Your First Guide to ...

# Database Design

A guide to relational database design and data organization for both beginners and technology professionals.

Andrew Comeau

**Your First Guide to Database Design**

For information regarding this book including official support and reprinting rights, please visit the official website at:
http://www.AndrewComeau.com

Or contact ...

Andrew Comeau
P.O. Box 770253
Ocala, Florida  34477
acomeau@drewslair.com

Proper names and contact information used as examples in this book, including company and individual names, are fictitious. No association with any real organization or individual is intended nor should it be inferred.

This book expresses the author's views and opinions. The information contained in this book is provided without any express or implied warranties or guarantee of fitness for a particular purpose. Neither the authors nor its distributors will be held liable for any damages, real or perceived, arising from use of the information in this book.

Cover photo obtained from Flickr.com (user: idleformat) under Creative Commons license.
Description: South Downs, near Jevington, East Sussex (England)
http://www.flickr.com/photos/idleformat/

Chapter VIII Photo obtained from Flickr.com (user: jeffreyw) under Creative Commons license.
http://www.flickr.com/photos/jeffreyww/

# Contents

This free preview includes the Introduction and Chapter I.  If you like what you see here and would like to read more, The complete book is available on the Kindle, the Nook and in PDF format at the links below …

**Amazon.com**

http://www.amazon.com/First-Guide-Database-Design-ebook/dp/B00FNAMMA8

**Barnes and Noble**

http://www.barnesandnoble.com/w/your-first-guide-to-database-design-andrew-comeau/1117049383?ean=2940148791621

**Scribd.com (PDF format)**

http://www.scribd.com/doc/174069216/Your-First-Guide-to-Database-Design

**Visit the book's official page at:**

http://www.andrewcomeau.com/Pages/db-design-guide.aspx

# Introduction

For all the books that have been written on database design, I decided to write this one to emphasize specific areas of design and to bring these ideas to a much wider audience. While the people who commonly design databases such as database administrators, software programmers and developers make up around 1% of the U.S. population and likely a similar percentage worldwide, an understanding of how to organize data and design a strong system to store it is helpful to anyone who needs to work with information or understand how software works.

## Who should read this book?

Personally, I would say *everyone* but then, I'm a little biased. Still, I've written this book with everyone in mind in hopes that it will benefit the most people. Ideally, you should read this book if you can say "Yes" to any of the following:

- *You work with large amounts of information on a regular basis and would like to know how to organize and understand it better.* This can be true even if you never work directly with an I.T. department. Maybe you're an accounting professional trying to make sense of expense data or an administrative assistant who's been asked to put together a database for payroll or employee information. Regardless of your job, an understanding of databases will enable you to produce better results and reduce your work in the long-term.

- *You communicate on a regular basis with I.T. professionals and need to do it more effectively.* Even if you never design a database yourself, knowing the principles involved goes a long way toward understanding the abilities and limitations of the software you work with. This will save you a lot of misunderstandings when asking for new reports or system enhancements.

- *You want to enhance your career and need a better understanding of software.* A promising career can start simply by being the person in the office who can produce results nobody else can. Database design is a valuable skill that will continue to be relevant for many years to come. While reading this book won't add it to your resume immediately, knowing the basics can put you in the position to volunteer for that project that will get you noticed later.

- *You're a hobbyist* with a copy of Microsoft Access or another database program and you want to organize your media collection or exercise program.

- *Finally, you're an I.T. professional*, maybe even a programmer or software developer, who is still a little fuzzy on the details of database design. Maybe you're just coming back to it after years of doing other work and need a refresher course. Having a reference by your side can be very helpful when working on your latest project.

That's still not quite everyone but it's a lot more than the 1% I mentioned above. Chances are, if you're reading this introduction, then you can benefit from the knowledge of database design you'll find in this book.

## Why *this* book?

Right now, you're probably comparing this book to others on database design and trying to decide which one to buy so what makes this one different?

First, this book is written for *you*. No matter who you are or how little experience you have with database and technology concepts, this book is written so that you will understand the ideas presented here. Plain English explanations of ideas come first; technical terms and jargon, second. The multiple examples used here are based on everyday situations that the average person can relate to.

Also, the focus here is on the foundation-level concepts that will help you understand database design regardless of which software you're using. Software and programming languages come and go and new ones can be learned by experienced people in a short time. It's more important to know the basic concepts behind software so that when a new system comes along, you will know what to expect from it and be able to evaluate and learn it that much faster.

