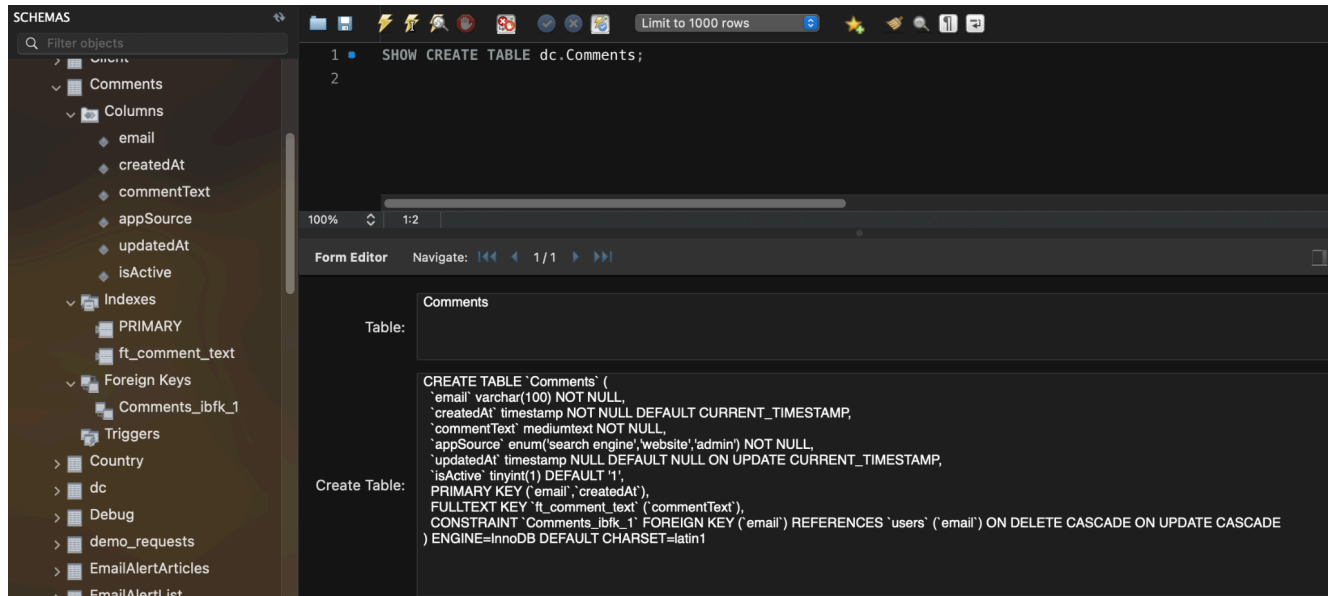


Overview

The **Comments** table stores feedback, text entries, and notes made by users or system processes. It keeps details such as who made the comment (email), what the comment says, where it came from (app source), and when it was created or updated. It also tracks whether each comment is active.



Purpose

The purpose of the **Comments** table is to record and manage user or system-generated comments for different applications and platforms. It supports data tracking for reviews, feedback, or alerts coming from multiple sources (like websites, apps, or broadcasts).

