Object Oriented Programming (Using Python)

UNIT- V

Using Databases with Python:

- Using Databases
- Single Table CRUD (Create, Read, Update, and Delete)
- Designing and representing a data model
- Reconstructing data with JOIN
- Many-to-many relationships

Prof. R. MADANA MOHANA
Professor, Artificial Intelligence & Data Science
http://rmadanamohana.com/

References

https://www.coursera.org/learn/python-databases/home/info

https://www.py4e.com/book

http://do1.dr-chuck.com/pythonlearn/EN_us/pythonlearn.pdf

https://www.py4e.com/code3/

https://www.py4e.com/lectures3/

https://www.youtube.com/playlist?list=PLlRFEj9H3Oj7Bp8-DfGpfAfDBiblRfl5p

http://sqlitebrowser.org/

Designing a Data Model

Database Model

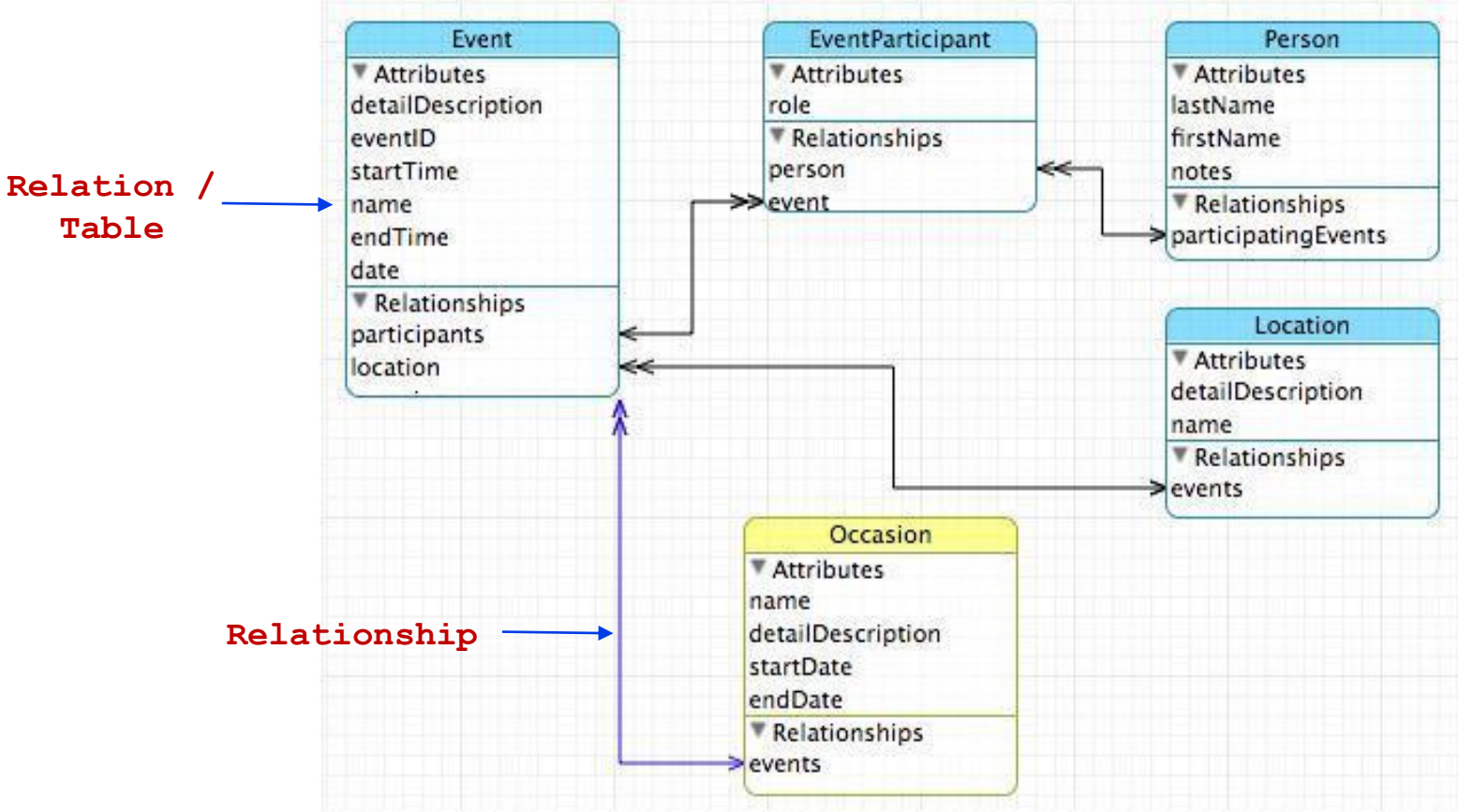
- A database model or database schema is the structure or format of a database, described in a formal language supported by the database management system.
- In other words, a "database model" is the application of a data model when used in conjunction with a database management system.

http://en.wikipedia.org/wiki/Database_model

Database Design

- Database design is an art form of its own with particular skills and experience
- Our goal is to avoid the really bad mistakes and design clean and easily understood databases
- Others may performance tune things later
- Database design starts with a picture...

Database Design



Building a Data Model

- Drawing a picture of the data objects for our application and then figuring out how to represent the objects and their relationships
- Basic Rule: Don’t put the same string data in twice - use a relationship instead
- When there is one thing in the “real world” there should be one copy of that thing in the database

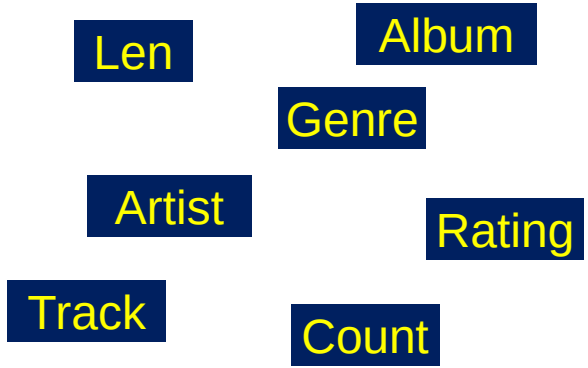
Building a Data Model - Example

Track	Len	Artist	Album	Genre	Rating	Count
Hells Bells	5:13	AC/DC	Who Made Who	Rock	★★★★★	61
Shake Your Foundations	3:54	AC/DC	Who Made Who	Rock	★★★★★	70
Chase the Ace	3:01	AC/DC	Who Made Who	Rock		56
For Those About To Rock (We ...	5:54	AC/DC	Who Made Who	Rock	★★★★★	61
Dúlamán	3:43	Altan	Natural Wonders M...	New Age		31
Rode Across the Desert	4:10	America	Greatest Hits	Easy Listen...	★★★★★	23
Now You Are Gone	3:08	America	Greatest Hits	Easy Listen...	★★★★★	18
Tin Man	3:30	America	Greatest Hits	Easy Listen...	★★★★★	23
Sister Golden Hair	3:22	America	Greatest Hits	Easy Listen...	★★★★★	24
Track 01	4:22	Billy Price	Danger Zone	Blues/R&B	★★★★★	26
Track 02	2:45	Billy Price	Danger Zone	Blues/R&B	★★★★★	18
Track 03	3:26	Billy Price	Danger Zone	Blues/R&B	★★★★★	22
Track 04	4:17	Billy Price	Danger Zone	Blues/R&B	★★★★★	18
Track 05	3:50	Billy Price	Danger Zone	Blues/R&B	★★★★★	21
War Pigs/Luke's Wall	7:58	Black Sabbath	Paranoid	Metal	★★★★★	25
Paranoid	2:53	Black Sabbath	Paranoid	Metal	★★★★★	22
Planet Caravan	4:35	Black Sabbath	Paranoid	Metal	★★★★★	25
Iron Man	5:59	Black Sabbath	Paranoid	Metal	★★★★★	26
Electric Funeral	4:53	Black Sabbath	Paranoid	Metal	★★★★★	22
Hand of Doom	7:10	Black Sabbath	Paranoid	Metal	★★★★★	23
Rat Salad	2:30	Black Sabbath	Paranoid	Metal	★★★★★	31
Jack the Stripper/Fairies Wear ...	6:14	Black Sabbath	Paranoid	Metal	★★★★★	24
Bomb Squad (TECH)	3:28	Brent	Brent's Album			1
clay techno	4:36	Brent	Brent's Album			2
Heavy	3:08	Brent	Brent's Album			1
Hi metal man	4:20	Brent	Brent's Album			1
Mistro	2:58	Brent	Brent's Album			1

