. Having extra indexes adds unnecessary maintenance.

4. **Large text fields.**
   TEXT fields like TEXTS, Categories, and DownloadLink can store long text but take up more space and slow down searches when there are many rows.

5. **No NOT NULL rules.**
   Important fields like FName and TStamp could be left blank, which could cause missing data.

Recommendations

- Add a new column called **StationID** that links to the RadioStation table's ID.
- Convert **TStamp** from text (VARCHAR) to **DATETIME** so the database can correctly sort and filter by date and time.
- Keep one **FULLTEXT** index on TEXTS but remove duplicates (TEXTS_2, TEXTS_3).
- Add **indexes** on StationID and TStamp to speed up lookups by station and time.
- Use **NOT NULL** on important columns like FName, StationID, and TStamp.
- Move **Categories** into its own table for better organization and easier searches.

4. Relationship Mapping

Current Structure

RadioClips is not actually connected to RadioStation. The SName field simply stores the name of a station as plain text, which the database does not recognize as a true link.

Recommended Structure

Each clip should have a **StationID** column that connects to **RadioStation.ID**.
This creates a one-to-many relationship:

- One radio station can have many clips.
- Each clip belongs to one radio station.

Example Structure:

RadioStation:

ID	StationName
1	KCFR
2	KOA

RadioClips:

ID	FName	StationID
1	file1.txt	1
2	file2.txt	2

5. Performance and Index Analysis

Scenario	Current Issue	Recommended Solution
Search clips by station	No StationID column or index.	Add StationID and an index on it.
Sort clips by date/time	TStamp is text, not a real date.	Convert TStamp to DATETIME and index it.
Search text content	Multiple full-text indexes slow the table.	Keep only one full-text index.
Prevent duplicate station names	Already handled in RadioStation.	Keep the UNIQUE index on StationName.

6. Data Type Improvements

- Change TEXT fields (City, State, Country) to VARCHAR(100) since these are short words, it may be useful to do this now if DigiClips were to expand internationally and 30 may not be large enough.
- Change TStamp from VARCHAR(100) to DATETIME for proper date sorting.
- Add NOT NULL to important columns to prevent missing data.
- Keep one FULLTEXT index for text searching.

List of Essential Columns to be made NOT NULL

RadioStation table:

- ID
- StationName
- Country

RadioClips table:

- ID
- FName
- TStamp (after converting to DATETIME)
- StationID (after adding and linking it as a foreign key)

7. Relationship Diagram

Table Name	Key Column	Key Type	Relationship Type	Connected To (Other Table → Column)	Description (in simple terms)
RadioStation	ID	Primary Key	**One-to-Many**	Connected from → RadioClips.StationID	Each radio station has a unique ID. One station can have many clips linked to it.
RadioClips	StationID	Foreign Key	**Many-to-One**	Connects to → RadioStation.ID	Each clip belongs to one radio station. This link shows which station created or aired the clip.

8. Summary of Findings

Area	Finding
Data structure	Tables store correct data but are not connected properly.
Relationships	Only text-based, not true database links.
Data types	Some fields use large text unnecessarily.
Indexes	RadioClips has 4 indexes (1 primary, 3 full-text). RadioStation has 2 (1 primary, 1 unique).
Constraints	Missing NOT NULL and foreign key rules.

Performance	Searches and lookups will slow down as data grows.

9. Recommendations Summary

1. Add **StationID** to RadioClips and connect it to RadioStation.ID.
2. Convert **TStamp** to **DATETIME**.
3. Add **NOT NULL** to key columns.
4. Keep only one **FULLTEXT** index on TEXTS.
5. Add **indexes** on StationID and TStamp.
6. Change **TEXT** to **VARCHAR(100)** where appropriate.
7. Add **created_at** and **updated_at** to both tables.
8. Keep **StationName_UNIQUE** in RadioStation to prevent duplicates.

10. Conclusion

The current database design successfully stores information about radio stations and their clips but lacks strong connections between the two tables. The RadioClips table relies on text matching instead of a real link, and several columns use inefficient data types. By adding proper relationships, removing duplicate indexes, and improving data rules, the database will become faster, cleaner, and more reliable for long-term use and analysis.