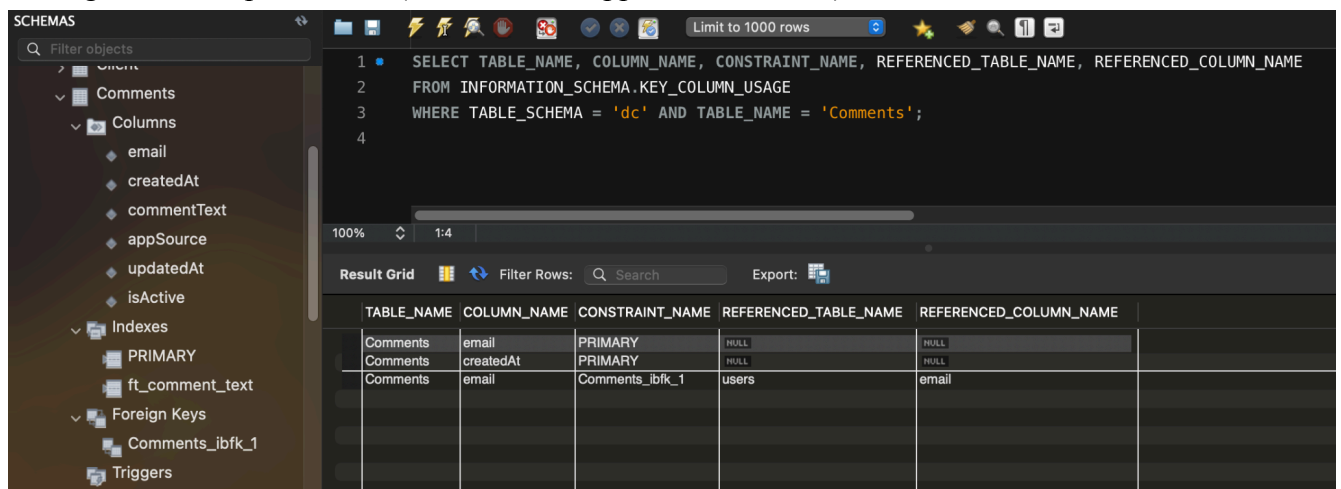
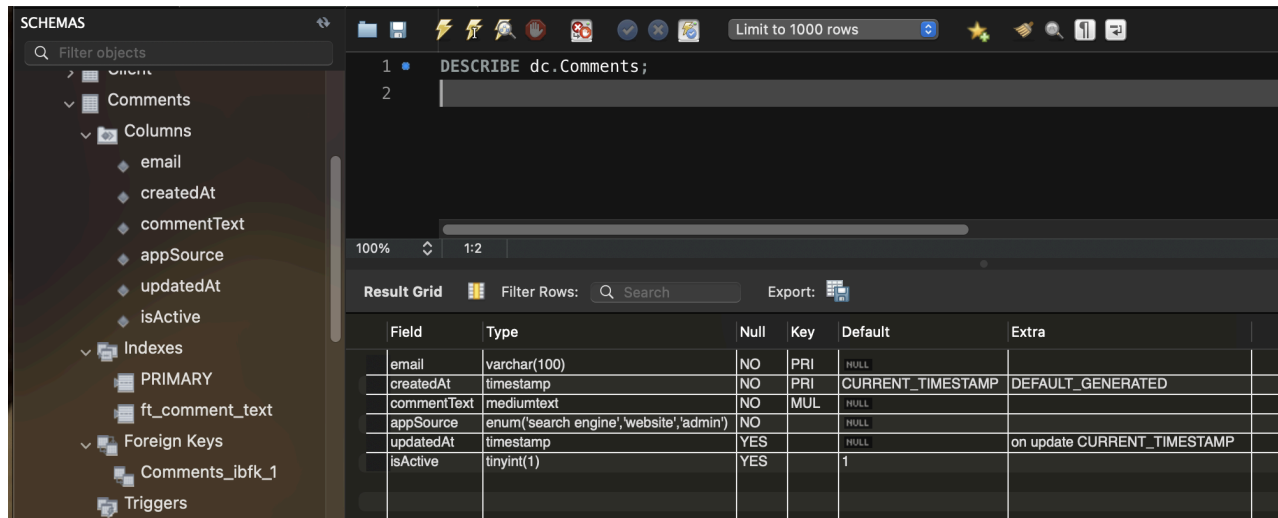


Table Columns

Column Name	Data Type	Description
email	VARCHAR(100)	The email address of the user who made the comment.
createdAt	TIMESTAMP	The date and time when the comment was first created
commentText	TEXT	The actual content of the comment
appSource	ENUM	The platform where the comment was made (for example, Web, Mobile, or API).
updatedAt	TIMESTAMP	The date and time when the comment was last updated.
isActive	TINYINT	A flag showing if the comment is active (1 = active, 0 = inactive)