Building a Data Model - Example

For each "piece of info"…

- Is the column an object or an attribute of another object?
- Once we define objects, we need to define the relationships between objects



Hells Bells	5:13	AC/DC	Who Made Who	Rock	★★★★★	61
Shake Your Foundations	3:54	AC/DC	Who Made Who	Rock	★★★★★	70
Chase the Ace	3:01	AC/DC	Who Made Who	Rock		56
For Those About To Rock (We …	5:54	AC/DC	Who Made Who	Rock	★★★★★	61
Dúlamán	3:43	Altan	Natural Wonders M…	New Age		31
Rode Across the Desert	4:10	America	Greatest Hits	Easy Listen…	★★★★★	23
Now You Are Gone	3:08	America	Greatest Hits	Easy Listen…	★★★★★	18

Building a Data Model - Example

Track

Album

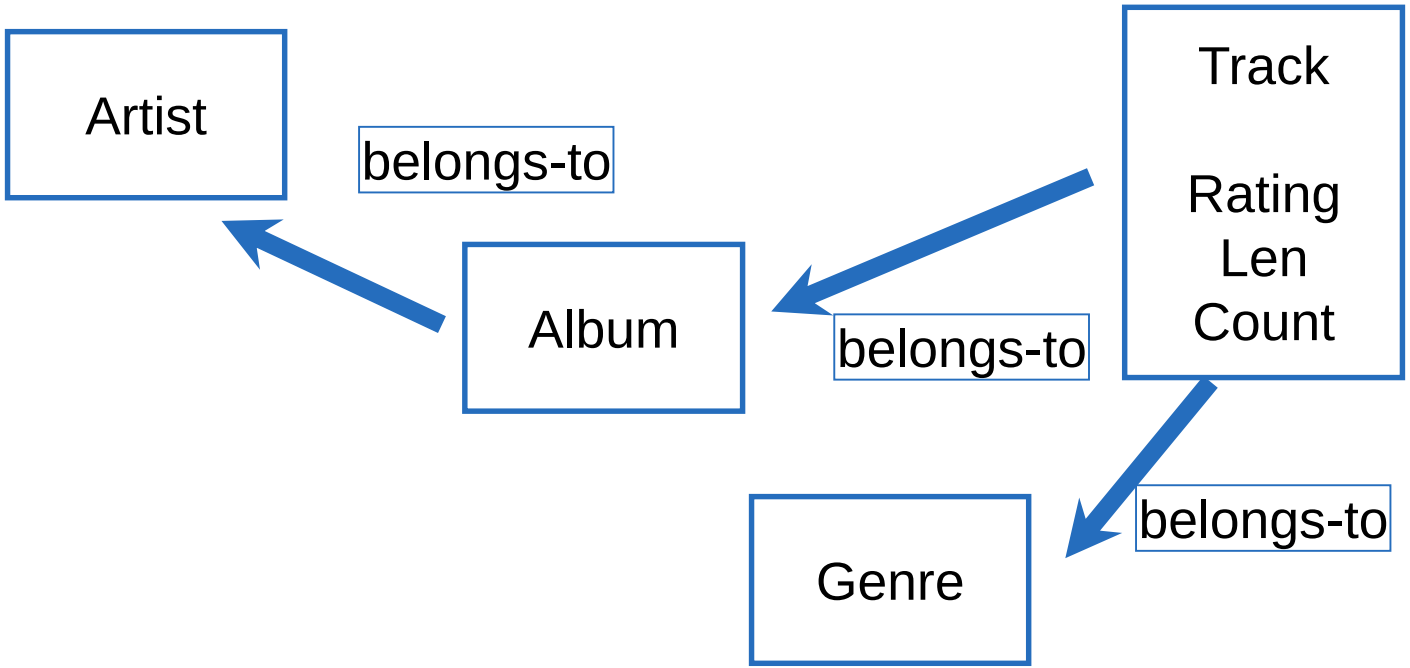
Artist

Genre

Rating

Len

Count



Hells Bells	5:13	AC/DC	Who Made Who	Rock	★★★★★	61
Shake Your Foundations	3:54	AC/DC	Who Made Who	Rock	★★★★★	70
Chase the Ace	3:01	AC/DC	Who Made Who	Rock		56
For Those About To Rock (We ...	5:54	AC/DC	Who Made Who	Rock	★★★★★	61
Dúlamán	3:43	Altan	Natural Wonders M...	New Age		31
Rode Across the Desert	4:10	America	Greatest Hits	Easy Listen...	★★★★★	23
Now You Are Gone	3:08	America	Greatest Hits	Easy Listen...	★★★★★	18

Representing Relationships in Database

Database Normalization (3NF)

- There is *tons* of database theory - way too much to understand without excessive predicate calculus
- Do not replicate data - reference data - point at data
- Use integers for keys and for references
- Add a special "key" column to each table which we will make references to. By convention, many programmers call this column "id"

http://en.wikipedia.org/wiki/Database_normalization

Database Normalization (3NF)

Hells Bells	5:13	AC/DC	Who Made Who	Rock	★★★★★	61
Shake Your Foundations	3:54	AC/DC	Who Made Who	Rock	★★★★★	70
Chase the Ace	3:01	AC/DC	Who Made Who	Rock		56
For Those About To Rock (We ...	5:54	AC/DC	Who Made Who	[illegible]	★★★★★	61
Dúlamán	3:43	Altan	Natural Wonders M...	New [illegible]ge		31
Rode Across the Desert	[illegible]	America	Greatest Hits	Easy Liste...	★★★★★	23
Now You Are Gone	3:08	America	Greatest Hits	Easy Listen...	★★★★★	18
Tin Man	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

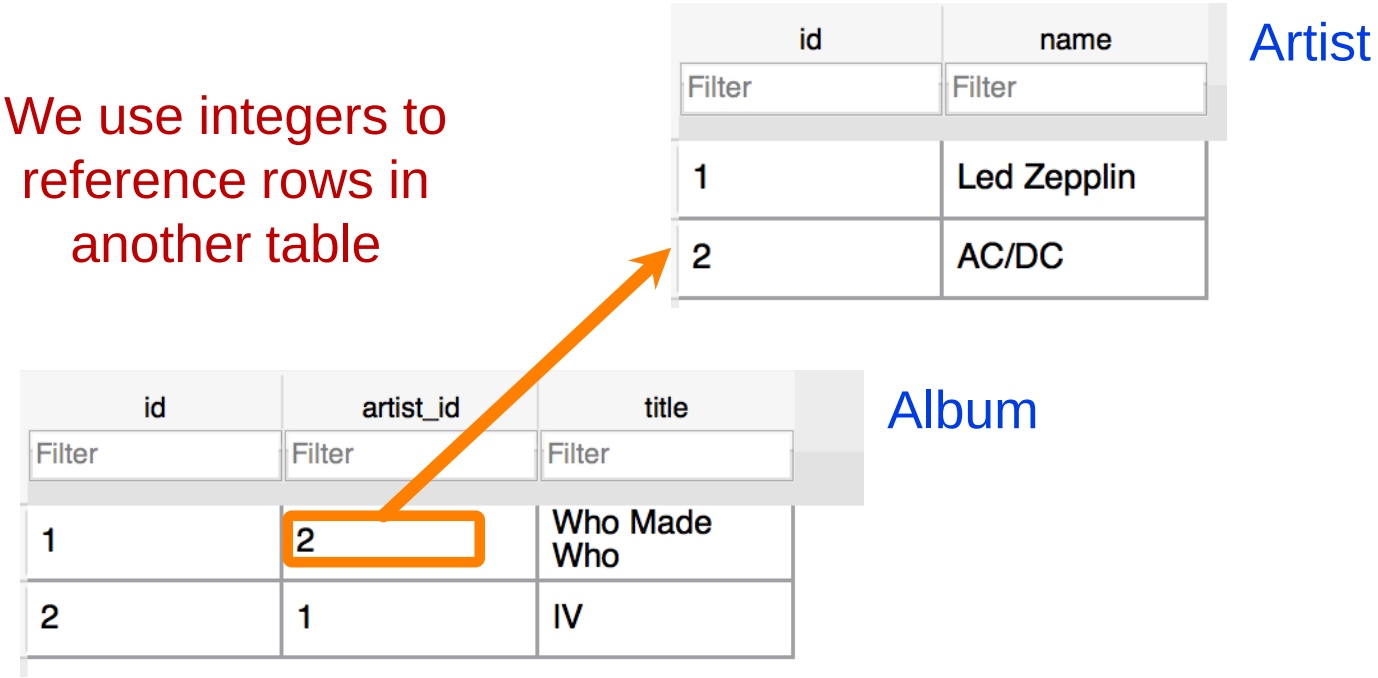
We want to keep track of which band is the "creator" of each music track...

