Data Science Apprenticeship Challenge

Section A: Initialization

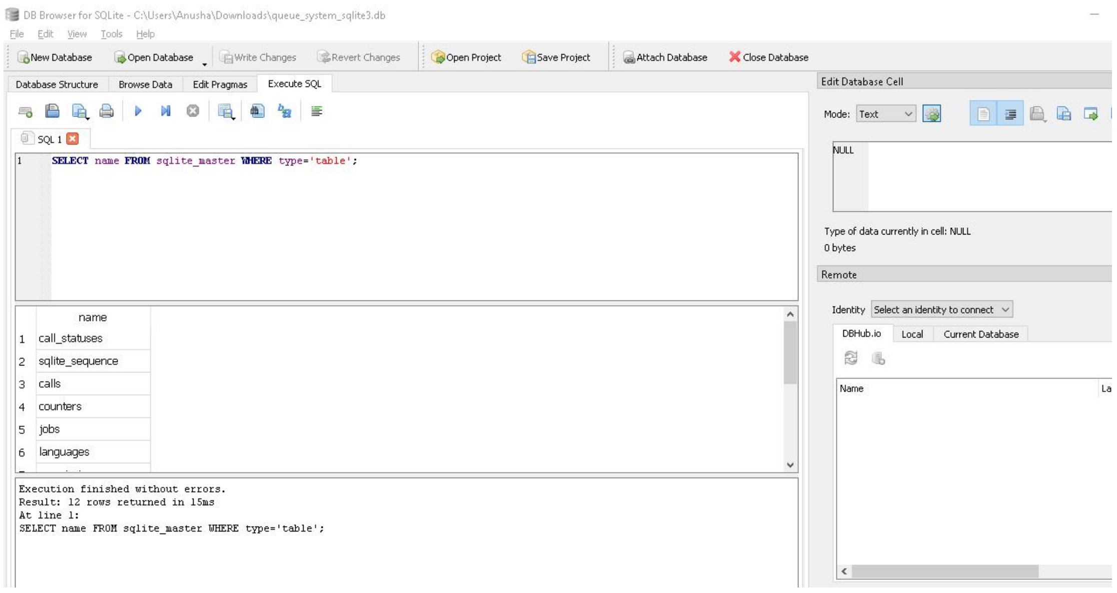
1. Sample Data

Objective: Loading the database and verifying the table names

Executing the query:

SELECT name FROM sqlite_master WHERE type='table';

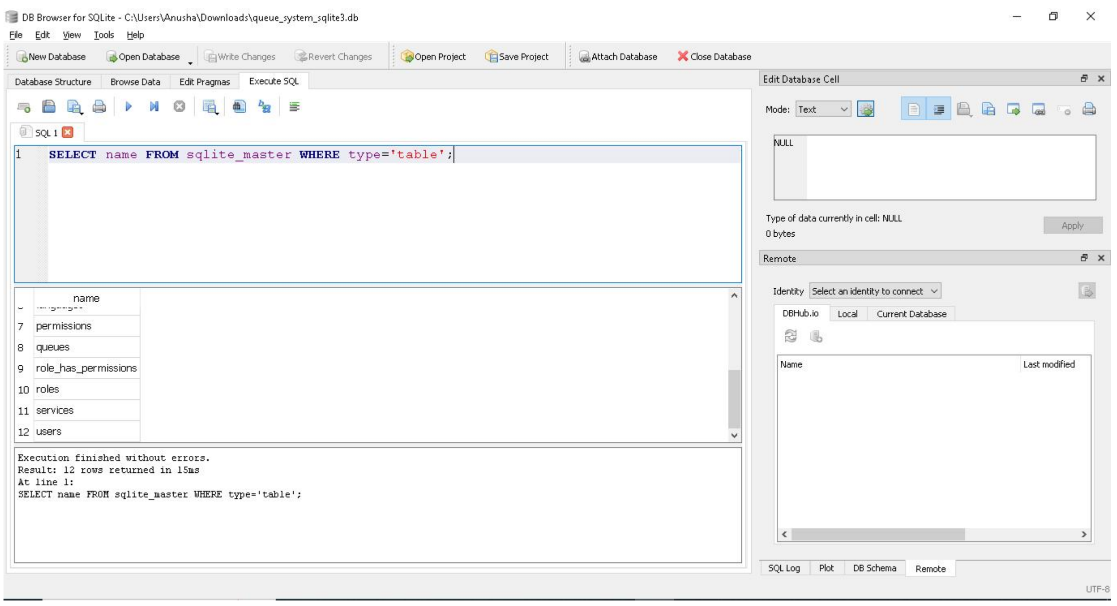
Result:



```
SELECT name FROM sqlite_master WHERE type='table';
```

	name
1	call_statuses
2	sqlite_sequence
3	calls
4	counters
5	jobs
6	languages

```
Execution finished without errors.
Result: 12 rows returned in 15ms
At line 1:
SELECT name FROM sqlite_master WHERE type='table';
```



```
SELECT name FROM sqlite_master WHERE type='table';
```

	name
7	permissions
8	queues
9	role_has_permissions
10	roles
11	services
12	users

```
Execution finished without errors.
Result: 12 rows returned in 15ms
At line 1:
SELECT name FROM sqlite_master WHERE type='table';
```

2. Tables Definition and Their Relation

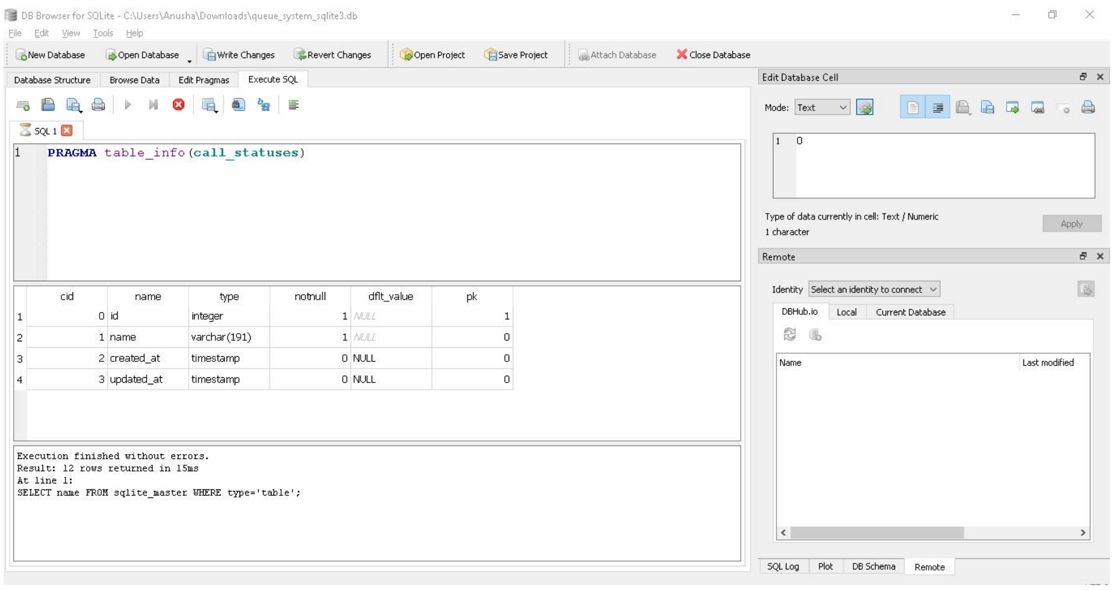
Objectives: Understand and explore the database schema.