A final reason to select this book is for the author's experience. I have been working with databases since the 1990s when I started designing some database programs for the company I was working for. Some of the examples in this book are based on actual systems I've designed, such as the Job Search Plus application. While I eventually ended up as a programmer rather than a database administrator, I have maintained a special focus on database design and it's a part of software development that I enjoy. Now, with the renewed interest I'm seeing in the field of software design, I've decided to share my experience with you through this book.

## Online Resources

There are certain limitations with both hard copy and eBook formats so, in order to ensure as much value to my readers as possible, I've setup an official page for this book on my website at http://www.AndrewComeau.com where you can find extra content including answers to reader questions, diagrams and downloads to supplement the material in this book. If you have any questions on anything you find here, that's the first place to stop.

## I want to hear from YOU!

As an independent, self-published author, I am not affiliated with any publishing house or software company and my only agenda is to provide you with the quality materials and knowledge you need. This book represents my own work as a professional and the message here is conveyed straight to you without interference or shaping by a third-party. Your support of self-financed, independent authors helps to ensure that quality, affordable alternatives like this one can continue to be available to the public.

If you have purchased this book or if you've even downloaded a sample and have more questions, I'd like to hear from you. I've always enjoyed corresponding with the visitors to my websites and the readers of my previous book, Microsoft Access for Beginners and if you would like to contact me, you're welcome to do so by e-mail at dbdesign@drewslair.com.

Andrew Comeau
Ocala, Florida
September 2013

# Chapter I: Databases: In Theory and Everyday Life

## What is a Database?

The word *database* is not something you hear in everyday conversation. It's one of those technical terms that's used by business and I.T. people. It might evoke images of computers and long reports of names and numbers or indecipherable data. Some people might think of marketing or mailing lists. In fact, a database is simply any collection of data that's organized so that it can be retrieved as needed. Usually, it refers specifically to data that has been stored within a computer system so that it can be quickly manipulated into reports. Databases take many forms but anytime a computer needs to present information of any kind, whether it be a store's customer data or patient data at your doctor's office, that data has to be retrieved from somewhere and it's usually in a database of one kind or another.

## Everyday Database Examples

Our everyday lives are constantly influenced by electronic data of one kind or another whether it's the web pages and e-mails on our computers or the uplink from a credit card terminal when we buy gas. In June 2011, the EMC corporation, a worldwide I.T. consulting firm, released a study which stated that the world's collection of electronic data was doubling every two years and forecasted that in 2011, 1.8 *zettabytes* of data would be created or copied. A zettabyte is 1*billion* gigabytes of information. To show you what that figure means, the average new home computer in 2012 might have had 500 gigabytes of storage included with it. That means the amount of data estimated to be generated in 2011 would fill *3.6 million* home computers or more than 2.7 *billion* CDs. That's a lot of data to store.

A large portion of this data is stored in separate files such as Microsoft Word documents and image files but much of it needs to be stored in a way that can be quickly accessed and searched by record. For some examples of this, let's look at the different ways you might access databases throughout the day.

- You get up in the morning and, if you're like me, you check your e-mail which means that your computer connects to whatever service you use and requests a list of new e-mails that have come in since you checked last. Your program retrieves at least the essential parts of the e-mail including the send and receive information and subject line. All of this information is stored within a database on your e-mail service and, if you use a program like Microsoft Outlook or Mozilla Thunderbird to manage your e-mail, it's downloaded to your e-mail program's database on your local computer.
- Many of your morning e-mails are probably spam which means that your e-mail address is stored in the database of a marketer somewhere, probably several.
- After you get the incoming e-mail, you decide to send an e-mail to your friend, Bob, asking if he wants to get together after work that evening for a movie. In the "Send To" field, you type in the first characters of Bob's name because the program accesses the

modern version of an address book to get Bob's e-mail address. Address books and Rolodexes of all kinds have always been a type of database with fields for the name, phone number, address, etc. but now they're often *electronic* databases that can be quickly searched and updated.

- Watching the news as you eat breakfast, you notice the headlines scrolling along the bottom of the screen along with the weather and other information. Whatever program the news service is using to display this information is accessing a database that stores the headlines so that they're easily updated as the news comes in. If you get your news from the Web, the site you use is probably using content management software which is a type of database that stores information on the news stories shown.

- On your way into work, you stop at the local coffee shop to get your venti coffee with a triple shot of espresso. When the cashier swipes your card, the software requests your record from the credit card company's database to verify that the card number you just provided is valid. You might swipe your card other times throughout the day at the restaurant for lunch, at the gas station on the way home, etc.. If the gas pump asks you to verify your ZIP code, that data is part of the record retrieved from the database, too.