What album does this song "belong to"??

Which album is this song related to?

Integer Reference Pattern

We use integers to reference rows in another table



Artist

id	name
Filter	Filter
1	Led Zepplin
2	AC/DC

Album

id	artist_id	title
Filter	Filter	Filter
1	2	Who Made Who
2	1	IV

Three Kinds of Keys

Primary key - generally an integer auto-increment field

Logical key - What the outside world uses for lookup

Foreign key - generally an integer key pointing to a row in another table

Key Rules

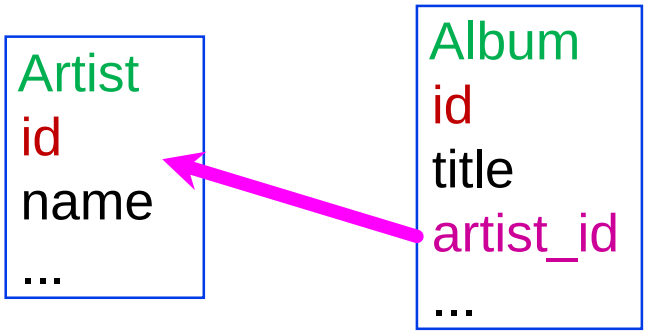
Best practices

- Never use your logical key as the primary key
- Logical keys can and do change, although slowly
- Relationships that are based on matching string fields are less efficient than integers

User
id
login
password
name
email
created_at
modified_at
login_at

Foreign Keys

- A foreign key is when a table has a column that contains a key which points to the primary key of another table.
- When all primary keys are integers, then all foreign keys are integers - this is good - very good



Representing a Data Model / Relationship Building (in Tables)

Track

Album

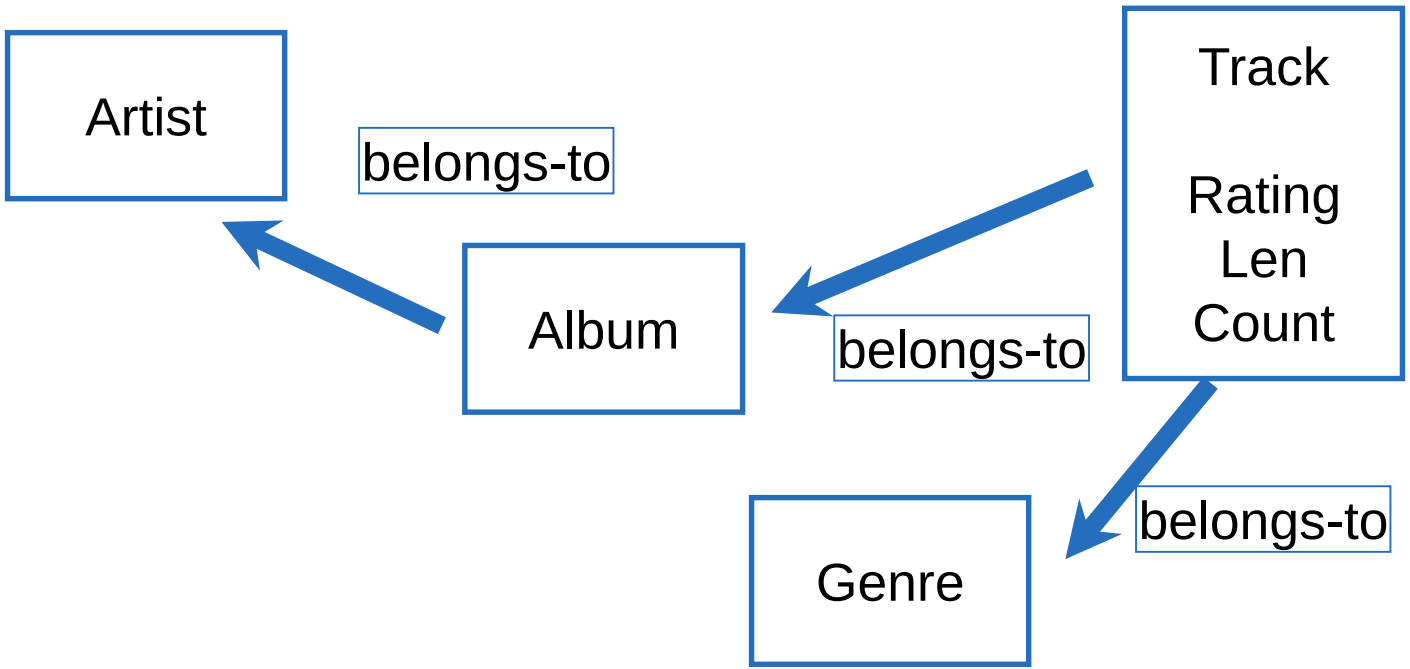
Artist

Genre

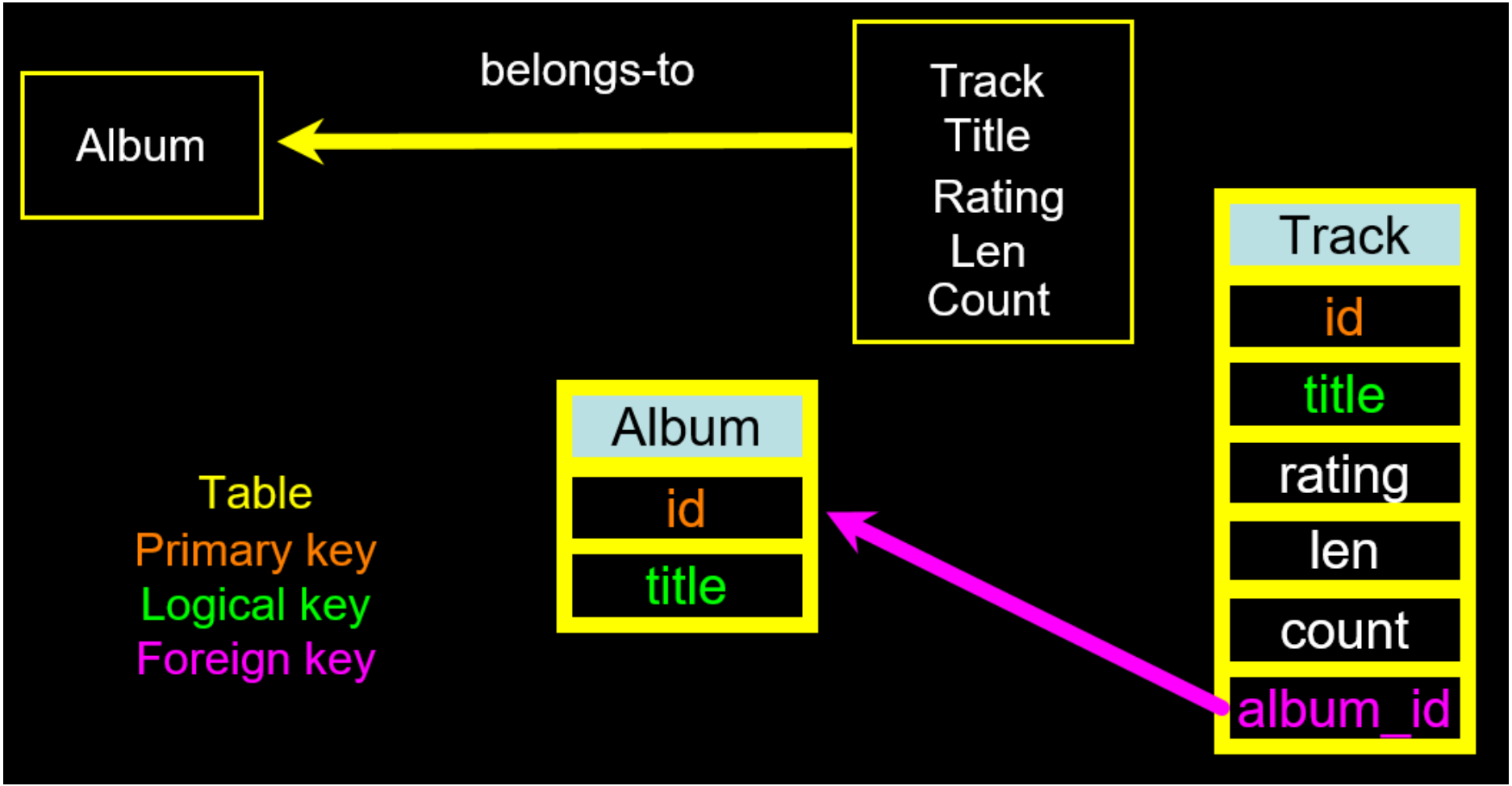
Rating

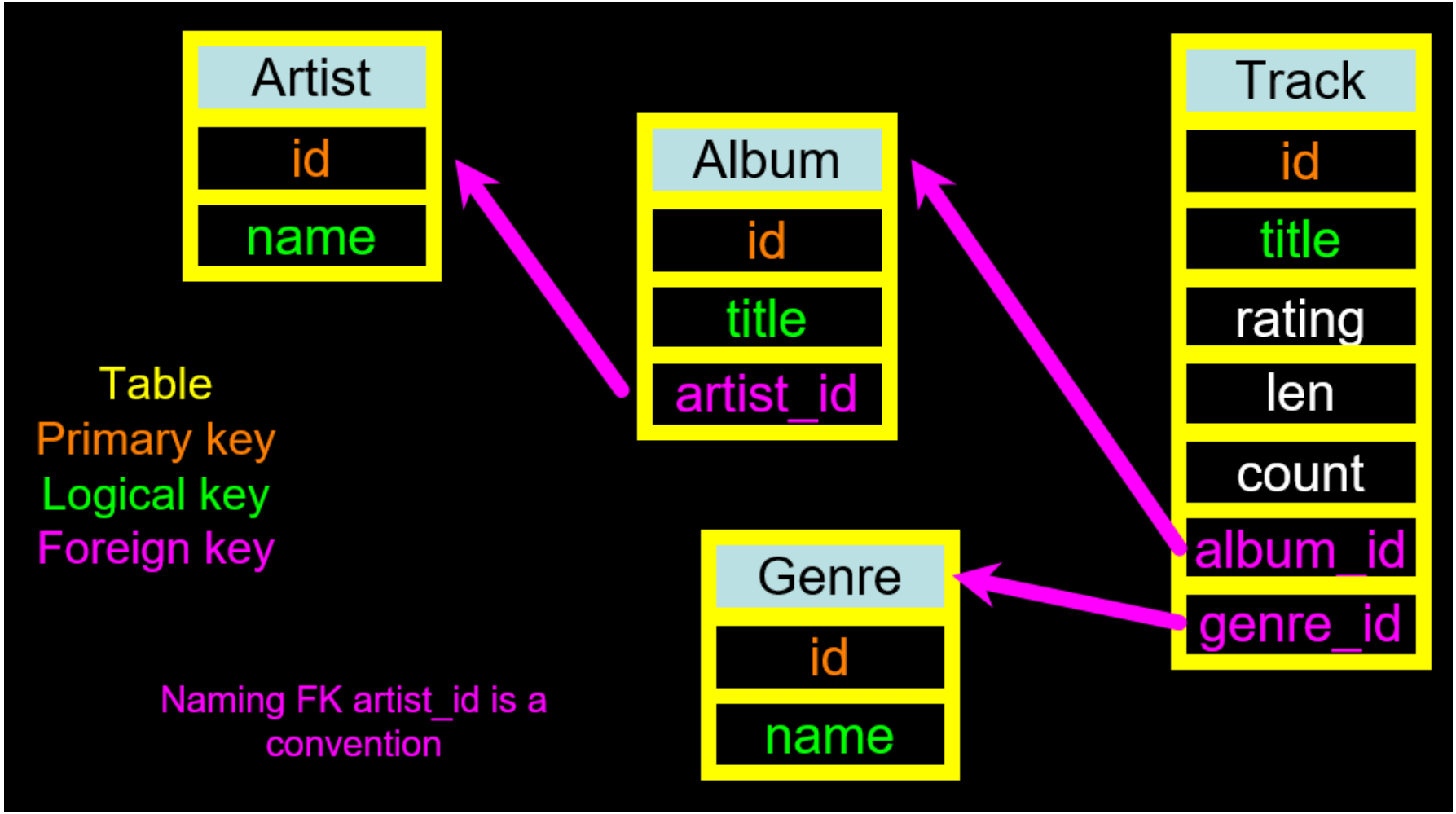
Len

Count



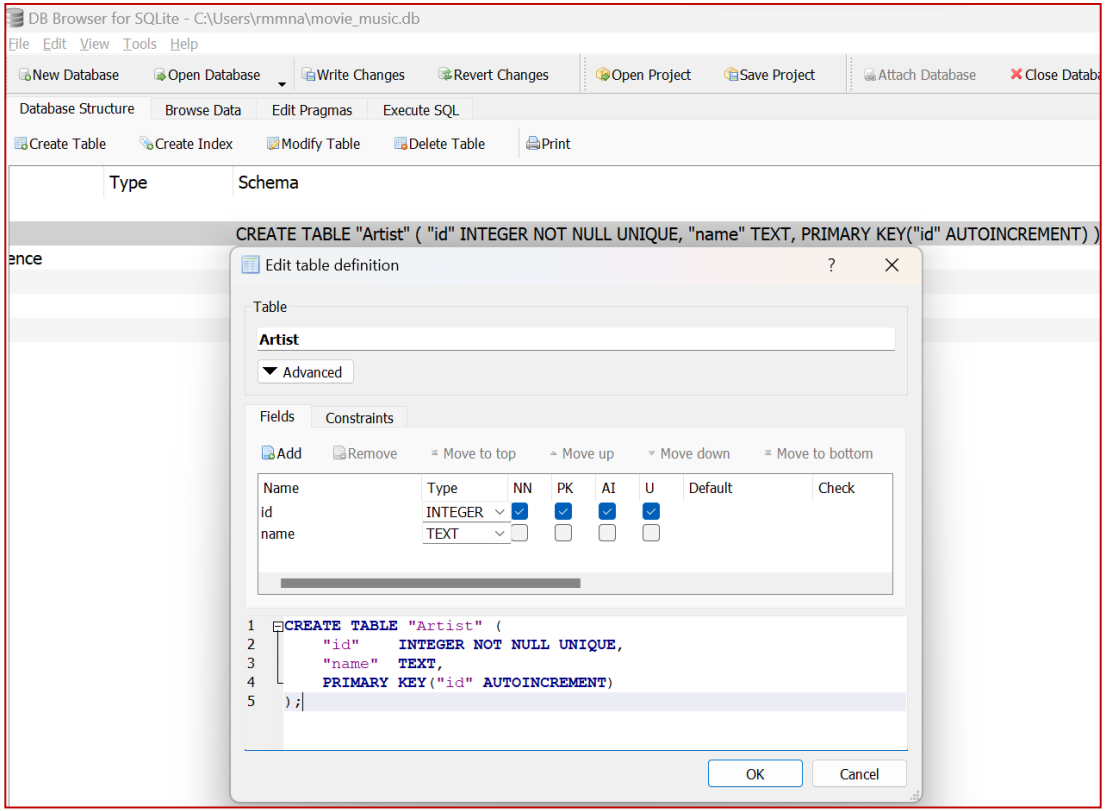
Hells Bells	5:13	AC/DC	Who Made Who	Rock	★★★★★	61
Shake Your Foundations	3:54	AC/DC	Who Made Who	Rock	★★★★★	70
Chase the Ace	3:01	AC/DC	Who Made Who	Rock		56
For Those About To Rock (We ...	5:54	AC/DC	Who Made Who	Rock	★★★★★	61
Dúlamán	3:43	Altan	Natural Wonders M...	New Age		31
Rode Across the Desert	4:10	America	Greatest Hits	Easy Listen...	★★★★★	23
Now You Are Gone	3:08	America	Greatest Hits	Easy Listen...	★★★★★	18


Album
belongs-to
Track
Title
Rating
Len
Count
Track
id
title
rating
len
count
album_id
Album
id
title
Table
Primary key
Logical key
Foreign key


Artist
id
name
Album
id
title
artist_id
Track
id
title
rating
len
count
album_id
genre_id
Genre
id
name
Table
Primary key
Logical key
Foreign key
Naming FK artist_id is a convention

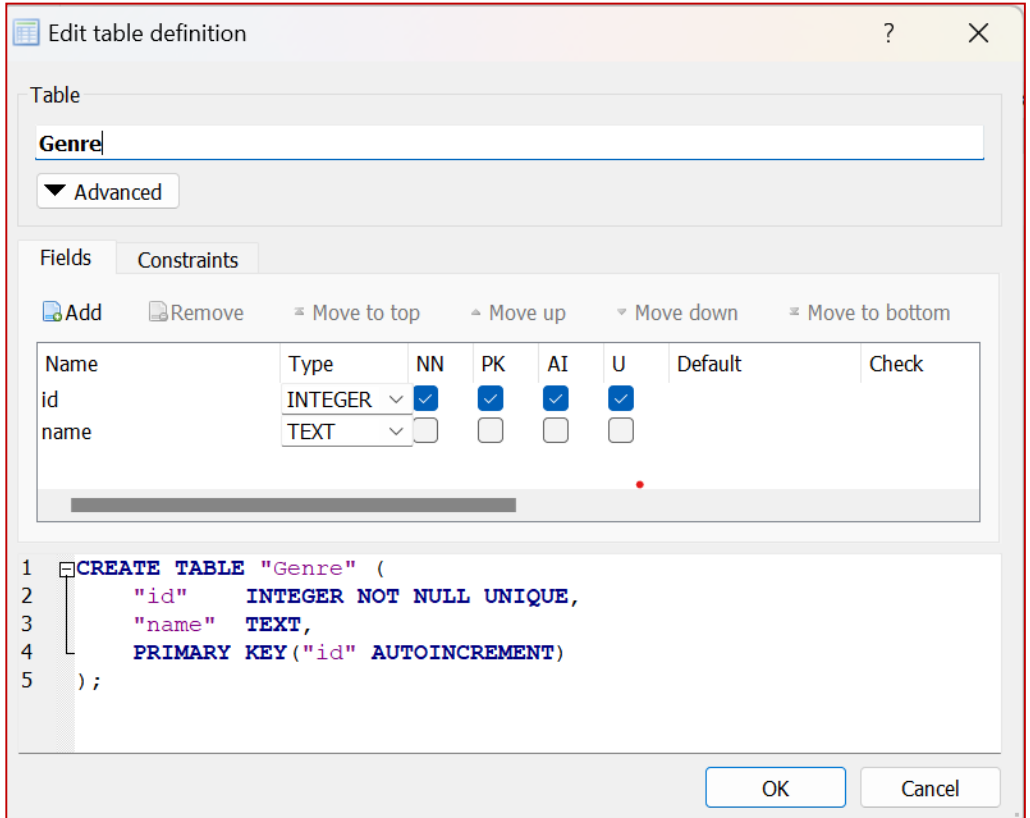
Creating Tables using SQLite

```
CREATE TABLE "Artist" ("id" INTEGER NOT NULL UNIQUE,
        "name" TEXT, PRIMARY KEY("id" AUTOINCREMENT));
```



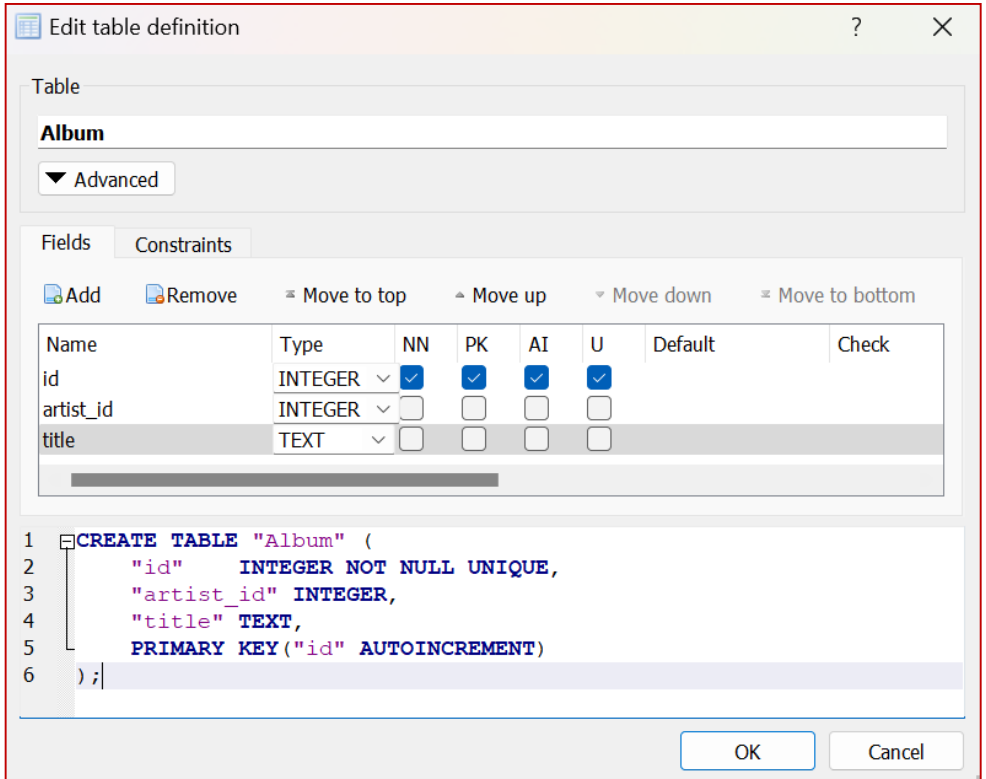
Creating Tables using SQLite

```
CREATE TABLE "Genre" ("id" INTEGER NOT NULL UNIQUE,
        "name" TEXT, PRIMARY KEY("id" AUTOINCREMENT));
```



Creating Tables using SQLite

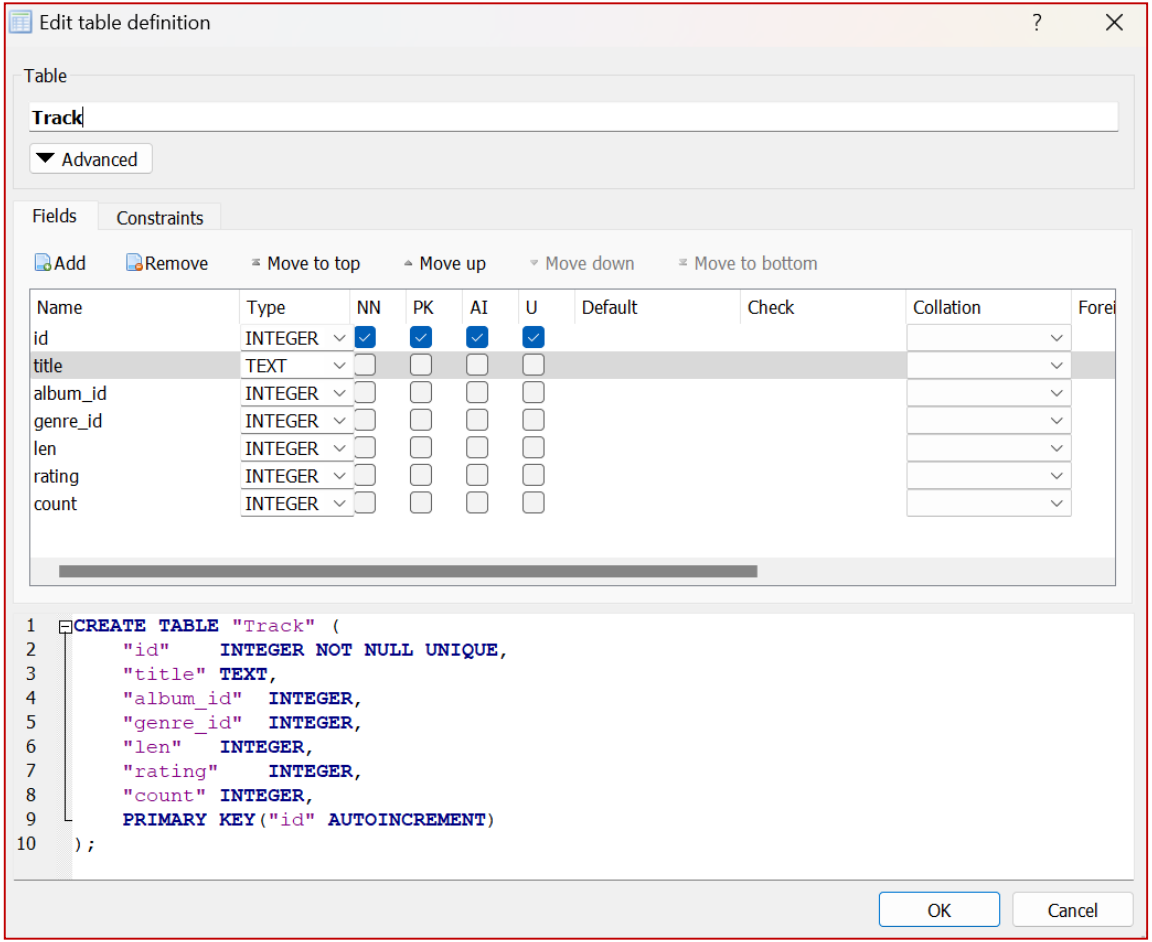
```
CREATE TABLE "Album" ("id" INTEGER NOT NULL UNIQUE,
      "artist_id" INTEGER, "title"  TEXT,PRIMARY KEY("id"
AUTOINCREMENT));
```



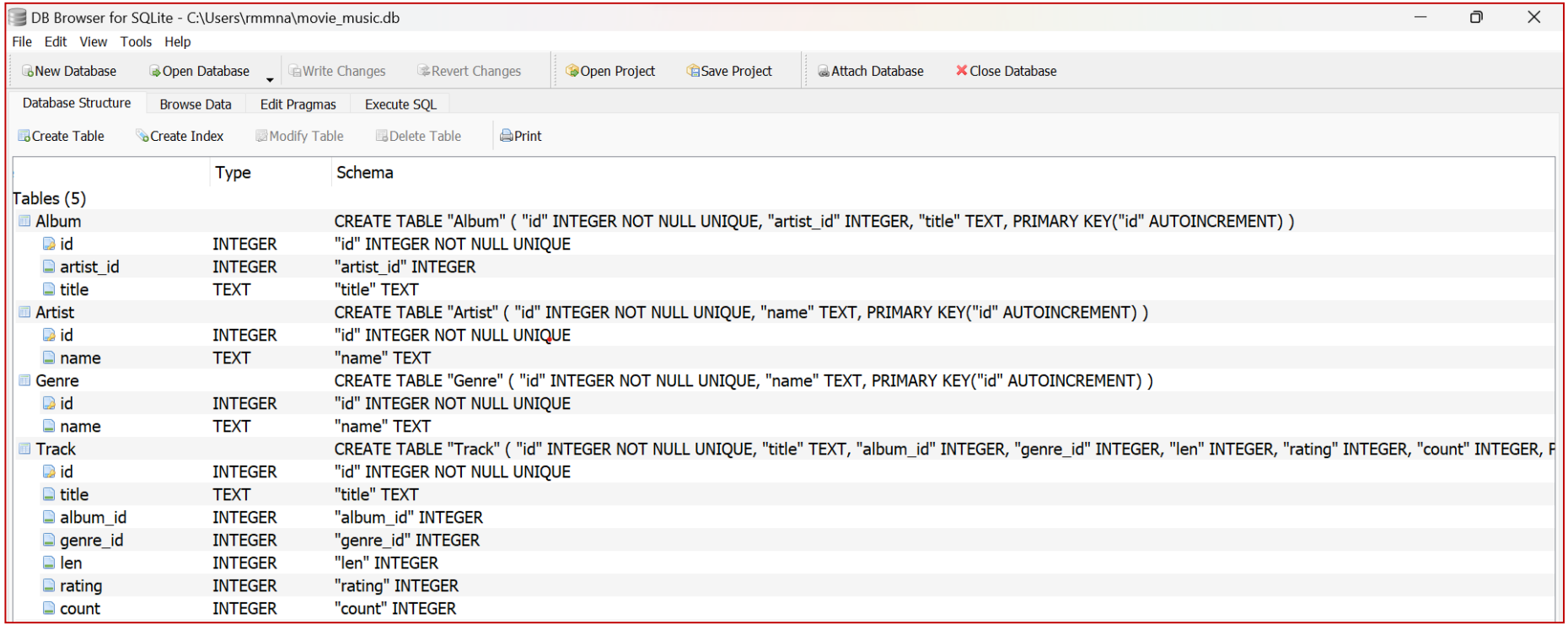
Creating Tables using SQLite

```
CREATE TABLE "Track" (
      "id"  INTEGER NOT NULL UNIQUE,
      "title"     TEXT,
      "album_id"  INTEGER,
      "genre_id"  INTEGER,
      "len" INTEGER,
      "rating"    INTEGER,
      "count"     INTEGER,
      PRIMARY KEY("id" AUTOINCREMENT)
);
```