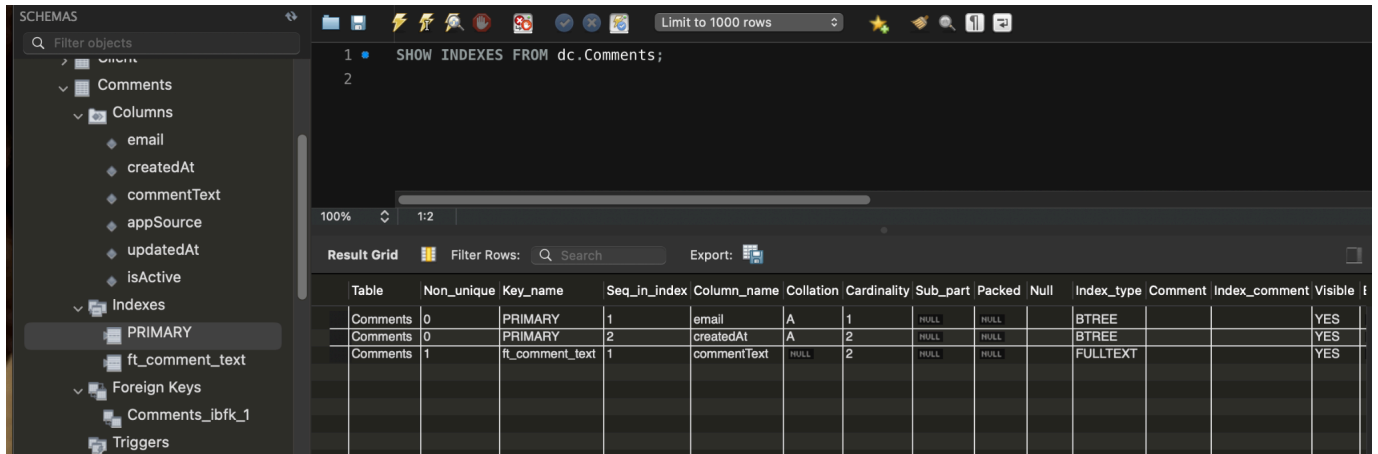


The screenshot shows a database management tool interface. On the left, a 'SCHEMAS' panel displays a tree view of the database structure, including 'Comments' and its columns. The main panel shows the 'DESCRIBE dc.Comments;' command executed, resulting in a table with the following columns: Field, Type, Null, Key, Default, and Extra. The data is as follows:

Field	Type	Null	Key	Default	Extra
email	varchar(100)	NO	PRI	HULL	
createdAt	timestamp	NO	PRI	CURRENT_TIMESTAMP	DEFAULT_GENERATED
commentText	mediumtext	NO	MUL	HULL	
appSource	enum('search engine','website','admin')	NO		HULL	
updatedAt	timestamp	YES		HULL	on update CURRENT_TIMESTAMP
isActive	tinyint(1)	YES		1	

Indexes and Keys

- Primary Key: email, createdAt
- Foreign Keys:
 - Fk_comment_text: full text index to speed up searches through commentText



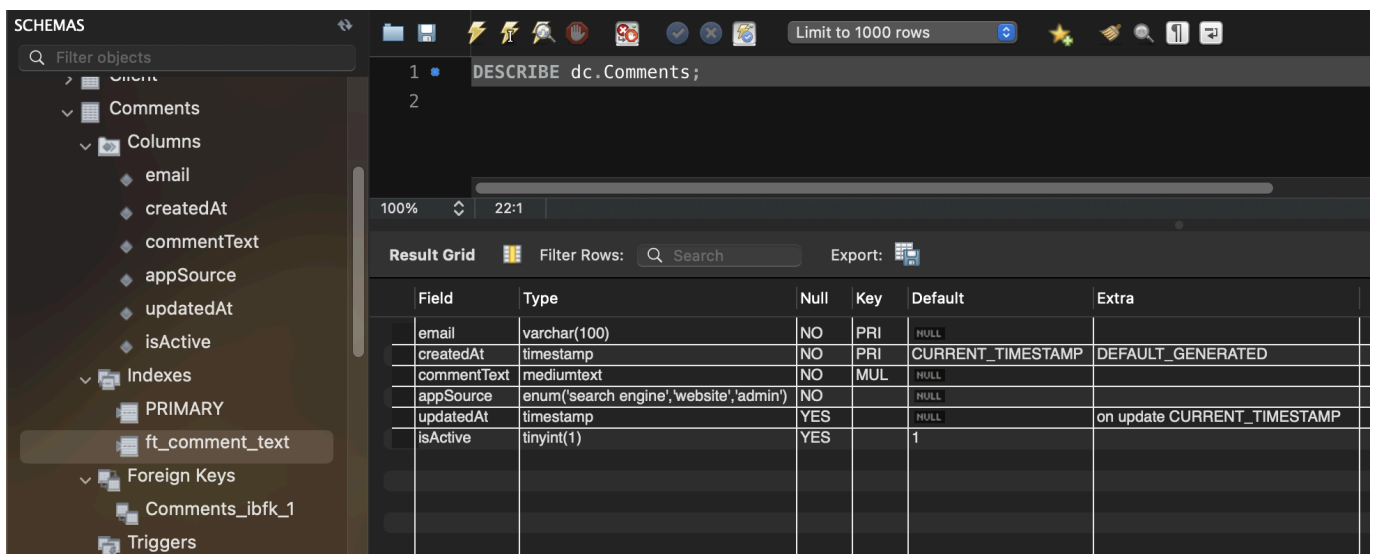
1 SHOW INDEXES FROM dc.Comments;