- If the building you work in has a moderate amount of security, you might have to punch in a code, swipe a security badge or even scan your fingerprint to get in the door before you can get to your desk or work site. The system then consults a security database in which you are hopefully still listed as a current and authorized employee and decides whether to admit you or make your morning more challenging. Depending on your company's policies, the card swipe and its result might be recorded back to the security database for reference.

- If you use a computer at work, any number of the programs you work with access data from any number of company databases to provide customer, order, shipping, employee or financial information that you use in your job. The computer itself, your printer and the phone you use might be listed in one or more inventory and administrative databases on your company network.

- One or several times that day, you might check Facebook or one of the other social networking sites. Each one of your friends' posts with stories about their kids, pictures of their cats and all the personal details they've entered are stored in the site's database for easy retrieval and sorting.

- After you leave work, if you're in too much of a hurry to meet Bob for the movie, you might get pulled over. After the officer takes your license and registration, he will help you learn to slow down by taking his time checking any record you might have in various local or state databases. If he gives you a ticket or written warning, the details of that citation will be entered into a database so that the necessary office, or court, can process it.

- Finally on your way home for the night, you stop by the pharmacy and the grocery store. The pharmacist uses a database, likely more than one, to verify the details of your prescription order and check for any interactions or side effects you need to know about. The clerk at the store scans the barcodes on your items which are simply numbers that the register uses to look up your items in a database to get the right price and adjust inventory.

- You probably used your credit card again at the grocery and pharmacy. These transactions are stored in a database on your credit card company's system along with the ones for the the venti coffee, the meal at the restaurant, the tank of gas and the movie tickets so that they can send you a statement at the end of the month.

In most of these examples, I talked about single databases being used to store data but since companies and government agencies love to keep information, the data can be linked to, duplicated, manipulated, sliced and diced many times over. As I've *repeatedly* warned people on a certain social network, once you put your information onto someone else's network, you lose all control of where it will go and who might use it. Essentially, you pay for the privilege of sharing your pictures and personal information over their network by potentially sharing it with anyone else who might find it useful ...

... but, I digress.

My real point here is that, in this electronic age, more and more of your daily activities generate or rely on information stored in databases of one kind or another. Unless you are completely off the grid, your daily life follows and leaves an electronic trail. It is in your best interests, and at worst certainly won't hurt, to learn how databases work so that you will be aware of how information is stored and shared between systems.

# Types of Databases

As I mentioned earlier, even your address book qualifies as a database. It's a list of records with a set of fields for names, addresses and other bits of contact information. It has a structure which you can use to quickly record and retrieve the information you need. Electronic databases can be as simple as a text file on a local computer where the information is stored in rows of text and as complex as multi-terabyte network databases, designed with sophisticated software costing thousands of dollars, hosted on network servers, secured with the highest levels of encryption and accessed by multiple programs and websites from around the world. It all depends on the requirements of the people using the data. The following are a few types of databases that you might see out in the real world.

## Text files - Data Exchange and Basic Storage

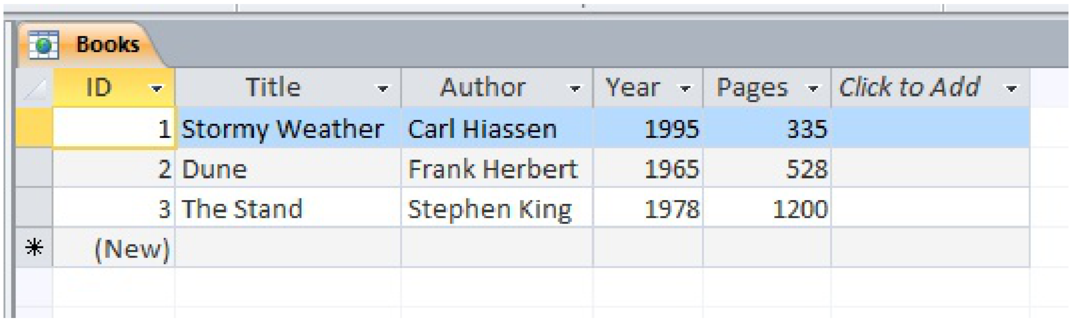
The simplest form of electronic database can be stored on a local computer or network in a plain text file. While providing the fewest features in terms of sorting and indexing, plain text formats are the most compatible with different software systems and are also readable by humans. Three prominent formats of text based databases right now are CSV, XML and JSON as I'll explain below.

