

# Twice Eleven

Paul R. Potts

19 Oct 2023

## 'Tis the Season

We're in the season of anniversaries, of both births and weddings. On September twenty-sixth, I celebrated my fifty-sixth birthday. The next day on september twenty-seventh, my son Benjamin turned ten. On October fourteenth, my son Sam celebrated his seventeenth birthday. On October twenty-ninth, Veronica and Joshua will be nineteen and fifteen, respectively. Tomorrow, on October twentieth, Grace and I will celebrate our twenty-second wedding anniversary. If you add up all the numbers, you get 255, which is the largest unsigned value that can be represented in a byte. That can't possibly be a coincidence. Well, OK, it can be, but it's funnier to imagine that it isn't.

Grace makes a special meal for everyone on their birthday. I wanted steak, baked potatoes, roasted broccoli, and a chocolate cake topped with whipped cream and raspberries from our garden. It's pretty costly to get steaks for nine people, so we don't do it regularly, but we indulge ourselves on my birthday. Grace made my cake with sour cream, and it was very dense — here's what it looked like:

We were too impatient to let it cool completely before we put the whipped cream on it, as you can see, so it started to melt down a bit, but it was still delicious!

For his birthday, Benjamin wanted to make homemade pizzas and have a lime cake with blueberry frosting. The pizzas were pretty tasty, although it is hard to really replicate what a real pizza oven does with a home oven. The cake was wild — the cake itself contained a lot of lime pulp and juice and was quite sour, while the frosting was quite sweet, so the whole effect was like eating sour candy. That wouldn't be to everyone's tastes, but it was a cake designed for a ten-year-old, and he loved it. I didn't get the greatest photo, in part because the lighting in the family room is bad, but as you can see, the cake itself was quite colorful, as he requested:

Not to be outdone, for his birthday, Sam requested his usual birthday meal bangers and mash, and a blueberry cake with lime frosting, sort of the reverse of Benjamin's cake. Grace made Sam's cake more like a a blueberry coffee cake, not so sweet, and quite delicious. Here's a not-so-great picture of Sam's cake:

And here's a picture of Sam. Grace and I took him with us to run our food



Figure 1: "Dad's Cake"



Figure 2: "Benjamin's Cake"



Figure 3: "Sam's Cake"

pickup errands, and stopped at a Tim Horton's drive-through. I took a short walk with him at a local park, and I ran around him taking pictures at various angles until I got this one:

Veronica and Joshua haven't yet told us what they want.

For our anniversary tomorrow, Grace and I ordered a big tray of sushi for curbside pickup, which has become a bit of an odd tradition for our anniversary dinner, but a tradition nonetheless. Like the steaks, we usually get sushi only once a year.

## **Gardening**

Not much is happening here. We've done very little. We did pay the guy that mows our lawn to bring his brother out and do some serious trimming of the overgrown hedges and tree in front of our front walkway. It was becoming impassable. I've harvested a few herbs and a big bag of fennel seed heads, but done little else. On Sunday, I went out to try to clear out a single fire ring bed of weeds in preparation for planting some fall garlic.

I didn't get very far into that process before I awakened a quiescent nest of yellowjackets in the soil. They swarmed out and I received several stings on my hand and wrist. Fortunately I have never had a severe reaction to bee, hornet, or wasp stings, so I was not too concerned, and the swelling went down in a few hours. I was glad that it wasn't Grace who was stung, since she tends to have severe reactions. But it was certainly demoralizing. You know that your gardening has gone off the rails when you start wondering to yourself whether pouring lighter fluid into a garden bed and setting it on fire would be safe for the produce you plan to grow there. (It wouldn't.)

The yellowjackets have won this round. Grace and I will deal with them in the spring. We did identify another location to try. So, I ordered some Vietnamese red garlic from Nature and Nurture seeds. We've got some rotted straw and bagged soil on hand. This weekend, inshallah, we will plant a bed of fall garlic.

Despite our complete failure to do much in the way of maintenance of the garden this year, it continues to produce a lot of goodies — herbs of various kinds including lavender and lemon verbena, tons of mint, raspberries (everywhere, including places we don't want them), and various greens. But parts of it look like complete chaos with weeds everywhere, so it's not exactly anything we'd want to show on a garden tour.