Steps:

Explore Schema:
Executing the query:

PRAGMA table_info(call_statuses);

Result:



```
PRAGMA table_info(call_statuses)
```

	cid	name	type	notnull	dflt_value	pk
1	0	id	integer	1	NULL	1
2	1	name	varchar(191)	1	NULL	0
3	2	created_at	timestamp	0	NULL	0
4	3	updated_at	timestamp	0	NULL	0

```
Execution finished without errors.
Result: 12 rows returned in 15ms
At line 1:
SELECT name FROM sqlite_master WHERE type='table';
```

Findings:

Table Name: call_statuses

id: A column with values of a integer datatype that serves as a primary key , uniquely identifying each row in the table.It is set to Not Null, which means it must have a value.

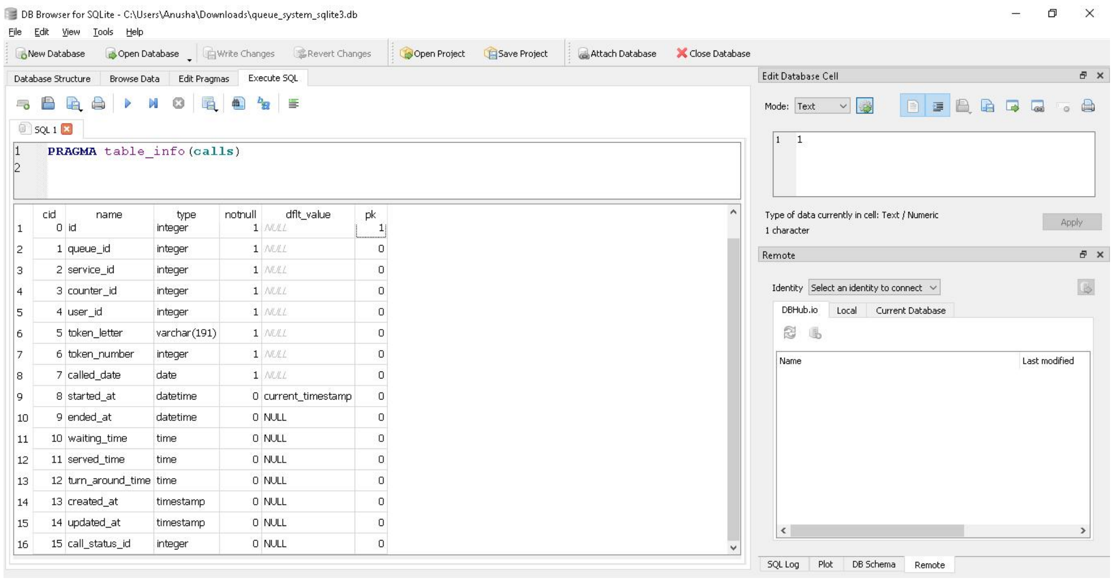
name: A column with values of a string datatype allowing the characters upto 191, which must have a value and cannot be set to Not Null.

created_at: A Column with values of a timestamp type.It can be set to Null.

updated_at: A column with values of a timestamp type and records when row was last updated,it can aswell be set to Null.

PRAGMA table_info(calls);

Result:



```
PRAGMA table_info(calls)
```

	cid	name	type	notnull	dflt_value	pk
1	0	id	integer	1	NULL	1
2	1	queue_id	integer	1	NULL	0
3	2	service_id	integer	1	NULL	0
4	3	counter_id	integer	1	NULL	0
5	4	user_id	integer	1	NULL	0
6	5	token_letter	varchar(191)	1	NULL	0
7	6	token_number	integer	1	NULL	0
8	7	called_date	date	1	NULL	0
9	8	started_at	datetime	0	current_timestamp	0
10	9	ended_at	datetime	0	NULL	0
11	10	waiting_time	time	0	NULL	0
12	11	served_time	time	0	NULL	0
13	12	turn_around_time	time	0	NULL	0
14	13	created_at	timestamp	0	NULL	0
15	14	updated_at	timestamp	0	NULL	0
16	15	call_status_id	integer	0	NULL	0

Mode: Text

Type of data currently in cell: Text / Numeric
1 character

Identity: Select an identity to connect

DBHub.io | Local | Current Database

Name | Last modified

Findings:

Table Name: calls

id: A unique integer identifier for each call record, it serves as a primary key.It can not be null.

queue_id: An integer representing the queue associated with the call.It cannot be null, and ensures every call is linked to a specific queue.

service_id: An integer that links the call to a specific service.This column can not be null.

counter_id: An integer that identifies the specific counter handling the each calls.This column as well can not be null.

user_id: An integer representing the user or agent managing the calls.This aswell can not be set to null.

token_letter: A column of string type and allowing upto 191 characters.It cannot be set to null.

token_number: An integer column representing the number part of tokens in a eac call.It cannot be null.

callled_date: A date column which records the date in which call was made.it cannot be null, since every call must have date associated with it.

started_date: A date column which records the date in which call begin.It has default value of current timestamp.

ended_date: A date column which records the date when call was ended.it is optional and can aswell be set to null.

waiting_time: A time column which records the duration of a call which was spent before the call was answered or addressed.it can be null.

served_time: A time column which records the duration of the serving of the calls.it can be set to null.

turn_around_time: A time fields which records the duration between the initiation of a call and its completion, it can be null.

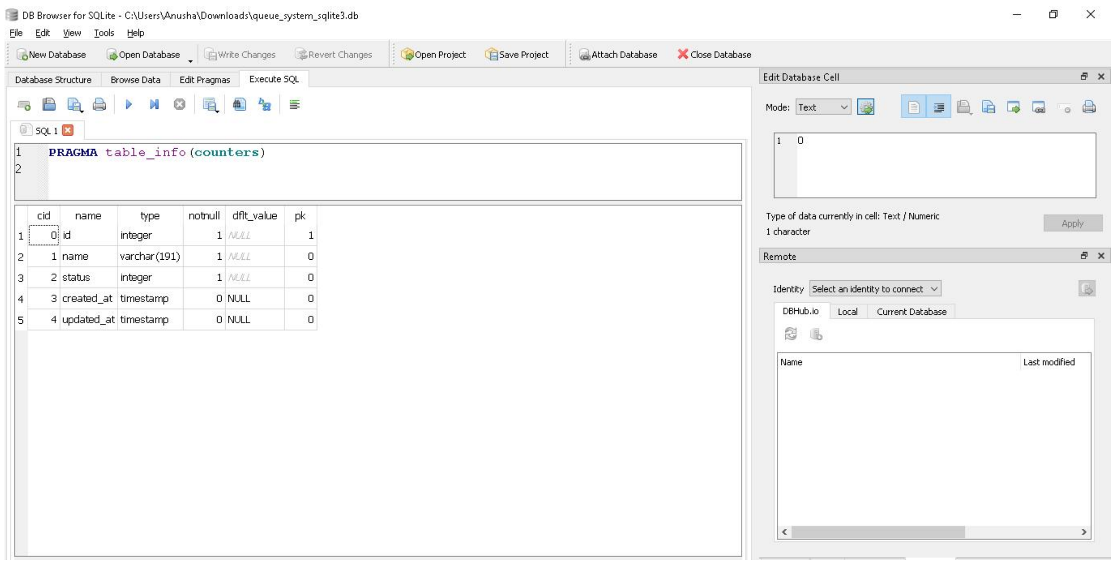
created_at: A time field which records the timestamp of insertion of the row.It can be null.

updated_at: A time field which records when the row was last updated at, it can aswell be null.

call_status_id: An integer column which links the call to a specific status. It can be null.It refrences the entry in another table call_statuses.

PRAGMA table_info(counters);

Result:



```
PRAGMA table_info(counters)
```

	cid	name	type	notnull	dflt_value	pk
1	0	id	integer	1	*NULL*	1
2	1	name	varchar(191)	1	*NULL*	0
3	2	status	integer	1	*NULL*	0
4	3	created_at	timestamp	0	NULL	0
5	4	updated_at	timestamp	0	NULL	0

Type of data currently in cell: Text / Numeric
1 character

Findings:

Table Name: counters

id: An integer column which serves as a primary key,and uniquely identifies each row in the table.This field cannot be null.

name: A string column alllowing upto 191 characters and represents the name of counter.It aswell cannot be null.

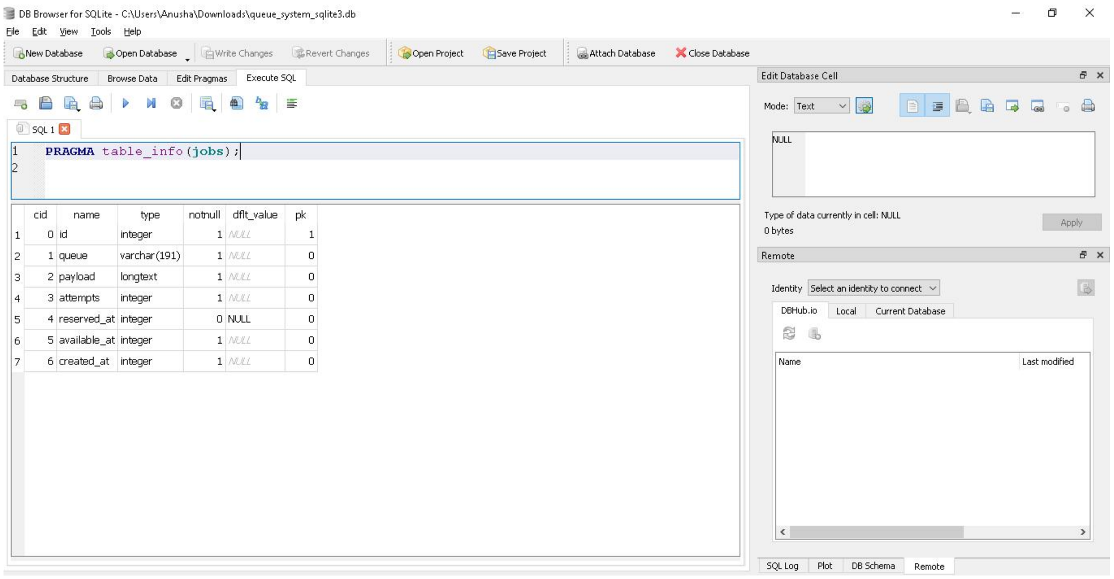
status: An integer column which indicates current status of counter.It aswell cannot be null and represents the status as active or inactive.

created_at: A timestamp field which records when the row was last created at.It can be null.

updated_at: A timestamp that recors when row was last updated at.It aswell can be null.

PRAGMA table_info(jobs);

Result:



```
PRAGMA table_info(jobs);
```

	cid	name	type	notnull	dflt_value	pk
1	0	id	integer	1	NULL	1
2	1	queue	varchar(191)	1	NULL	0
3	2	payload	longtext	1	NULL	0
4	3	attempts	integer	1	NULL	0
5	4	reserved_at	integer	0	NULL	0
6	5	available_at	integer	1	NULL	0
7	6	created_at	integer	1	NULL	0

Findings:

Table Name: jobs

id: An integer column which serves as a primary key for the table and uniquely identifying each entry in the table.This column cannot be null.

queue: A string column allowing upto 191 characters and represents the name of the queue to which job belongs.It cannot be null.

payload: A longtext column containing the data required to process the job.It cannot be null.

attempts: An integer column which is used to track the number of times the job has been attempted .It aswell cannot be set to null.