### CSV

*CSV* stands for Comma Separated Values. It's the simplest database format in which the data is stored in a consistent set of fields and separated by commas or other separators such as spaces and tabs. If you were to store a book collection in this format with the title, author, release year and pages, it might look like this:

```
"Stormy Weather","Carl Hiassen",1995,335
"Dune","Frank Herbert",1965,528
"The Stand","Stephen King",1978,1200
```

Other names for this format include *comma delimited data* and *tabular data*. It's a format that goes back decades to a time before the PC and can still be used to transfer data between programs that might not read each other's native formats. For example, you could save the data in a Microsoft Excel spreadsheet to CSV and import it into another analysis program that did not read Excel format but could import CSV data.

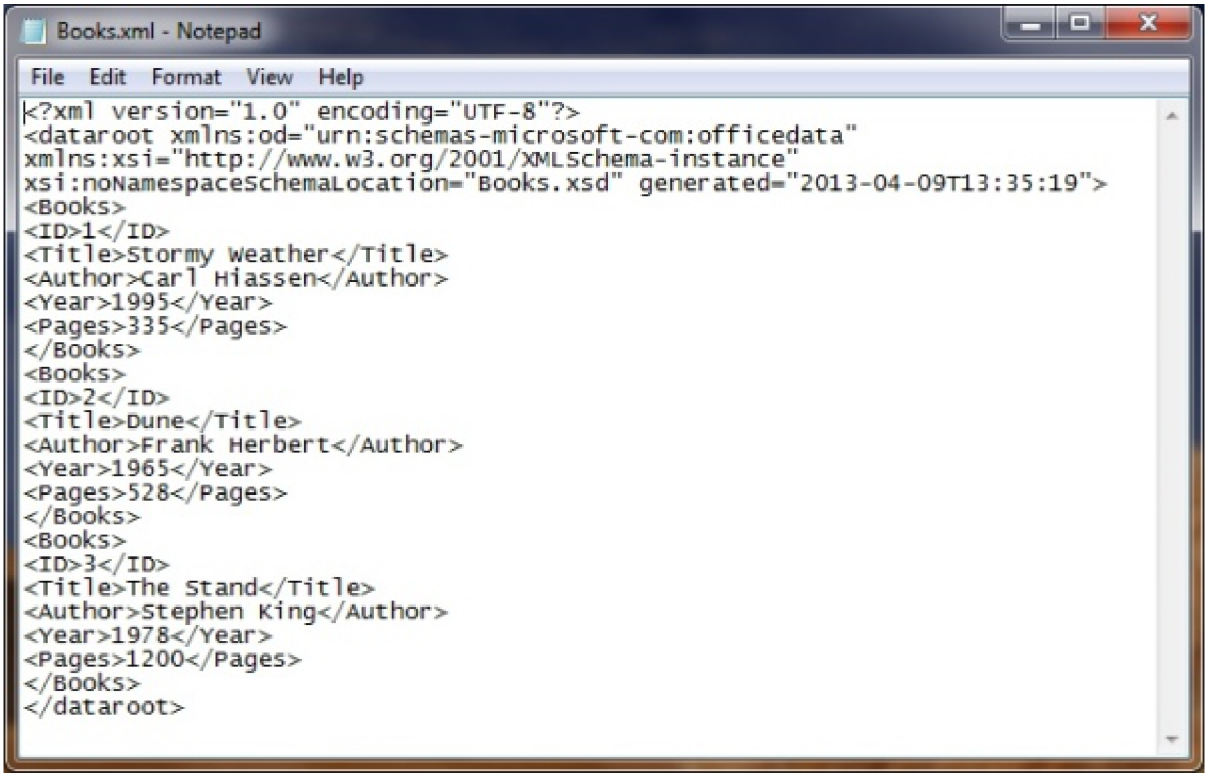


|  | ID | Title | Author | Year | Pages | Click to Add |
|---|---|---|---|---|---|---|
|  | 1 | Stormy Weather | Carl Hiassen | 1995 | 335 |  |
|  | 2 | Dune | Frank Herbert | 1965 | 528 |  |
|  | 3 | The Stand | Stephen King | 1978 | 1200 |  |
| ✳ | (New) |  |  |  |  |  |

*Figure 1.1 - CSV is commonly used for the transfer of data between programs but can also be used to store and work with small amounts of data.*

### XML

A newer standard of text data is the *XML* format. XML stands for Extensible Markup Language and was developed in the 1990s mainly for data exchange over the Internet. