Reporting Tables Analysis

Sprint 3: NewspaperResults, MagazineResults, and NewspaperFeeds

Overview

This report explains the structure and relationships between the tables NewspaperFeeds, MagazineFeeds, Sentiment_Analysis_keywords, SentimentAnlyResult. It describes what each table stores, the indexes they use, what issues were found, and how to improve them.

NewspaperResults and MagazineResults

The **NewspaperFeeds** table keeps information about each newspaper article, such as the link of the article, name of the newspaper and link of the newspaper. It is a master list of all the newspaper articles.

Column Name	Data Type	Description
ID	INT	A unique link to the article. (Not NULL)
Title	VARCHAR(1000)	The name of the newspaper.
Author	VARCHAR(100)	The name of the author
Summary	VARCHAR(5000)	The summary of the article
PublishDate	DATETIME	The date when the article was published
NewspaperLink / MagazineLink	VARCHAR(1000)	Link of the newspaper
ImageURL	VARCHAR(1000)	Url of the images
UpdateDate	DATETIME	When was the article last updated

GUID	VARCHAR(1000)	GUID of the resource
AddedDate	DATETIME	When was the article added

Indexes

Index Name	Type	Purpose
PRIMARY	PRIMARY	Exists on the ID column and ensures unique record identification.
GUID_UNIQUE	UNIQUE	Exists on the GUID column. It avoids duplicate ingestion of articles.
idx_Newspaper_Results_Summary / idx_MagazineResults	FULLTEXT	Enables full-text search within article summaries

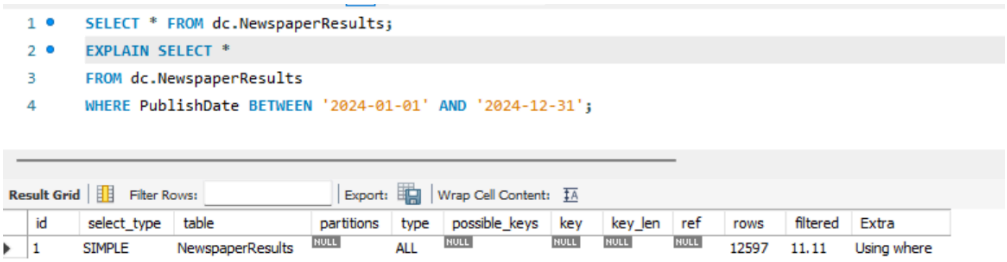
Stored Procedures

Name	Purpose
Get_ID_and_GUID_Newspaper / Get_ID_and_GUID_Magazine	Returns specific newspapers and magazine based on id and guid
MagazineResult / NewspaperResult	Adds a row to the tables
Magazine_Search / Newspaper_Search	Searches the text summaries
Update_Magazine / Update_Newspaper	Updates the newspapers and magazines cited and then updates the the column “updated_date”

Observations / Issues

1. Both NewspaperResults and MagazineResults are not connected to any tables. There are no foreign keys.
2. **Schema Consistency:**
   The two tables share nearly identical structures, which supports unified reporting but also introduces redundancy that could be normalized in future releases.
3. **Data Type Validation:**
   All columns use appropriate types for their data. However, VARCHAR(1000) for GUIDs and links may be over allocated, optimization to VARCHAR(255) would suffice.

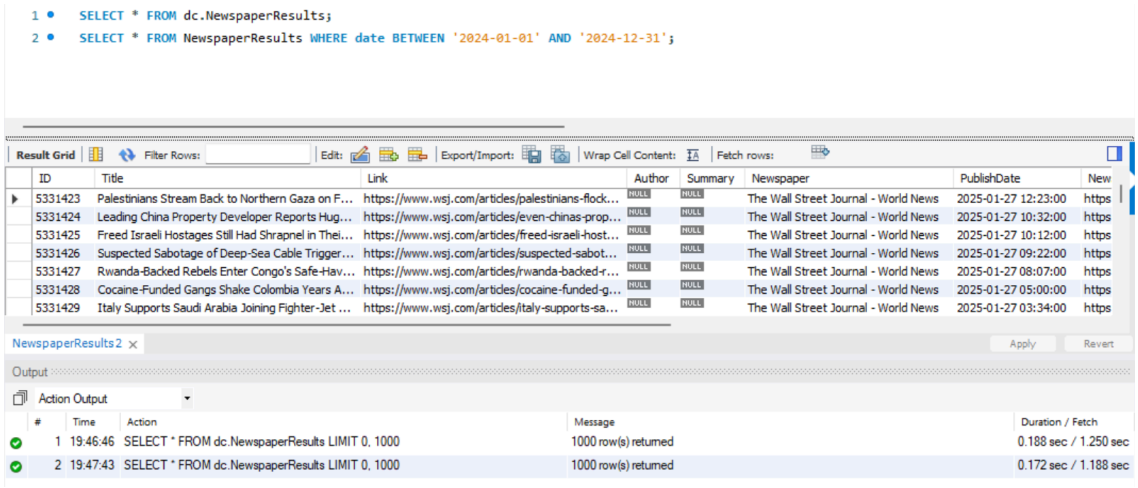
Data query:



```
SELECT * FROM dc.NewspaperResults;
EXPLAIN SELECT *
FROM dc.NewspaperResults
WHERE PublishDate BETWEEN '2024-01-01' AND '2024-12-31';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	NewspaperResults	NULL	ALL	NULL	NULL	NULL	NULL	12597	11.11	Using where



```
SELECT * FROM dc.NewspaperResults;
SELECT * FROM NewspaperResults WHERE date BETWEEN '2024-01-01' AND '2024-12-31';
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Fetch rows:

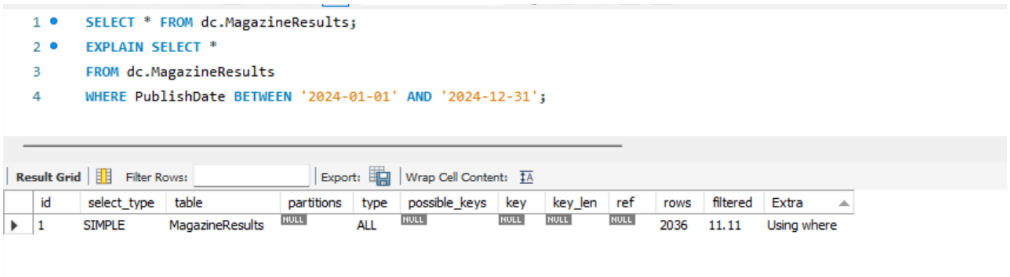
ID	Title	Link	Author	Summary	Newspaper	PublishDate	New
5331423	Palestinians Stream Back to Northern Gaza on F…	https://www.wsj.com/articles/palestinians-flock…	NULL	NULL	The Wall Street Journal - World News	2025-01-27 12:23:00	https
5331424	Leading China Property Developer Reports Hug…	https://www.wsj.com/articles/even-chinas-prop…	NULL	NULL	The Wall Street Journal - World News	2025-01-27 10:32:00	https
5331425	Freed Israeli Hostages Still Had Shrapnel in Thei…	https://www.wsj.com/articles/freed-israeli-host…	NULL	NULL	The Wall Street Journal - World News	2025-01-27 10:12:00	https
5331426	Suspected Sabotage of Deep-Sea Cable Trigger…	https://www.wsj.com/articles/suspected-sabot…	NULL	NULL	The Wall Street Journal - World News	2025-01-27 09:22:00	https
5331427	Rwanda-Backed Rebels Enter Congo's Safe-Hav…	https://www.wsj.com/articles/rwanda-backed-r…	NULL	NULL	The Wall Street Journal - World News	2025-01-27 08:07:00	https
5331428	Cocaine-Funded Gangs Shake Colombia Years A…	https://www.wsj.com/articles/cocaine-funded-g…	NULL	NULL	The Wall Street Journal - World News	2025-01-27 05:00:00	https
5331429	Italy Supports Saudi Arabia Joining Fighter-Jet …	https://www.wsj.com/articles/italy-supports-sa…	NULL	NULL	The Wall Street Journal - World News	2025-01-27 03:34:00	https