## **Work**

My job is going along; I'm still a contractor working fully remotely. It's been almost a year, and my whole team has grown, and we've developed our knowledge. Mostly, I like my co-workers. They are generally all supportive and helpful. Several of the other contractors who started at the same time as me were abruptly



Figure 4: "Sam"

terminated when their contracts expired, but mine was renewed. I've been asked by a younger co-worker to mentor him in C programming, so we're starting a series of regular meetings to talk about that. I get to be a professor again, sort of.

I still have no paid time off, and the expenses for our family of nine aren't shrinking, even if technically the rate of inflation is dropping. That's not much of an economic miracle unless one gets routine raises that outpace inflation. As an example — we just had our septic tank pumped. That cost us \$975. The same service two years ago cost \$700. But Paul Krugman and the other very smart economists are talking about the wonderful state of the economy and the awesome success of Bidenomics, so I must be mistaken.

I haven't taken any time off to speak of since last Christmas, and that's not good for my concentration and productivity. I'm planning to take the week off between Christmas and New Year's, but taking a week unpaid is going to eat up a lot of the money I've been able to (very slowly) save over the past year.

Last month my boss encouraged me to apply for a full-time permanent job opening, but when I took a look at it, I realized that it would require me to accept about a 45% reduction in take-home pay compared to what I'm earning as a contractor. I've done some calculations, and even though it would be great to have health insurance premiums paid for, that's too big a cut. I don't think we could cover our basic expenses. I'd be making less than when I started working for Thorlabs in 2015 and we started making arrangements to move back to Washtenaw County. Basically, the position is the wrong grade for someone with my experience.

Meanwhile, a reorganization is starting, and I'm going to have a new manager. That always is a reason to be concerned. I'm not planning on bailing out soon, but I'm not sure I'll ever be able to turn this work into a secure full-time position with a salary that will meet our needs, so I have to consider other options.

## Health

I'm still having trouble with my hearing. I have constant tinnitus, and I'm very sensitive to loud noises. Last night Elanor was banging on a drum and blowing a plastic recorder behind me at dinner time, and I hadn't put on my 3M Peltor ear protector headset or put in the much smaller Loop earplugs. As a result, last night and this morning my tinnitus was much worse. Just one good scream or blast of noise can trigger a temporary worsening of my hearing, as if my hearing was a PA system that was producing feedback from the microphones picking up sound from the speakers. It's very strange.

I've recently discovered that several of my symptoms seem like they may be related to the vagus nerve. It's well-established now that COVID can damage vagus nerve function.

The vagus nerve winds through the body and affects a whole bunch of systems.

Its function is generally to trigger various autonomic functions. So, when it isn't working right, you can wind up with "dysautonomia," which is an umbrella term for many problems that can occur when the autonomic nervous system isn't doing what it is supposed to do. For many people with dysautonomia symptoms, it doesn't regulate blood pressure correctly, and they wind up with POTS (postural orthostatic hypotension). I've long thought that the tingling and burning and numbness in my hands and feet, and purple toes, might be related to microclots, or capillary damage, but apparently this can occur due to improperly regulated circulation due to a malfunctioning vagus nerve. Tachycardia on even slight exertion is another symptom, when the vagus nerve doesn't transmit the signals that regulate the heart rate.

Recently, I've been having digestive problems too — I have heartburn for hours after each meal, despite taking omeprazole and sometimes simethicone and famotidine. My stomach doesn't seem to be emptying for a long, long time. I discovered that this is called "gastroparesis" — the digestive system is literally paralyzed. This can be because the vagus nerve isn't telling it to do its job, so the stomach just sits there, full of food and acid, with unpleasant results. It is not that severe in my case, as I haven't been vomiting or losing weight, but it's concerning.

So how do you treat this? It's not that clear. There are some medications, but I'd have to convince my doctor that this is really happening and he should prescribe them. I have lost confidence in my current doctor, and his medical practice, for various reasons, including the fact that they seem to have abandoned all COVID infection precautions, and Grace has had some bad experiences with him, and the practice, recently.