Creating Tables using SQLite

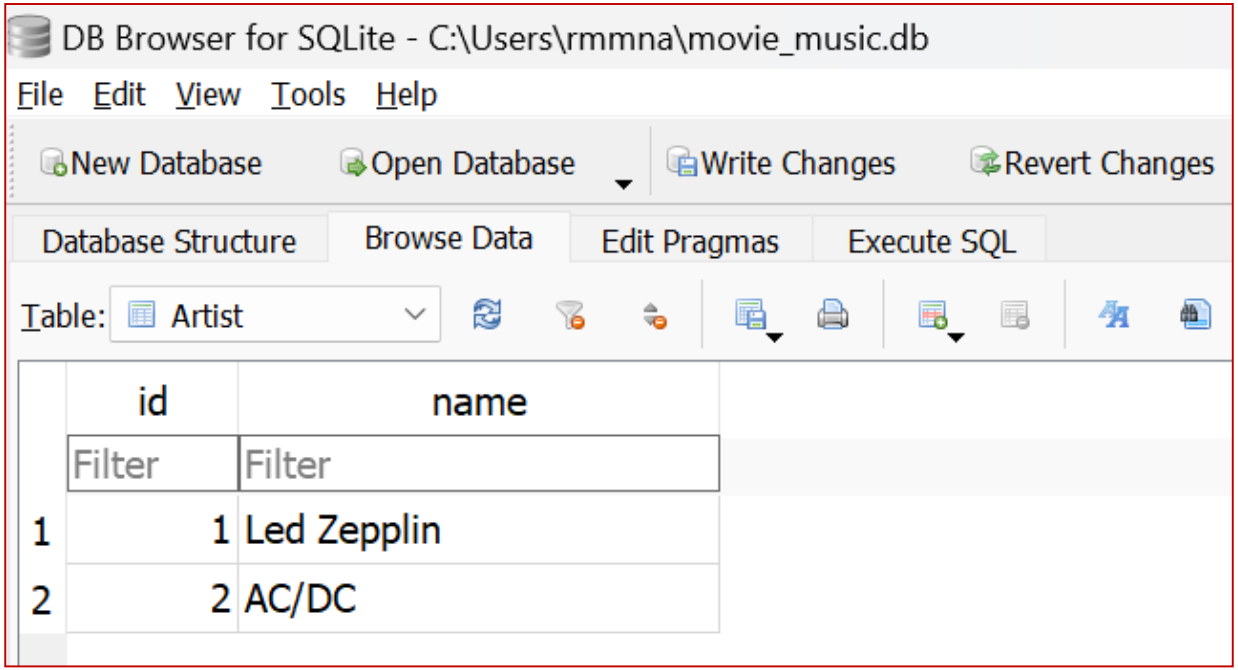


Inserting Relational Data



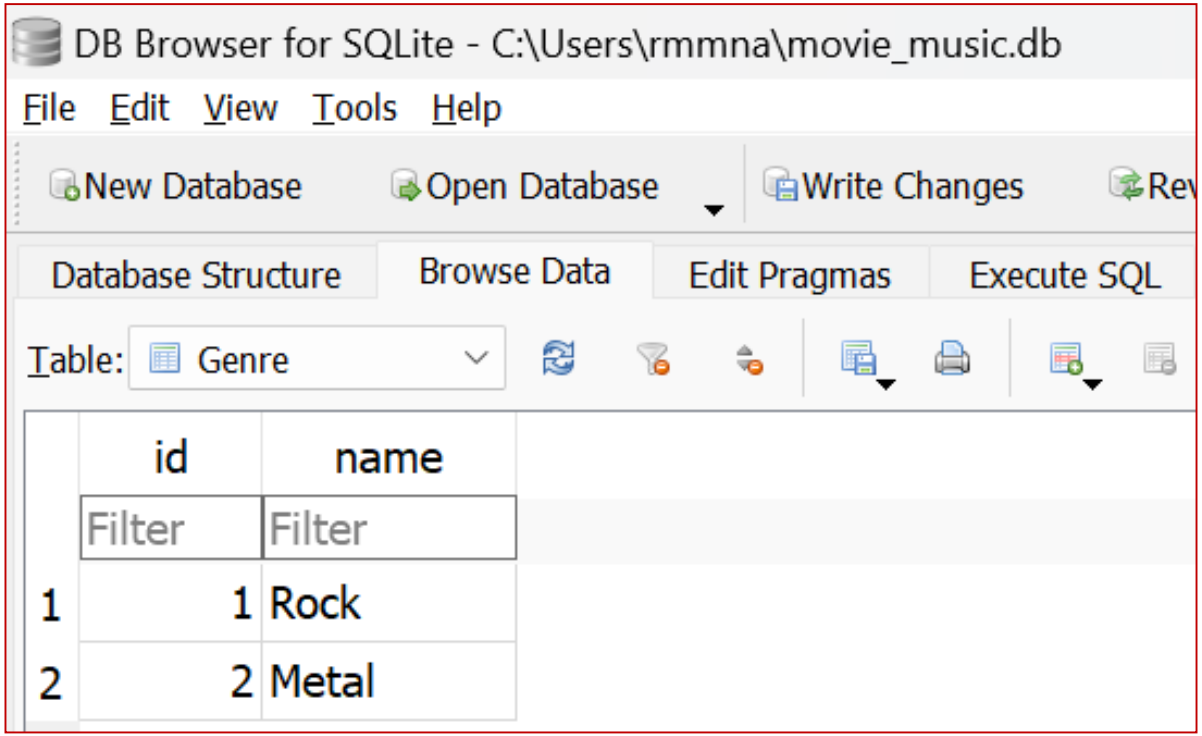
Inserting Relational Data

```
insert into Artist (name) values ('Led Zepplin')
insert into Artist (name) values ('AC/DC')
```



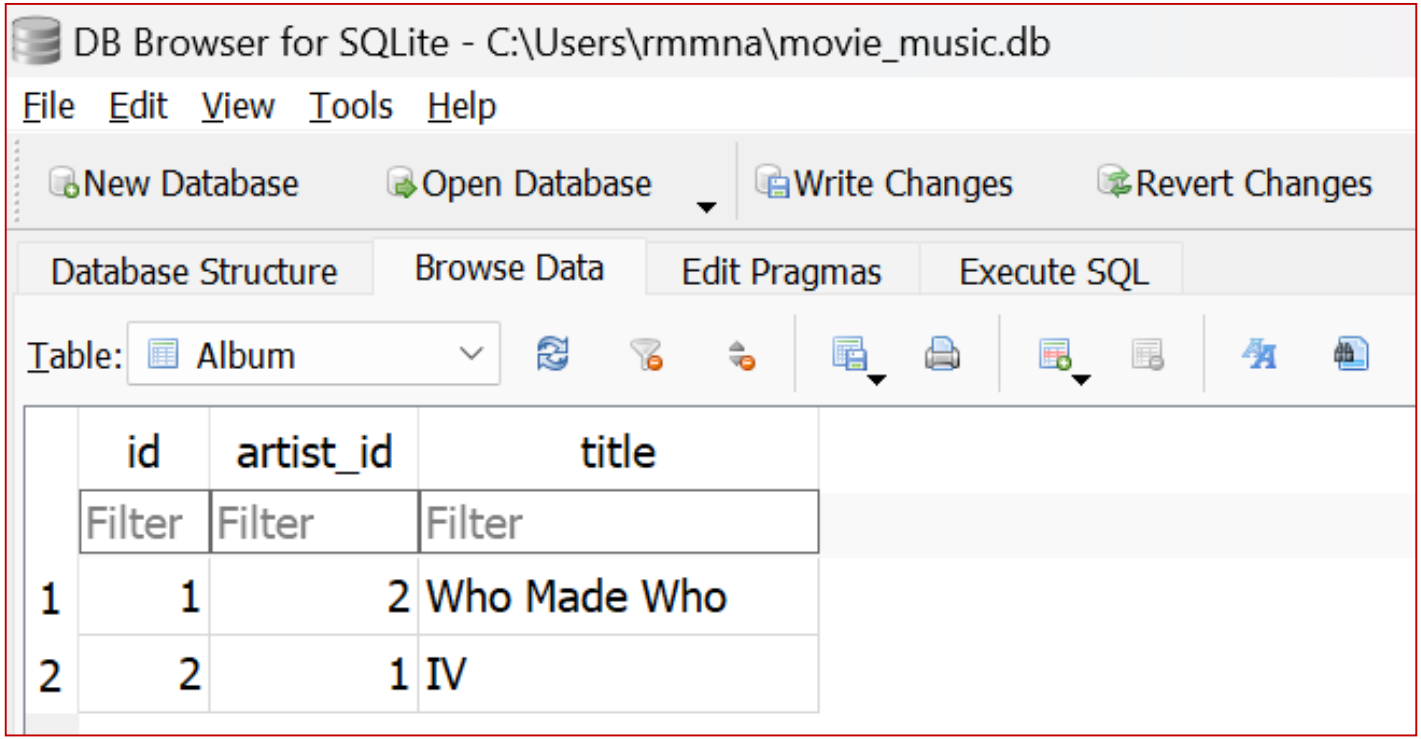
Inserting Relational Data

```
insert into Genre (name) values ('Rock')
insert into Genre (name) values ('Metal')
```