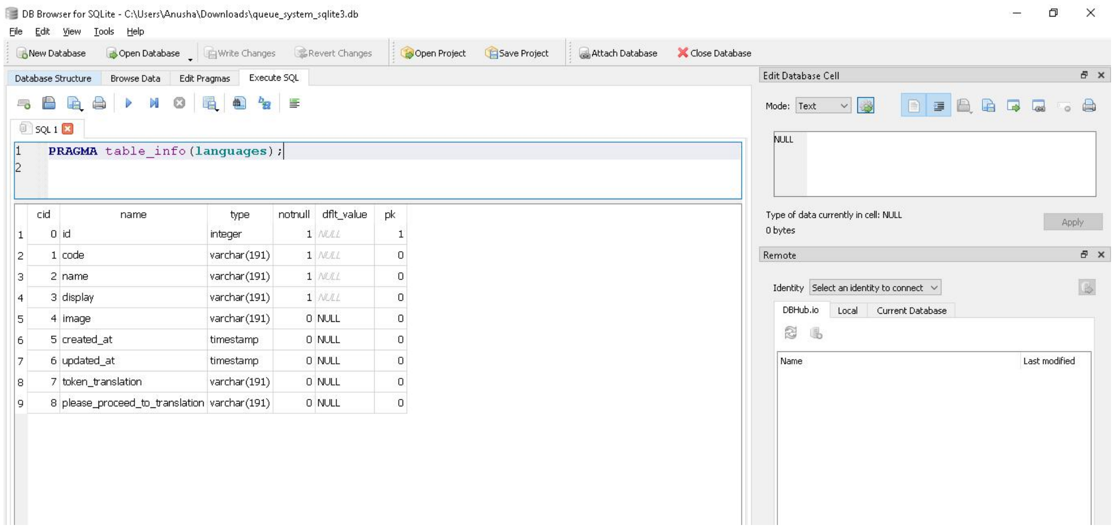
reserved_at: An integer column which records the timestamp when the job was reserved.It can be set to null.

available_at: An integer column which stores the timestamp when the job was available.It can be null.

created_at: It cannot be null and records when job was created.

PRAGMA table_info(languages);

Result:



```
PRAGMA table_info(languages);
```

	cid	name	type	notnull	dflt_value	pk
1	0	id	integer	1	NULL	1
2	1	code	varchar(191)	1	NULL	0
3	2	name	varchar(191)	1	NULL	0
4	3	display	varchar(191)	1	NULL	0
5	4	image	varchar(191)	0	NULL	0
6	5	created_at	timestamp	0	NULL	0
7	6	updated_at	timestamp	0	NULL	0
8	7	token_translation	varchar(191)	0	NULL	0
9	8	please_proceed_to_translation	varchar(191)	0	NULL	0

Mode: Text

NULL

Type of data currently in cell: NULL
0 bytes

Identity: Select an identity to connect

DBHub.io | Local | Current Database

Name	Last modified

Findings:

Table Name: languages

id: An integer column which is a primary key, uniquely identifying entry of each language in the table.It cannot be null.

code: A string column allowing the variable length of upto 191 characters and used to uniquely identify languages.

name: A string column allowing the variable length of upto 191 characters and contains the name of language.It cannot be null.

display: A string column allowing the variable length of upto 191 characters and specifies how the languages name should be shown.It cannot be null.

image: A string column allowing the variable length of upto 191 characters and stores url of image.It can be null.

created_at: A timestamp column which records the timestamp when the entry of language was made.It can be set to null.

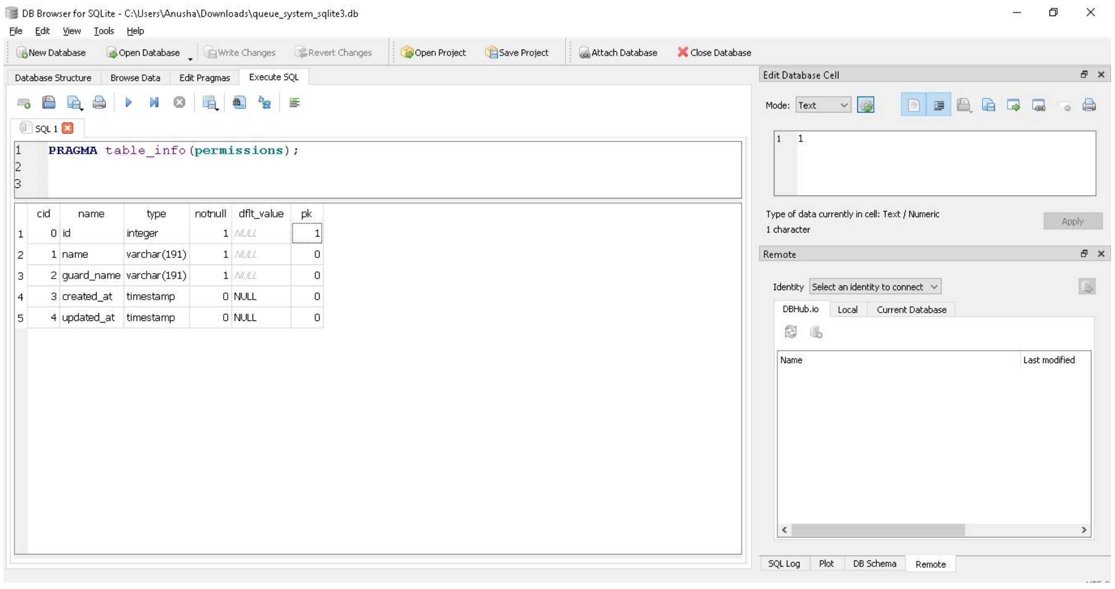
updated_at: A timestamp column which records the last timestamp when entry of languages was updated.This aswell can be set to null.

token_translation: A string column allowing variable length of characters upto 191.it can be null if no token is defined.

please_proceed_to_translation: A string column allowing the variable character upto length of 191.it can aswell be null.

PRAGMA table_info(permissions);

Result:


DB Browser for SQLite - C:\Users\Anusha\Downloads\queue_system_sqlite3.db
File Edit View Tools Help
New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database
Database Structure Browse Data Edit Pragmas Execute SQL
SQL 1
Edit Database Cell
Mode: Text
Type of data currently in cell: Text / Numeric
1 character
Apply
Remote
Identity Select an identity to connect
DBHub.io Local Current Database
Name Last modified
SQL Log Plot DB Schema Remote

```
PRAGMA table_info(permissions);
```

	cid	name	type	notnull	dflt_value	pk
1	0	id	integer	1	NULL	1
2	1	name	varchar(191)	1	NULL	0
3	2	guard_name	varchar(191)	1	NULL	0
4	3	created_at	timestamp	0	NULL	0
5	4	updated_at	timestamp	0	NULL	0

Findings:

Table Name: permissions
id: An integer column which serves as a primary key, and uniquely identifies an each entry for permissions.It can not be set to null.

name: A string column allowing the character upto length of 191.This column cannot be set to null.

guard_name: A string column allowing the character upto length of 191.This field cannot be set to null.