NewspaperResults 2 | Apply | Revert

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	19:46:46	SELECT * FROM dc.NewspaperResults LIMIT 0, 1000	1000 row(s) returned	0.188 sec / 1.250 sec
2	19:47:43	SELECT * FROM dc.NewspaperResults LIMIT 0, 1000	1000 row(s) returned	0.172 sec / 1.188 sec



```
SELECT * FROM dc.MagazineResults;
EXPLAIN SELECT *
FROM dc.MagazineResults
WHERE PublishDate BETWEEN '2024-01-01' AND '2024-12-31';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	MagazineResults	NULL	ALL	NULL	NULL	NULL	NULL	2036	11.11	Using where

- Data Query Speed: Returned 1000 rows in 0.172 seconds
- There is no sentiment filer in both of these tables
- For both MagazineResults and NewspaperResult, there is no index at PublishDate.

Recommendations

- Adding an index on PublishDate. Currently EXPLAIN output shows a full table scan (type = ALL) with no index being used. This indicates that MySQL is scanning all ~2,000+ rows even though the query filters on PublishDate.
- FULLTEXT indexes can become large and should be monitored for performance overhead during inserts/updates.

- Keep GUID_UNIQUE index as it ensures data integrity.

NewspaperFeeds

The **NewspaperFeeds** table keeps information about each newspaper article, such as the link of the article, name of the newspaper and link of the newspaper. It is a master list of all the newspaper articles.

Column Name	Data Type	Description
URL	VARCHAR(1000) (Primary Key)	A unique link to the article. (Not NULL)
NewspaperName	VARCHAR(150)	The name of the newspaper.
NewspaperLink	VARCHAR(150)	Link of the newspaper in which the article was published

Indexes

Index Name	Type	Purpose

PRIMARY	PRIMARY, BTree on URL	The main index automatically created on the URL column. The link is unique and is searchable.
URL_UNIQUE	BTree on URL	Adds a uniqueness to the url column

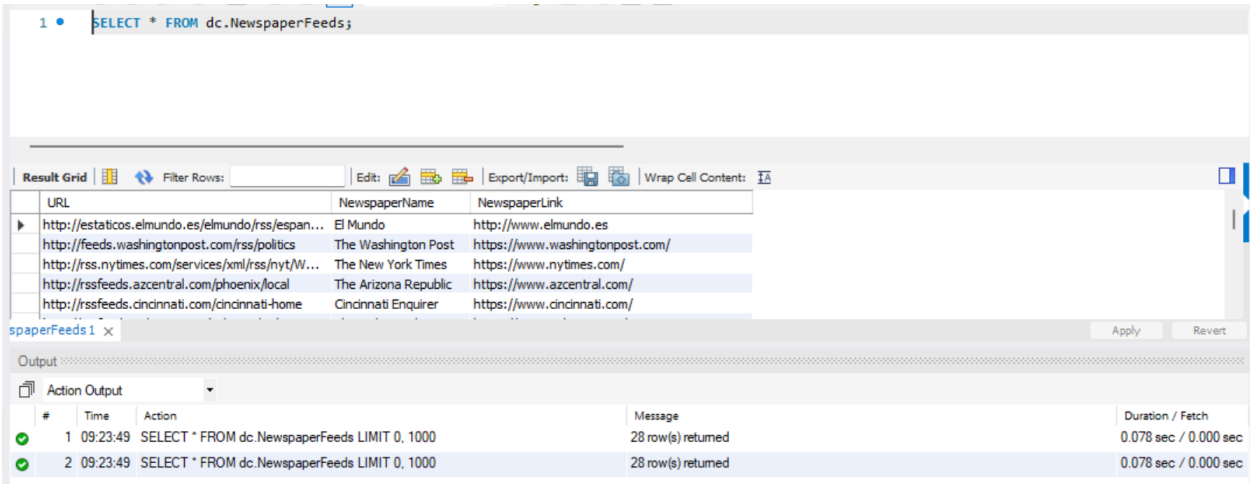
Stored Procedures

Name	Purpose
Get_Magazine_Feeds / Get_Newspaper_Feeds	Returns the tables with url of the article, name and link of the newspaper/magazine

Observations / Issues

4. **The primary key being the url**
   URLs are long text fields and using them as primary keys leads to slow indexing and higher memory usage.
5. **URL_UNIQUE index on the same column as the PRIMARY**
   The PRIMARY KEY already implies uniqueness. The UNIQUE constraint adds a duplicate index enforcing the same rule.
6. **No foreign keys. The table is not connected to anything.**
   It is very hard to get information about this table and no other tables can access this table neither for receiving or storing data. The data is isolated.