2

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible
Comments	0	PRIMARY	1	email	A	1				BTREE			YES
Comments	0	PRIMARY	2	createdAt	A	2				BTREE			YES
Comments	1	ft_comment_text	1	commentText		2				FULLTEXT			YES

Observations and Issues

1. **Table Structure is Clean:** The table contains all necessary fields for tracking and managing comments. There are no missing required columns, and data types are appropriate for the stored information.

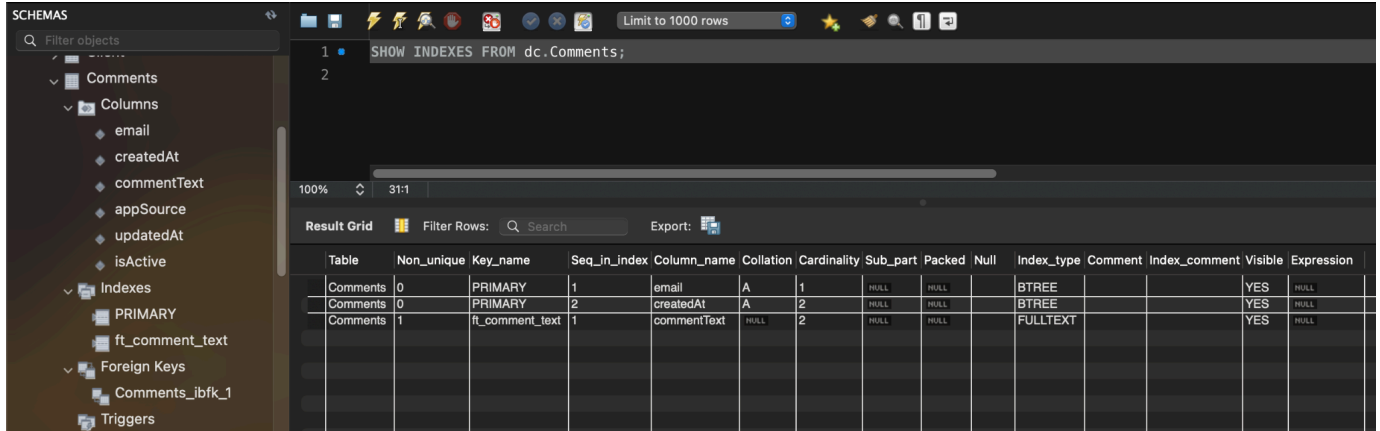


1 DESCRIBE dc.Comments;

2

Field	Type	Null	Key	Default	Extra
email	varchar(100)	NO	PRI		
createdAt	timestamp	NO	PRI	CURRENT_TIMESTAMP	DEFAULT_GENERATED
commentText	mediumtext	NO	MUL		
appSource	enum('search engine','website','admin')	NO			
updatedAt	timestamp	YES			on update CURRENT_TIMESTAMP
isActive	tinyint(1)	YES		1	

2. **Full Text Search Enabled:** The **ft_comment_text** index allows searching within the **commentText** field efficiently. This improves lookup speed for keyword or phrase searches.



1 SHOW INDEXES FROM dc.Comments;

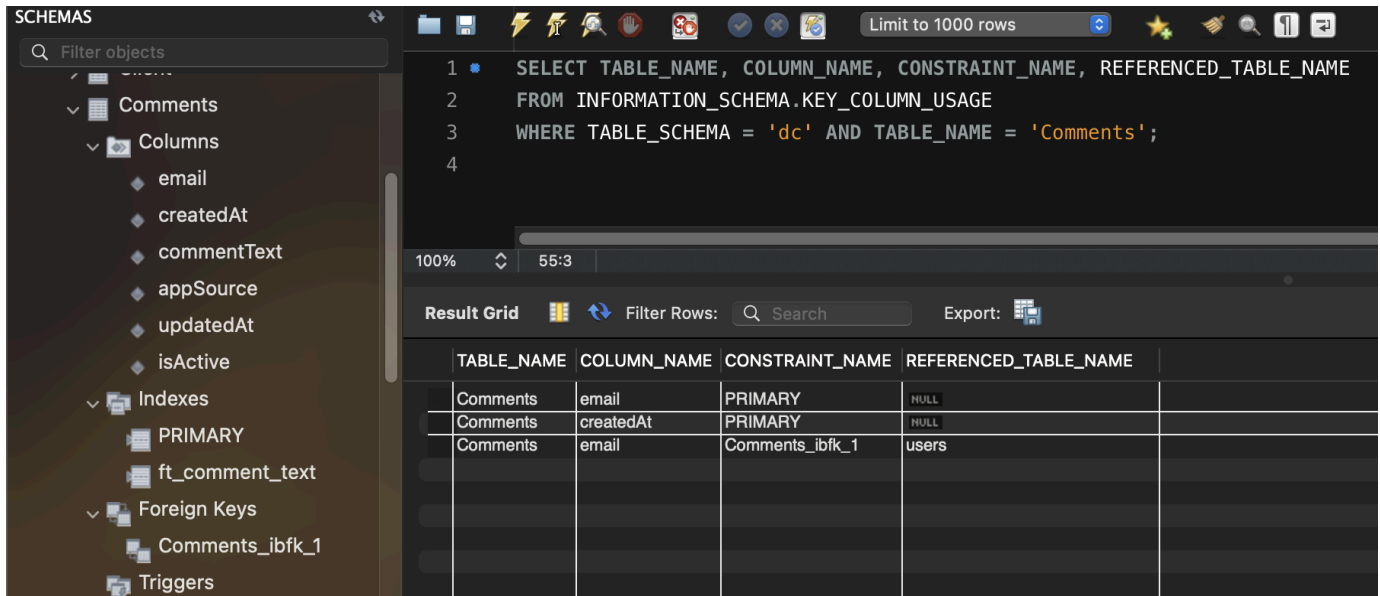
2

100% 31:1

Result Grid Filter Rows: Search Export:

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Expression
Comments	0	PRIMARY	1	email	A	1				BTREE			YES	
Comments	0	PRIMARY	2	createdAt	A	2				BTREE			YES	
Comments	1	ft_comment_text	1	commentText		2				FULLTEXT			YES	

3. **Foreign Key Relationship Present:** The **Comments** table includes one foreign key constraint named **Comments_ibfk_1**. This links the **email** column in **dc.Comments** to the **email** column (or corresponding key) in the **users** table. This ensures that every comment is tied to a valid user record, improving data consistency and traceability.



1 SELECT TABLE_NAME, COLUMN_NAME, CONSTRAINT_NAME, REFERENCED_TABLE_NAME

2 FROM INFORMATION_SCHEMA.KEY_COLUMN_USAGE

3 WHERE TABLE_SCHEMA = 'dc' AND TABLE_NAME = 'Comments';

4

100% 55:3

Result Grid Filter Rows: Search Export:

TABLE_NAME	COLUMN_NAME	CONSTRAINT_NAME	REFERENCED_TABLE_NAME
Comments	email	PRIMARY	
Comments	createdAt	PRIMARY	
Comments	email	Comments_ibfk_1	users

Recommendations:

1. **Validate Email References:** Since **email** is now a foreign key linked to the **users** table (**Comments_ibfk_1**), ensure all inserted emails exist in **users**.
2. **Determine how Timestamps are Set:** Look and see how the timestamps are currently set. Ensure that they automatically populate to ensure consistency in time tracking for creation and edit of comments
3. **Regular Backup:** Since comments can grow large quickly, back up this table regularly to improve performance.
4. **Archive Old Comments:** Archive or delete rows where **inactive=0** and **updatedAt** is older than 2 years old. This allows for more efficient search operations