Inserting Relational Data

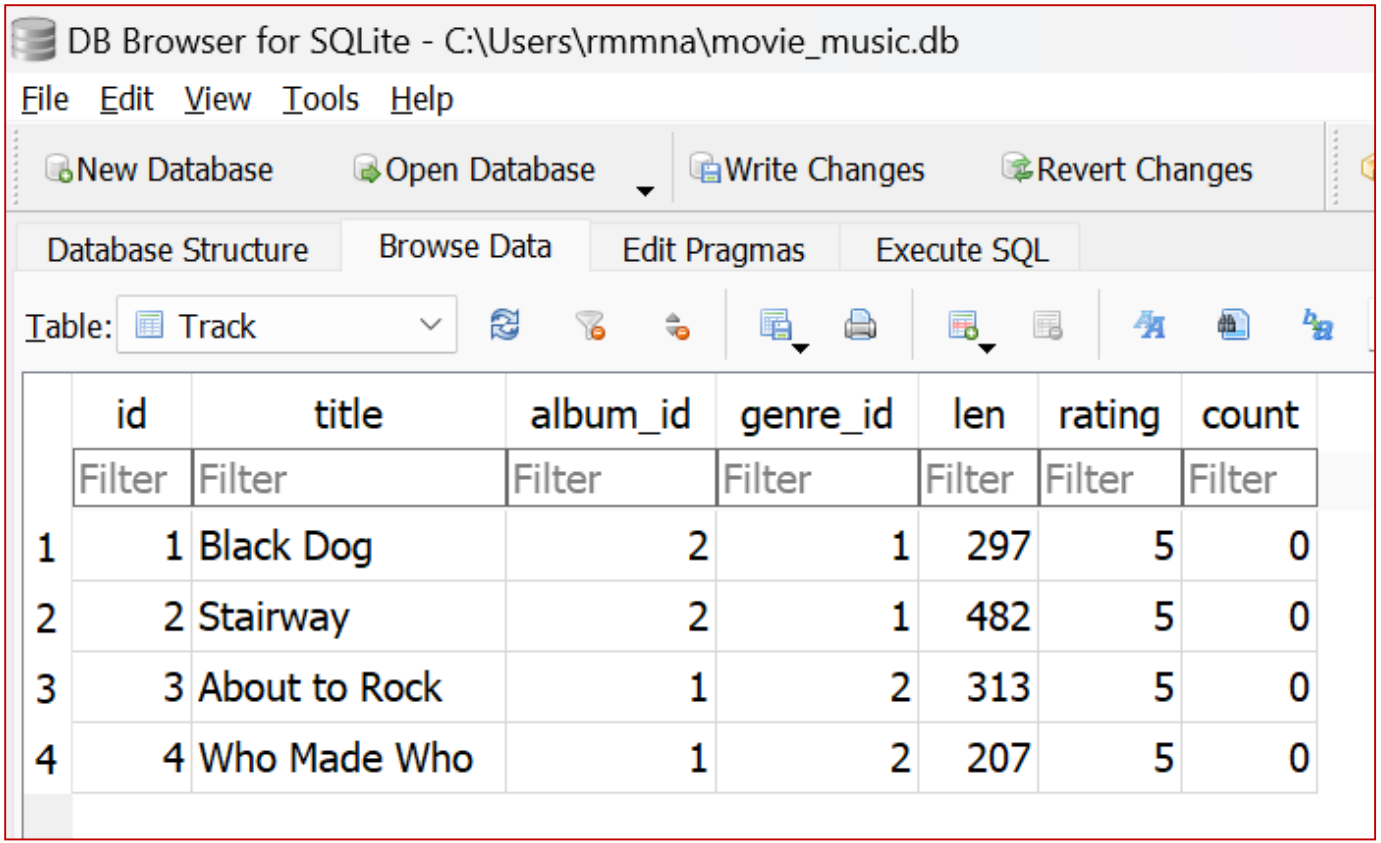
```
insert into Album (title, artist_id) values ('Who Made Who', 2)
insert into Album (title, artist_id) values ('IV', 1)
```



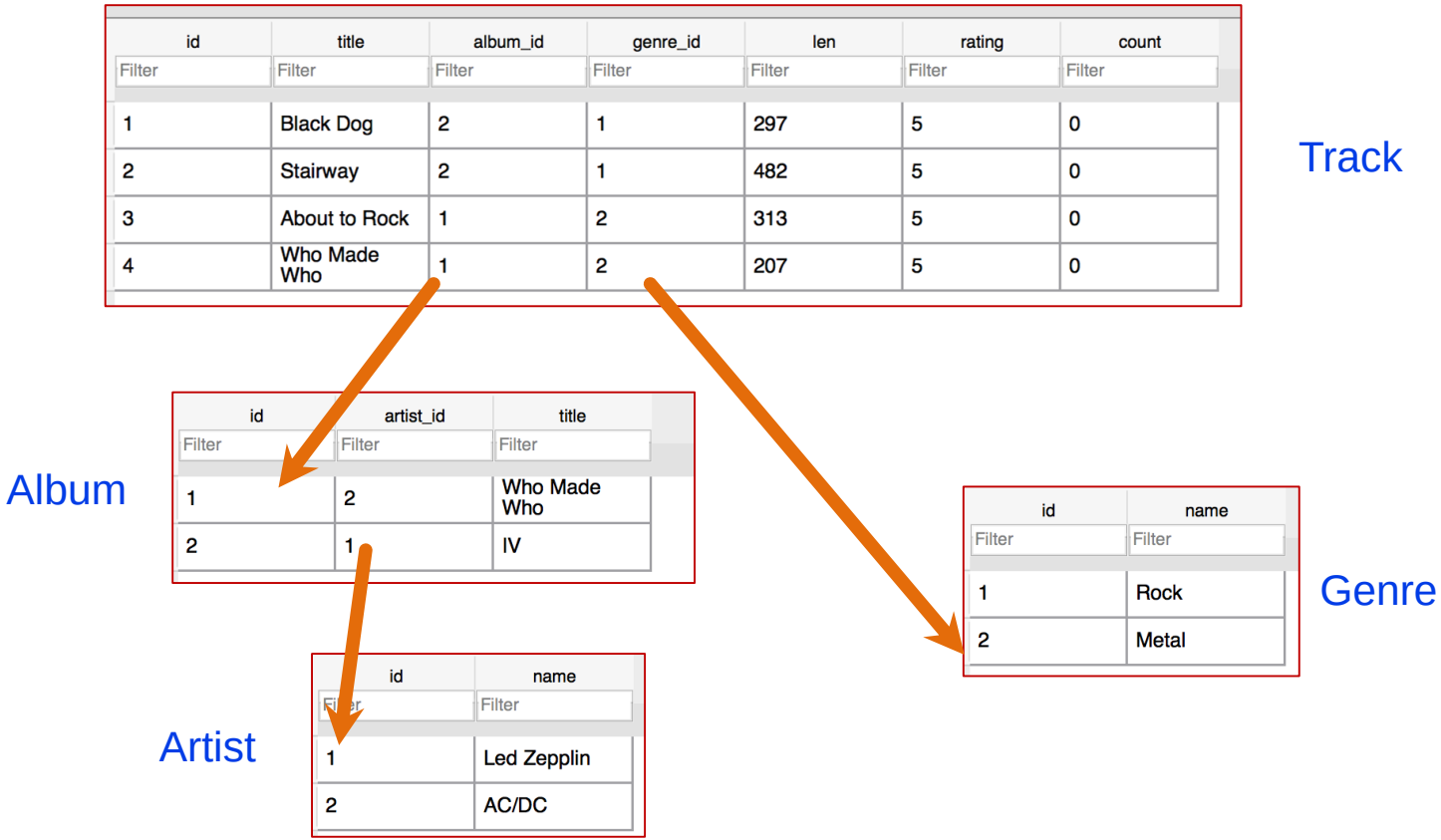
Inserting Relational Data

```
insert into Track (title, rating, len, count, album_id,
genre_id) values ('Black Dog', 5, 297, 0, 2, 1)

insert into Track (title, rating, len, count, album_id,
genre_id) values ('Stairway', 5, 482, 0, 2, 1)

insert into Track (title, rating, len, count, album_id,
genre_id) values ('About to Rock', 5, 313, 0, 1, 2)

insert into Track (title, rating, len, count, album_id,
genre_id) values (' Who Made Who ', 5, 313, 0, 1, 2)
```

Inserting Relational Data



	id	title	album_id	genre_id	len	rating	count
	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	1	Black Dog	2	1	297	5	0
2	2	Stairway	2	1	482	5	0
3	3	About to Rock	1	2	313	5	0
4	4	Who Made Who	1	2	207	5	0


id
title
album_id
genre_id
len
rating
count
Filter
1
Black Dog
2
1
297
5
0
2
Stairway
2
1
482
5
0
3
About to Rock
1
2
313
5
0
4
Who Made Who
1
2
207
5
0
Track
Album
id
artist_id
title
Filter
1
2
Who Made Who
2
1
IV
Artist
id
name
Filter
1
Led Zepplin
2
AC/DC
Genre
id
name
Filter
1
Rock
2
Metal