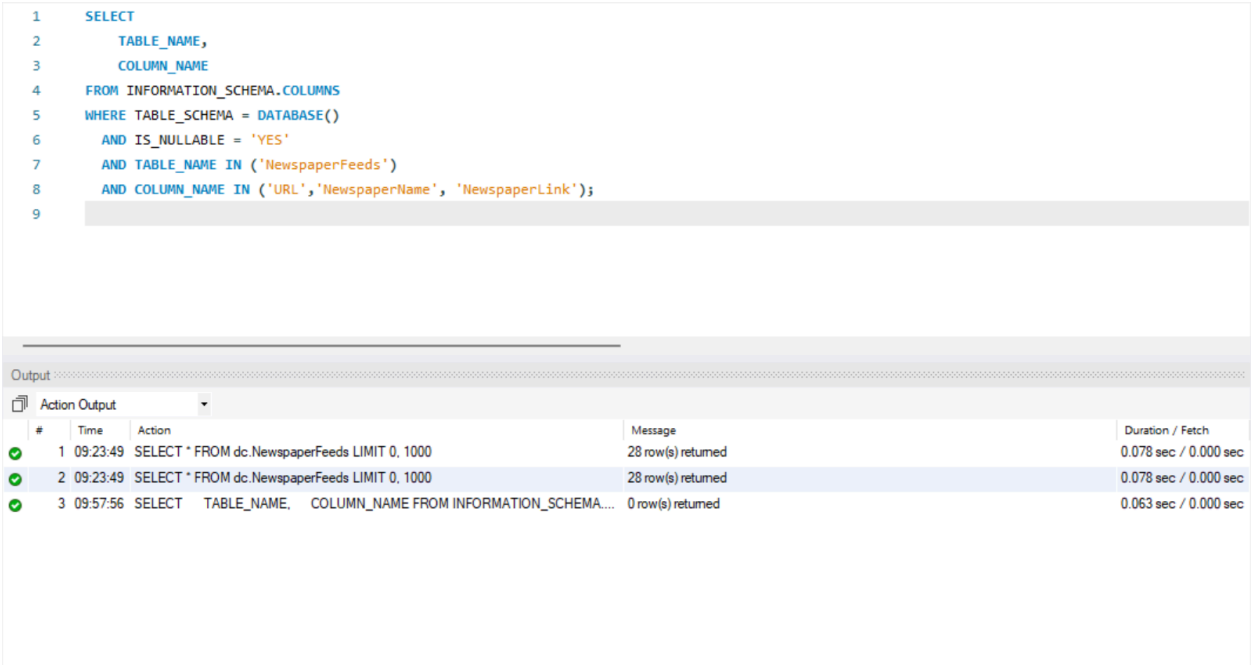
Data query:



Observation: The query worked and returned 28 rows in 0.078

Issue: There are only 28 rows of data in this table and because the data is isolated most of its links are outdated.

Null and Missing values query:



Recommendations

- URL_UNIQUE index can be dropped

- Reconsider the usage of URL as the PRIMARY index

- Reoptimize the connection between the tables and add foreign keys accordingly.

Recommended Structure

- Create a unified relationship structure where feeds (newspaper and magazine) act as the core data sources, and sentiment/keyword tables depend on them.
- NewspaperFeeds and MagazineFeeds: Act as master content sources.
- Sentiment_Analysis_Keywords: Links to feed tables through FeedID (foreign key).
- SentimentAnlyResult: Links to both FeedID (source article) and KeywordID (keyword used).

One-to-Many Relationships:
- One article (in NewspaperFeeds or MagazineFeeds) → Many sentiment keywords
- One keyword → Many sentiment results.
- One article → Many sentiment results (indirectly via keywords).

Benefits of This Structure
- Enables sentiment analysis across all article types.
- Ensures referential integrity (no orphan sentiment or keyword entries).
- Simplifies analytical queries (e.g., sentiment trends by publisher or keyword).
- Improves maintainability and scalability as data volume grows.

Data Type and Index Improvements

- Use INT AUTO_INCREMENT for all ID fields.
- Convert URL fields from primary to unique indexes.
- Change Finished_at and Submission_Time to DATETIME.
- Replace long TEXT fields with VARCHAR(255) where appropriate.
- Use ENUM or JSON for categorical and structured sentiment data.

Indexes to Add
- On FeedID in Sentiment_Analysis_Keywords and SentimentAnlyResult
- On Keyword in Sentiment_Analysis_Keywords
- On Finished_at in SentimentAnlyResult
- On ArticleID in both feed tables

Recommendations Summary

1. Introduce surrogate keys (ArticleID, FeedID, KeywordID, ResultID).
2. Establish foreign key relationships among all four tables.

3. Convert text timestamps to DATETIME.
4. Add NOT NULL constraints on essential columns.
5. Simplify indexes by keeping only the most relevant.
6. Normalize repeated large text fields.

Conclusion

The reporting tables provide useful data independently but lack the relational structure needed for efficient cross analysis. By establishing foreign key relationships, and optimizing data types and indexes, the database can support faster queries, better integrity, and scalable analytics across newspaper and magazine content.