There's a battery-powered electronic device, sort of like a TENS device used to treat pain, but specifically to apply a mild electric current to the vagus nerve via some points in the neck. There are some similar devices that work on the outer ear. Both are considered non-invasive. There is an invasive version of these devices that are implanted, like a pacemaker, with wires wrapped around the vagus nerve. I certainly hope that I will never need something like that.

There also seems to be a very low-tech solution that helps, at least a little bit. Apparently massage to the arch of the feet and the ball of the big toe can help stimulate the vagus nerve. So, I have a hard plastic cylinder with bumps on it, a foot roller, which I keep under my desk, and a few times a day I spend a few minutes rolling my feet back and forth over the bumps. It seems a bit silly but it does seem to help. I've also been asking Grace to massage my feet on some nights before bed, and it seems like that will also sometimes get my stomach moving again.

There are numerous recent papers about the role of vagus nerve damage in long COVID. It's certainly not the whole picture, but it points to some possible treatments. So, I will probably be trying a transcutaneous vagus nerve stimulation device soon. It's not something my health insurance will cover, but fortunately



it isn't terribly expensive, and I'm willing to try just about anything that is low-risk.

## The Library Database Project

I've long wanted to keep some sort of database of our entire library. Years ago I was using a Mac application called Delicious Library, which had some really nifty features. I could type in a title, and it would look up different editions via Amazon's database. I could then select the edition you had, and it would download the data and put a little image of the book's cover on a virtual shelf. I could rearrange these virtual shelves to match our real shelves. I could even use a camera to scan the bar codes on books and CDs, and it would look them up, getting them right about 80% of the time. Unfortunately, a few years ago, Amazon enshittified their database API, and those features stopped working.

Delicious Library was never perfect; it was often extremely slow, taking several minutes to update a book. I think it used Berkeley DB under the hood, but it had very limited export options. You can export a CVS file containing dozens of columns, but the data is very irregular and hard to work with. It might be possible to get at the database files that the application manages behind the scenes and open them up with Berkeley DB, but Berkeley DB is now some kind of Oracle product and I don't want to get into all that nonsense. I might be able to import the exported CSV file into Excel, or in my case actually LibreOffice Calc, but it is such a mess it may just be easier to start over.

I don't actually want to keep thousands of books in a spreadsheet, and I certainly don't want to use a proprietary application. What I really want is a true relational database, something that I can query using SQL, and create different views of the data. And I want a lot of flexibility to import and export files. The tools all have to be open source.

What I settled on is SQLite. SQLite is a library and a standard file format that allows applications to work with small databases. (Even our library, with thousands of books, CDs, and videos, is still a small database for this kind of purpose; even my laptop will easily hold the whole thing in memory, and queries should be pretty much instantaneous).

The "front end" I'm currently using is a program for Mac called DB Browser for SQLite. Although it does crash occasionally after running for a long while and doing lots of editing, for the most part it is very nice and easy to work with. It's a graphical user interface, but when you're making tables or performing queries, it's using SQL under the hood. And not very far under the hood; you can open up a window and see what it's doing. You can also write your own SQL and execute it if you want to.

The database file is on my Synology file server. I save my work frequently and the server backs itself up to a cloud storage service every week, so I'm not terribly worried about losing work due to an occasional crash. The files are so small

that it is very easy to periodically make a ZIP file of the whole folder and give it a date. The latest one, which includes all the data, as well as a number of exported CSV and HTML files, takes up only 86KiB. Text files are very small compared to image files and video files.

## SQL

SQL stands for “Structured Query Language,” and it is a very old, but still widely used, tool for talking to databases and getting them to do your bidding.

Years ago, when I was in college, I was an English major, not a Computer Science major. I took all the programming classes I could fit into my schedule, and so got a minor in Computer Science. Some of my classmates who actually completed the Computer Science major took a course on databases, but I did not. So that was a bit of a hole in my education.

A few years later I learned how to write front ends for Oracle using Apple’s WebObjects, and then later to debug stored procedures written in PL/SQL. Along the way I did teach myself a bit about databases, and I had to use basic SQL to work with Oracle and PostgreSQL, but I didn’t ever consider myself an expert.

I last used SQL in a job over twenty years ago. So it’s been an interesting challenge to try to get back into it. I don’t really love the syntax of SQL, and I often have trouble figuring out where to put clauses to make them work, but there’s no denying that it is an incredibly useful and powerful tool, and one can’t really be a well-rounded programmer without knowing at least a little bit about how to use databases and SQL.

A home library might seem like it would require only a very simple database, but our home library is unusually large. Also, as your friendly neighborhood computer geek, I want the database to be structured just so. I was unable to find a sample database schema that would do what I wanted, so I have come up with my own. I’ve refined it over time, as I gradually add data to the database.

One of the nice things about this GUI tool is that if I want to change the table design, which I’ve done as I gradually worked out what I wanted to do, it’s easy to do. If for some reason I can’t modify the existing table, which happens sometimes if I want to make major changes, I can dump the data out to CSV files, destroy the table, make a new table, and import the data back into the new table. So I’ve been able to rip up and rearrange the database structure as I go, although it has settled down for the time being.

I’m sure that I’m not using keys in an optimally efficient way, but given the size of the database, that doesn’t really matter. I’ve also had to accept the idea that I’m not going to be able to complete this project quickly. Right now I’ve got just over three hundred paper books in the database. I estimate that there may be five thousand to go. The data entry is quite tedious. I try to chip away at it, doing a few books a day. As I go on, I’m getting more efficient at it. And as I

go on, I may be able to find ways to speed it up, such as importing chunks of data from the aforementioned CSV files dumped from Delicious Library.

Let's take a look at my database design.

### My Database Design

I would like to have some nice entity relationship diagrams to include here, that show how the tables and keys fit together. There are a number of tools that will make such diagrams from SQL, including some open-source programs that run on Macs, and some web-based tools that supposedly come with free trials. I tried an open-source tool, implemented in Java, but had endless problems getting it to work, due to compatibility problems with JDBC libraries on Apple Silicon. I was once pretty good at solving problems with Java, but that was over 20 years ago and I was paid to do it. I also tried one of the online tools that looked promising. I signed up for a free trial, and the tool would import my SQL, but I could not get it to make the diagrams I was hoping for. So, for the time being I have given up on making nice diagrams from my SQL.

Instead, I'll just show you my SQL. If you want to play with this design, it ought to be pretty easy to use just about any relational database, such as PostgreSQL, to create a database with the same schema.

**Authors** The authors table can be created using the following SQL:

```
CREATE TABLE "Authors" (  
    "Name" TEXT NOT NULL,  
    "Short Description" TEXT NOT NULL,  
    "Bio URL" TEXT,  
    "Notes" TEXT,  
    PRIMARY KEY("Name")  
);
```

The “Authors” table exists to maintain a set of canonical author names. They are entered as first name, last name. The name is the primary key. Other tables can link to these authors using foreign keys, and these references must match canonical author names. The required short description contains years of birth and death (if the author has died), and nationality. The optional “Bio URL” field contains links to a biography on Wikipedia or other online source describing the author. In the optional “Notes” field, I sometimes include information about pen names and alternate spellings of foreign names. Currently, there are 196 authors in this table. The first entry is for James Agee, and the fields look like this:

- Agee, James
- American novelist and journalist 1909-1955
- [https://en.wikipedia.org/wiki/James\\_Agee](https://en.wikipedia.org/wiki/James_Agee)
- NULL

(The NULL indicates there is nothing in the “Notes” column.)

**Publishers** The publisher table is generated by the following SQL:

```
CREATE TABLE "Publishers" (  
    "Name" TEXT NOT NULL,  
    "URL" TEXT,  
    "Notes" TEXT,  
    PRIMARY KEY("Name")  
);
```

The name in this case is the name of the publisher. A typical entry looks like this:

- Berkley
- <https://www.penguin.com/berkley-overview/>
- Now an imprint of Penguin

Due to the truly massive amount of consolidation that has occurred in the publishing industry, the majority of the publisher names on the spines of my books now refer to companies that are either imprints of other publishers, subsidiaries of other publishers (sometimes two or three layers deep, after a long history of mergers or acquisitions), or completely defunct. This consolidation has been a disaster for small publishers and diversity in the book market. By recording the publisher name as it appears on the books, as well as some current information about the publisher, if I can find any, I’m trying to leave bread crumbs, which might help someone in the future to find more information about an edition, even in cases where the publisher displayed on the book has changed its name, often more than once.

**Paper Book Types** This table contains canonical descriptions of different types of paper books. The SQL is very simple:

```
CREATE TABLE "Paper Book Types" (  
    "Book Type" TEXT NOT NULL,  
    PRIMARY KEY("Book Type")  
);
```

The editions table refers to the book type as a foreign key, so that the book types follow a canonical set of descriptions. Here are a few of them:

- Chapbook
- Hardcover
- Trade Paperback
- Mass Market Paperback
- Jacketed Hardcover with Traycase
- Jacketed Hardcover with Slipcase
- Hardcover with Slipcase, No Jacket as Issued
- Paperback Book with Unusual Features

“Paperback Book with Unusual Features” covers special editions like Haruki Murakami’s *The Strange Library*, a slim little art book with covers that fold and wrap around the entire book, so that you open it as if you were unwrapping a present. This list of paper book types will grow as I add books that are unusually sized, boxed sets, books that are collections of pamphlets, issues of magazines, etc. If the book is more or less a standard kind of book, like a slipcased hardcover edition, but has interesting features like foil printing, a psychedelic cover, a fabric-wrapped cover, die cutting on the cover, dust jacket, or slipcase, silk-screening on the page edges, tip-ins, included floppy discs or CD-ROMs, etc., I describe those features in the “Notes” field.

**Paper Book Shelving Areas** This is another simple table:

```
CREATE TABLE "Paper Book Shelving Areas" (  
    "Description" TEXT NOT NULL,  
    PRIMARY KEY("Description")  
);
```

I use this table to hold a canonical list of shelving *areas*, not shelving units. For example, I currently have two bookshelves devoted to Library of America books. They get a single name in this table. Sometimes the names just refer to a distinct section of books within a shelving unit, such as our set of books from New York Review Books, which are shelved together because they form a rainbow of matching volumes. Technically I could interleave them with other books in the other shelving areas by author, but they look cool shelved together:

**Paper Book Editions** This table is where things start to get interesting, since three of the columns contain foreign keys that refer to columns in other tables; this is where the “relational” in “relational databases” really comes into play.

```
CREATE TABLE "Paper Book Editions" (  
    "Author" TEXT,  
    "Other Contributors" TEXT,  
    "Title" TEXT,  
    "Publisher" TEXT,  
    "Book Type" TEXT,  
    "ISBN-13" TEXT UNIQUE,  
    "Edition Notes" TEXT,  
    PRIMARY KEY("ISBN-13"),  
    FOREIGN KEY("Publisher") REFERENCES "Publishers"("Name"),  
    FOREIGN KEY("Book Type") REFERENCES "Paper Book Types"("Book Type"),  
    FOREIGN KEY("Author") REFERENCES "Authors"("Name")  
);
```

The primary key for this table is the ISBN-13, “canonicalized” into the form `###-####-###-#`. Why this form, with the dashes placed where I’ve



Figure 5: “New York Review Books, the First Three Shelves”

placed them? It's complicated, but it has to do with the way ISBNs are broken into a heirarchy of variable-length fields, and I wanted a generic format that would be consistent for all of them.

ISBN doesn't work for everything. Editions can have the same ISBN with different cover art, or different physical formats. Some books have no ISBN number. Folio Society editions don't have them. Some Subterranean Press editions don't have them. And many older books don't have them. In this case I make up a number that includes the publisher name followed by a number, for example, Folio-001. If there is anything else that seems important to distinguish this edition from other editions of the same book, I put it in the notes field.

The "Other Contributors" gives me a field I can use for translators, editors, illustrators, and people who contribute introductions, prefaces, forewords, afterwords, or whatever, as specified on the title page or cover.

This scheme allows me to have multiple editions of books with the same title and author. For example, I might have editions of the same original book translated by two different translators. It also allows me to have multiple copies of the same book, since this table does not actually hold the books themselves. They're in the next table.

**Paper Books** At last, the books. This table mostly references other tables:

```
CREATE TABLE "Paper Books" (  
  "ID" INTEGER UNIQUE,  
  "Brief Author & Title" TEXT,  
  "ISBN-13" TEXT,  
  "Shelving Area" TEXT,  
  "Shelf Number" INTEGER,  
  "Shelf Order" INTEGER,  
  "Copy Notes" TEXT,  
  PRIMARY KEY("ID" AUTOINCREMENT),  
  FOREIGN KEY("Shelving Area") REFERENCES "Paper Book Shelving Areas"("Description"),  
  FOREIGN KEY("ISBN-13") REFERENCES "Paper Book Editions"("ISBN-13")  
);
```

The "Brief Author & Title" field is not the full title or the canonical author or publisher. The full title is in the "Paper Book Editions" table. Indirectly, the editions table references the authors and publishers, so this field is technically redundant and not strictly necessary, but because I can't read ISBNs while I'm entering data into the "Paper Books" table, it is easy to get confused and lose my place. To avoid that, I wanted a readable way to easily identify the books in this table. So, the "Brief Author & Title" contains the author's last name and the book's title, as in "Proust: Swann's Way."

I'm not entirely happy with this design, due to the redundancy, but in practice it has proven helpful. If I could easily edit the data in the "Paper Books" table

while I could *see* the indirectly linked author and title on the screen, populated after I enter the ISBN-13 which is used as a foreign key, I wouldn't need this field. I'm sure there are ways to do this, but they probably involve tools that build forms. The lightweight DB Browser for SQLite tool doesn't seem to do that, and I'm not sure what tools I might try next.

The "Shelf Order" field needs a bit of explanation. Sometimes, author and title aren't enough to specify the order that I use for books on the shelf. This is true of anthologies and omnibus editions containing work by a single author. For example, I have Library of America editions of the novels of Philip K. Dick called *Four Novels of the 1960s* and *Five Novels of the 1960s and 70s*. On the shelf, I put the books in chronological order, rather than alphabetical order by title. But the database normally sorts books in alphabetical order by title, after shelf number and author. Thus, they appear out of order in the database, since alphabetically, "Five" comes before "Four."

"Shelf Order" can be NULL, and is NULL in most cases, so that I don't have to specify a value for every book; alphabetical order is the default. But this allows me to specify an ordering for a sequence of books by an author, when alphabetical order isn't what I want.

**Sample Views** These are the first six shelves of our Library of America collection:

Here's a SQL statement to create a view to show the Library of America books that have authors (that is, they are single-author volumes, not anthologies or collections of some sort):

```
CREATE VIEW "loa-shelves-single-author-volumes-view" AS select
    'Paper Books'. 'Shelving Area',
    'Paper Books'. 'Shelf Number',
    'Paper Book Editions'. Author,
    'Paper Book Editions'. Title,
    'Paper Book Editions'. Publisher,
    'Paper Book Editions'. 'ISBN-13',
    'Paper Book Editions'. 'Book Type',
    'Paper Book Editions'. 'Edition Notes',
    'Paper Books'. 'Copy Notes'
from
    'Paper Books'
inner join 'Paper Book Editions'
    on 'Paper Book Editions'. 'ISBN-13'='Paper Books'. 'ISBN-13'
where 'Paper Books'. 'Shelving Area'="Library of America" and 'Paper Book Editions'. Auth
order

    by 'Paper Books'. 'Shelving Area', 'Paper Books'. 'Shelf Number', 'Paper Book Editions
    case 'Paper Books'. 'Shelf Order' when null then 'Paper Book Editions'. Title else
```





Figure 6: “Library of America First Six Shelves”

That did not come easily; I had to read some SQL reference materials and scrutinize a number of different examples to fit those pieces together, but it works. The view makes it look like I have a nice simple table of just the Library of America single-author volumes, when really it is pulling data from the database and organizing columns from multiple tables.