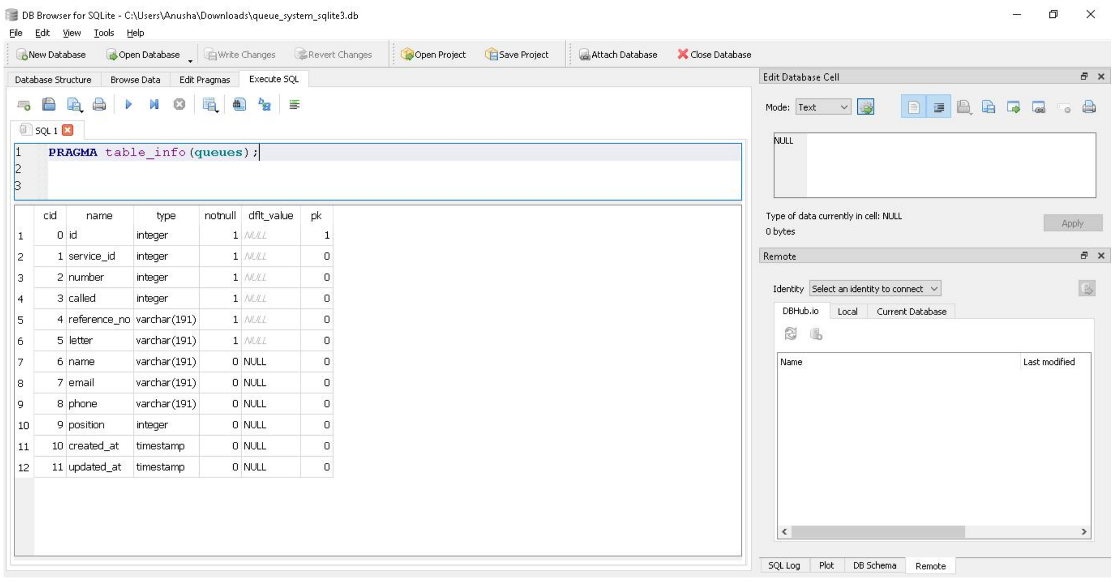
created_at: A timestamp column which records the timestamp when the permission entry was created.Since this column is optional and can be set to null.

updated_at: A timestamp column which records the timestamp when entry for permission was last updated at.This column can aswell be set to null.

Note: Schema itself doesn’t define the foreign key relationship, but role_has_permissions table might be refrencing to id column in permission table to point the role with specific permissions.

PRAGMA table_info(queues);

Result:



```
PRAGMA table_info(queues);
```

	cid	name	type	notnull	dflt_value	pk
1	0	id	integer	1	NULL	1
2	1	service_id	integer	1	NULL	0
3	2	number	integer	1	NULL	0
4	3	called	integer	1	NULL	0
5	4	reference_no	varchar(191)	1	NULL	0
6	5	letter	varchar(191)	1	NULL	0
7	6	name	varchar(191)	0	NULL	0
8	7	email	varchar(191)	0	NULL	0
9	8	phone	varchar(191)	0	NULL	0
10	9	position	integer	0	NULL	0
11	10	created_at	timestamp	0	NULL	0
12	11	updated_at	timestamp	0	NULL	0

DB Browser for SQLite - C:\Users\Anusha\Downloads\queue_system_sqlite3.db
File Edit View Tools Help
New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database
Database Structure Browse Data Edit Pragmas Execute SQL
SQL 1
Edit Database Cell
Mode: Text
NULL
Type of data currently in cell: NULL
0 bytes
Apply
Remote
Identity Select an identity to connect
DBHub.io Local Current Database
Name Last modified
SQL Log Plot DB Schema Remote

Findings:

Table Name: queues

id: An integer column serving as a primary key for the table and uniquely identifies each entry made to the queue.It cannot be set to null.

service_id: An integer column representing the id of the service associated with the queue.It can not be null aswell.

number: An integer column representing the number attached to each queue entry.This column can not be null.

callled: An integer column indicating if the queue number has been called or not.It cannot be set to null and uses a boolean value to identify(1 for called and 0 for not called).

refrence_no: A string column(varchar 191) which contain the reference number associated with the each queue entry.It cannot be null.

letter: A string column(varchcar 191) which represents a letter associated with each queue entry.It aswell cannot be null.

name: A string column(varchar 191) which stores the name associated with queue entry. It can be set to null.

email: A string column(varchar 191) which is used to store the email address associated with each queue entry.It can be null.

phone: A string column(varchar 191) which is used to store the phone number associated with each queue entry.It aswell can be null.

position: An integer column storing the records of position of entry in the queue.It aswell can be null.

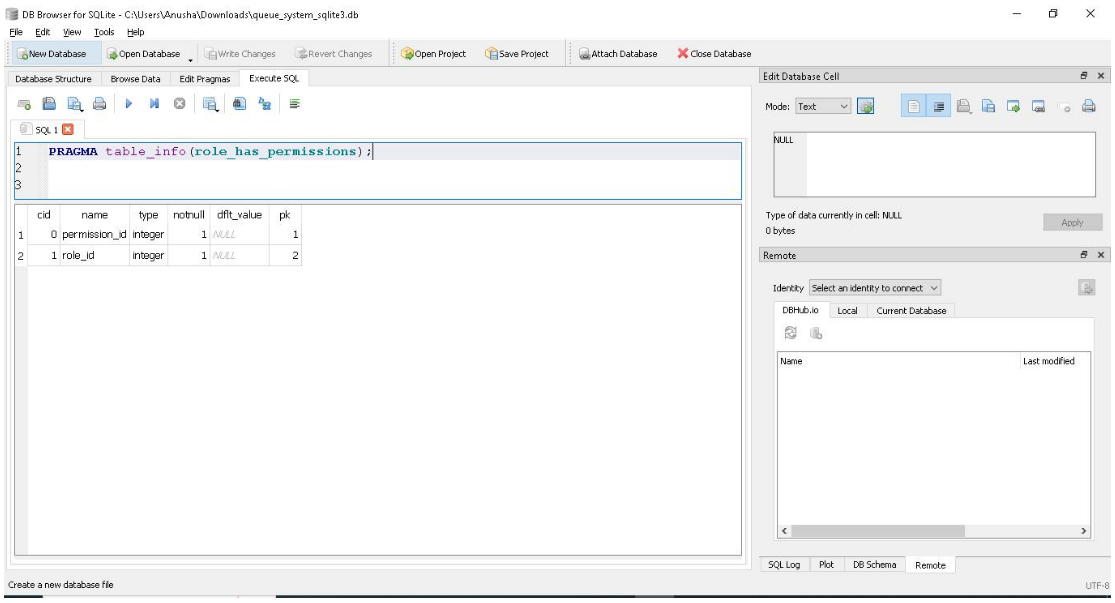
created_at: A timestamp column which records the timestamp when the queue was created at.It aswell can be null.

updated_at: A timestamp column which records the timestamp when the last record for queue entry was updated at.

Note: service_id might be serving as the foreign key linking to the id column in services table.

PRAGMA table_info(role_has_permissions);

Result:



```
PRAGMA table_info(role_has_permissions);
```

	cid	name	type	notnull	dflt_value	pk
1	0	permission_id	integer	1	NULL	1
2	1	role_id	integer	1	NULL	2

NULL

Type of data currently in cell: NULL
0 bytes

Findings:

Table Name: role_has_permissions

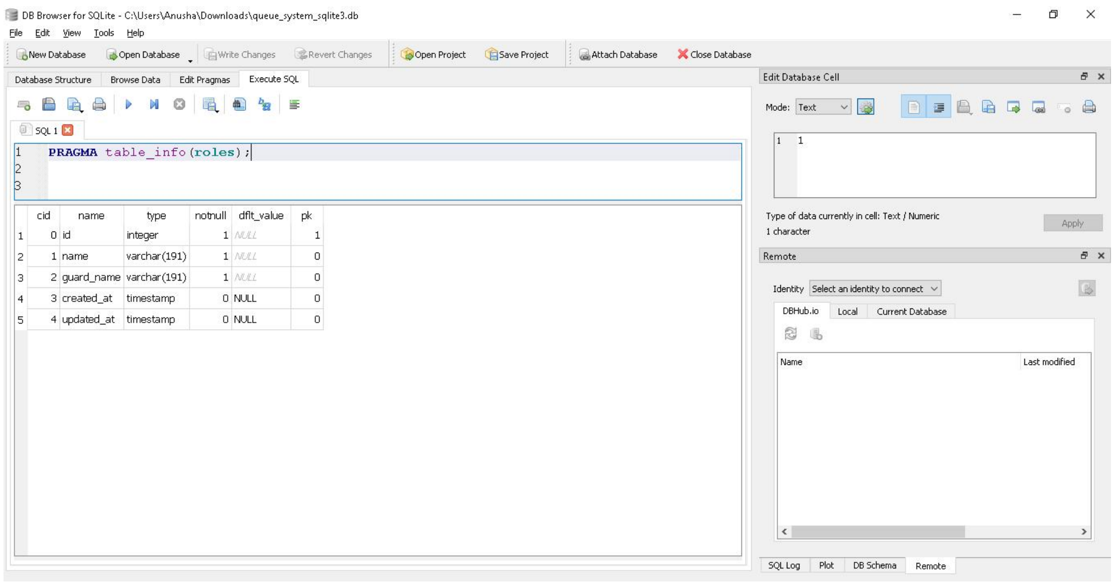
permission_id: An integer column which serves as a part of a composite primary key.It can not be null and serves as a foreign key linking to the permissions table.

role_id: An integer column which also serves as a part of composite primary key.It aswell serves as a foreign key linking to the roles table.It cannot be null.

Note: permission_id references to the id column in the table permissions and role_id references to the id column in the roles table.

PRAGMA table_info(roles);

Result:



DB Browser for SQLite - C:\Users\Anusha\Downloads\queue_system_sqlite3.db

```
PRAGMA table_info(roles);
```

	cid	name	type	notnull	dflt_value	pk
1	0	id	integer	1	NULL	1
2	1	name	varchar(191)	1	NULL	0
3	2	guard_name	varchar(191)	1	NULL	0
4	3	created_at	timestamp	0	NULL	0
5	4	updated_at	timestamp	0	NULL	0

Findings:

Table Name: roles

id: An integer column serving as a primary key and uniquely identifying each entry in the table roles.It cannot be null.

name: A string column(varchar 191) recording the name of the roles.It aswell cannot be null.

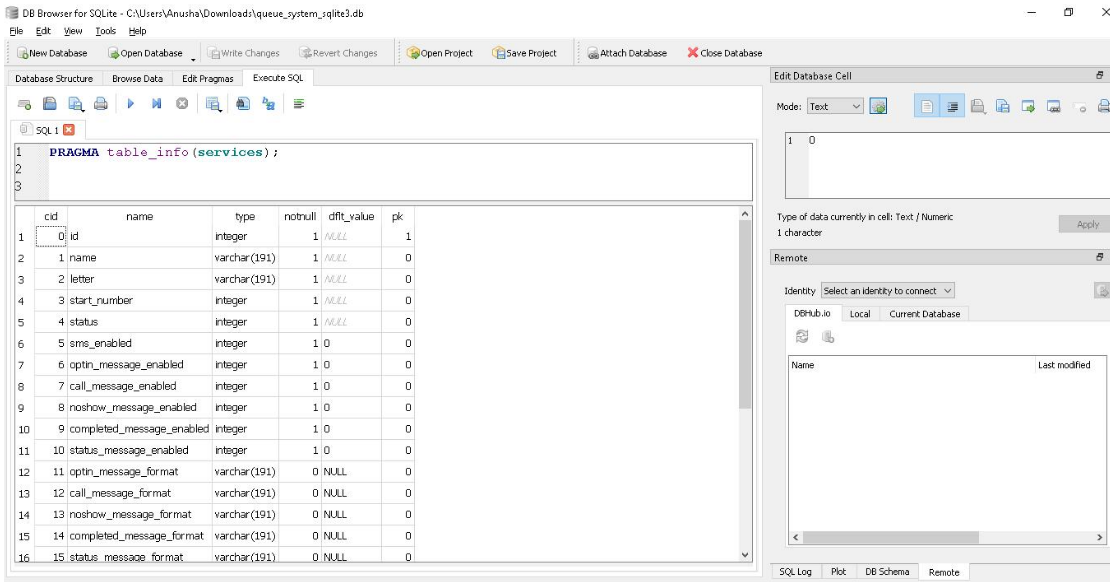
guard_name: A string column(varchar 191) which records the name of the guard defining the context for applicable role.It cannot be null.

created_at and updated_at: A timestamp column recording the timestamp when the entry for role was created at and when the entry for role was last updated at.

Note: Since the table roles doesnot specify the foreign key relationship but role_has_permissions column references the id column in table roles.

PRAGMA table_info(services);

Result:



DB Browser for SQLite - C:\Users\Anusha\Downloads\queue_system_sqlite3.db

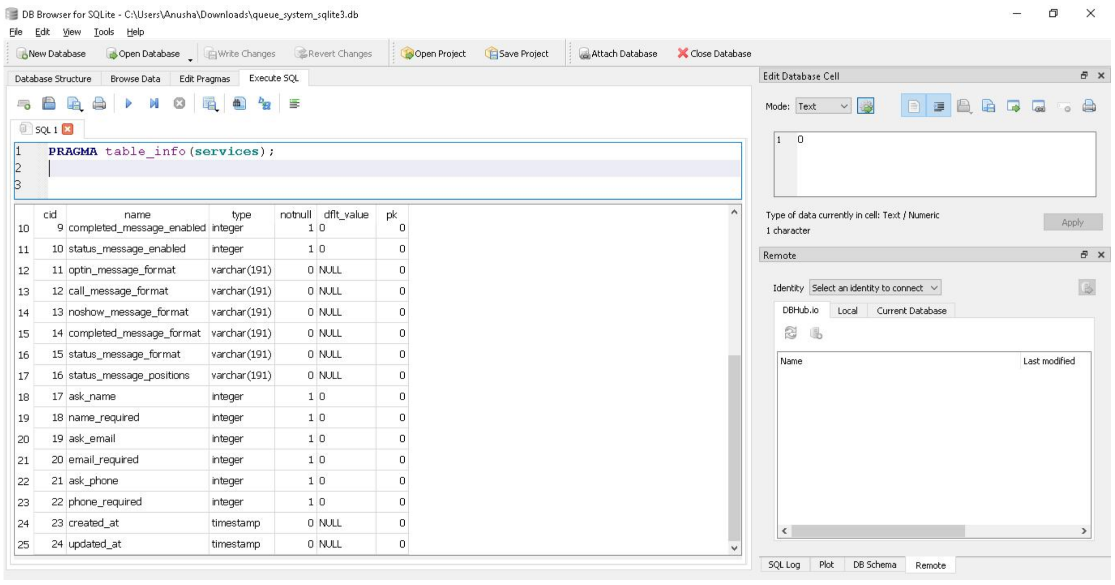
```
PRAGMA table_info(services);
```

	cid	name	type	notnull	dflt_value	pk
1	0	id	integer	1	NULL	1
2	1	name	varchar(191)	1	NULL	0
3	2	letter	varchar(191)	1	NULL	0
4	3	start_number	integer	1	NULL	0
5	4	status	integer	1	NULL	0
6	5	sms_enabled	integer	1	0	0
7	6	optin_message_enabled	integer	1	0	0
8	7	call_message_enabled	integer	1	0	0
9	8	noshow_message_enabled	integer	1	0	0
10	9	completed_message_enabled	integer	1	0	0
11	10	status_message_enabled	integer	1	0	0
12	11	optin_message_format	varchar(191)	0	NULL	0
13	12	call_message_format	varchar(191)	0	NULL	0
14	13	noshow_message_format	varchar(191)	0	NULL	0
15	14	completed_message_format	varchar(191)	0	NULL	0
16	15	status_message_format	varchar(191)	0	NULL	0

Edit Database Cell — Mode: Text

1 0

Type of data currently in cell: Text / Numeric
1 character



DB Browser for SQLite - C:\Users\Anusha\Downloads\queue_system_sqlite3.db

```
PRAGMA table_info(services);
```

	cid	name	type	notnull	dflt_value	pk
10	9	completed_message_enabled	integer	1	0	0
11	10	status_message_enabled	integer	1	0	0
12	11	optin_message_format	varchar(191)	0	NULL	0
13	12	call_message_format	varchar(191)	0	NULL	0
14	13	noshow_message_format	varchar(191)	0	NULL	0
15	14	completed_message_format	varchar(191)	0	NULL	0
16	15	status_message_format	varchar(191)	0	NULL	0
17	16	status_message_positions	varchar(191)	0	NULL	0
18	17	ask_name	integer	1	0	0
19	18	name_required	integer	1	0	0
20	19	ask_email	integer	1	0	0
21	20	email_required	integer	1	0	0
22	21	ask_phone	integer	1	0	0
23	22	phone_required	integer	1	0	0
24	23	created_at	timestamp	0	NULL	0
25	24	updated_at	timestamp	0	NULL	0

Edit Database Cell — Mode: Text

1 0

Type of data currently in cell: Text / Numeric
1 character

Findings:

Table Name: services

id: An integer column serving as a primary key.It cannot be null.

name: A string(varchar 191) column storing the records of the name of services. It cannot be null.

letter: A string(varcahr 191) column storing the records of identifier for the services.It aswell cannot be null.

start_number: An integer column recording the starting number of the services.It cannot be null.

status: An integer column recording the status of the service.It cannot be null.

sms_enabled: An integer column recording the values as 0 or 1 indicating if sms has been enabled or not.It cannot be null.

optin_message_enabled: An integer column recording the values as 0 or 1 indicating whether opt in message are enabled or not.It cannot be null.

call_message_enabled: An integer column recording the values as 0 or 1 indicating whether call message are enabled or not.It cannot be null.

noshow_message_enabled: An integer column recording the values as 0 or 1 indicating whether the no show message are enabled or not.It cannot be null.

completed_message_enabled: An integer column recording the values as 0 or 1 indicating whether the completed message are enabled or not.It cannot be null.

status_message_enabled: An integer column recording the values as 0 or 1 indicating whether the status messages are enabled or not.It cannot be null.

optin_message_format: A string(varchar 191) recording the format for opt in message.It can be null.

call_message_format: A string(varchar 191) recording the format for call message.It can be null.

noshow_message_format: A string(varchar 191) recording the format of the no show message.it can be null.

completed_message_format: A string(varchar 191) recording the format of the completed message.It can be null.

status_message_format: A string(varchar 191) recording the format of the status message.It aswell can be null.

status_message_positions: A string(varchar 191) recording the position of status messsage.It aswell can be optional.

ask_name: An integer column which records if the service ask for a customer name.It cannot be null.

name_required: An integer column which records values as (0 or 1) indicating if the providing the name is required during service or not. It cannot be null aswell.

ask_email: An integer column recording the values as 0 or 1 indicating if service had asked for customers email or not.It cannot be null.

email_required: An integer column recording the values as 0 or 1 indicating if email is required for the service or not.It cannot be null.

ask_phone: An integer column recording the values as 0 or 1 indicating if service asked for a customers phone or not.It cannot be null.

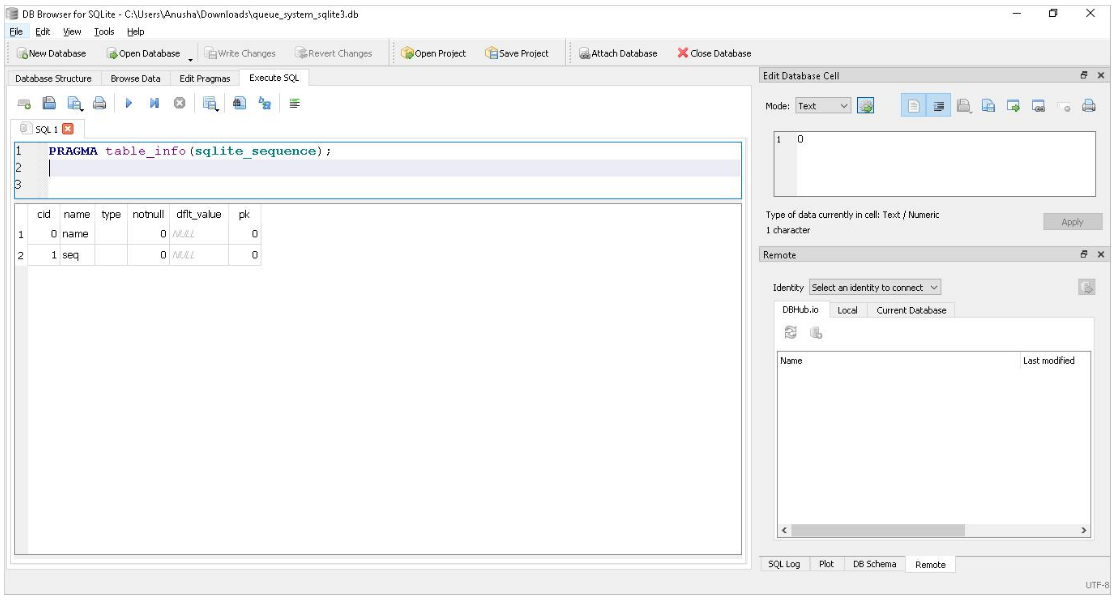
phone_required: An integer column recording the values as 0 or 1 indicating if phone number is required or not.It cannot be null.

created_at, updated_at: A timestamp column recording when the entry for service was first created and when the entry for service was last updated.It can be null.

Note: The service table can be linked to table queue, where service_id column make a reference to the service(id) to associate queue with services.

PRAGMA table_info(sqlite_sequence);

Result:


DB Browser for SQLite - C:\Users\Anusha\Downloads\queue_system_sqlite3.db

```
PRAGMA table_info(sqlite_sequence);
```

```
   cid  name  type  notnull  dflt_value  pk
1    0  name              0  NULL         0
2    1  seq               0  NULL         0
```

Type of data currently in cell: Text / Numeric
1 character

Findings:

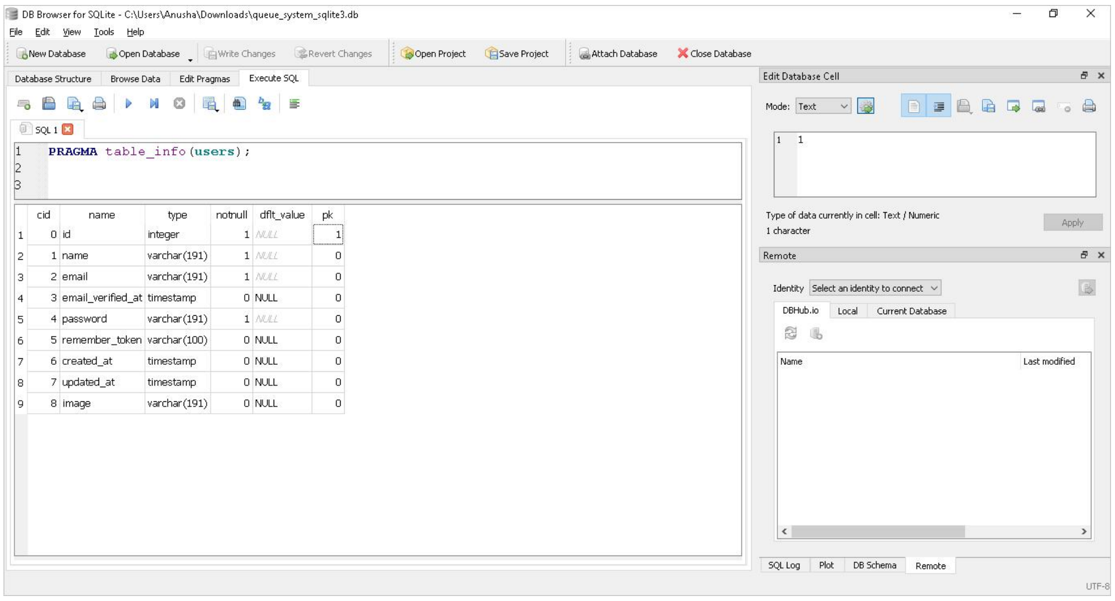
Table Name: sqlite_sequence

name: A string column which stores the name of the table that contains an auto increment field.

seq: An integer column which stores the current value of a autoincrement sequence for the specified table.

PRAGMA table_info(users);

Result:



DB Browser for SQLite - C:\Users\Anusha\Downloads\queue_system_sqlite3.db

```
PRAGMA table_info(users);
```

	cid	name	type	notnull	dflt_value	pk
1	0	id	integer	1	NULL	1
2	1	name	varchar(191)	1	NULL	0
3	2	email	varchar(191)	1	NULL	0
4	3	email_verified_at	timestamp	0	NULL	0
5	4	password	varchar(191)	1	NULL	0
6	5	remember_token	varchar(100)	0	NULL	0
7	6	created_at	timestamp	0	NULL	0
8	7	updated_at	timestamp	0	NULL	0
9	8	image	varchar(191)	0	NULL	0

Findings:
Table Name: users

id: Integer column serving as a primary key, uniquely identifying each users.It cannot be null.

name: String column (varchar 191) storing the name of each users.It cannot be null.

email: String column (varchar 191) storing the email of each user.It cannot be null.

email_verified_at: A timestamp column recording the timestamp when users email address was verified at.It can be null.

password: String column(varchar 191) storing the hashed password for user authentication.It cannot be null.

remember_token: String column (varchar 100) storing the token used to remember the session.It can be null.

created_at, verified_at: Timestamps storing the record of when user was first created at, and when user was last updated at.Both of it can be null.

Image: String(varchar 191) storing path or url of users profile image.