Here are the first few rows produced by the above view, in CSV form. CSV is a bit awkward to read because some fields are quoted, and some aren't. Quoting is normally used when the field contains the separator character, which in this case is a comma. For readability I have exported these records using a space as the field separator, which makes the data a bit more legible. When it's done this way all strings that contain spaces are quoted. I've also inserted blank lines to make it clearer where the rows begin and end.

```
"Library of America" 1 "Agee, James" "Let Us Now Praise Famous Men, A Death in the Family, a
"Library of America" 1 "Alcott, Louisa May" "Work, Eight Cousins, Rose in Bloom, Stories & C
"Library of America" 1 Anderson,Sherwood "Collected Stories" "Library of America" 978-159853
```

I can use **pandoc** to convert the CSV to Markdown, and put the Markdown in this document for processing by **pandoc**. That will produce a nice HTML table, although it is very wide, and with the style sheet I'm using for this newsletter, the HTML likely won't be very readable in your browser, and in the PDF, the table will march right off the edge of the page. But here it is anyway:

Shelving Area	Shelf Number	Author	Title
Library of America	1	Agee, James	Let Us Now Praise Famous Men, A Death in the Family, a
Library of America	1	Alcott, Louisa May	Work, Eight Cousins, Rose in Bloom, Stories & C
Library of America	1	Anderson,Sherwood	Collected Stories

The point is not so much not that I can read the data in an ideal way, but that using a database like this gives me a lot of options for different kinds of reports.

One thing I haven't figured out is how to ignore leading articles like "the," "a," and "an" while sorting. There are ways to do this, but I haven't tried them yet, and I'm not sure which ones are best practices (or even which ones work) in SQLite. For now, if I have a problem with sorting, I fix it using the "Shelf Order" field, but I don't want to have to rely on this solution as the database gets larger; in some cases I have a lot of books by specific authors, and I don't want to have to maintain shelf order numbers for all of them.

For the Library of America anthologies, the view of those books is created a little bit differently; I don't want to show the author, which is NULL; in fact, I select only the Library of America books with empty author fields:

```
CREATE VIEW "loa-shelves-anthologies-view" AS select
  'Paper Books'. 'Shelving Area',
```

```

    'Paper Books'. 'Shelf Number',
    'Paper Book Editions'. 'Other Contributors',
    'Paper Book Editions'. Title,
    'Paper Book Editions'. Publisher,
    'Paper Book Editions'. 'ISBN-13',
    'Paper Book Editions'. 'Book Type',
    'Paper Book Editions'. 'Edition Notes',
    'Paper Books'. 'Copy Notes'
from
    'Paper Books'
inner join 'Paper Book Editions'
    on 'Paper Book Editions'. 'ISBN-13'='Paper Books'. 'ISBN-13'
where 'Paper Books'. 'Shelving Area'="Library of America" and 'Paper Book Editions'. Author
order

    by 'Paper Books'. 'Shelving Area', 'Paper Books'. 'Shelf Number',
       case 'Paper Books'. 'Shelf Order' when null then 'Paper Book Editions'. Title else

```

### How Long Will This Take?

I've completed about 22 individual shelves. There are 130 more shelves of books to catalog, not counting the children's books in the family room, the DVDs, the Blu-rays, and the CDs. So, this is obviously going to take a while. I'll be looking for shortcuts, but there's going to be a lot of data entry and editing. I am hoping that the catalog might be fully complete by this time next year. As I go, I'm taking photographs of each shelf. These are going into folders on the server as a reference. These can be updated periodically. I don't expect them to stay perfectly synchronized with the database, since the collection slowly changes. As a shortcut, if I add books, I can just add photos of the new books to the folders, rather than taking a whole new set of pictures of the shelves.

It would be nice if I could incorporate the photos right into the view, but so far my experiments putting JPEG file into "blobs" in the database made the DB Browser application run incredibly slowly, even though the JPEG files really aren't that large. So I'll be avoiding that for the time being.

The fact that this is all built on SQL and text suggests that I might be able to come up with future uses for it in the future, such as a web-based front end for checking out books. But I'm not going to dive into that anytime soon, and I'm not going to "overdesign" for future front-ends that may never happen.

I have more to say, but if I keep trying to add more, I won't get this out for weeks or months, so I'm going to cut it off here and send it out.

Have a great week!

## **About This Newsletter**

You can find all my newsletters in my writing archive.

As always, this content is available for your use under a Creative Commons Attribution-NonCommercial 4.0 International License. If you'd like to help feed my coffee habit, you can leave me a tip via PayPal. Thanks!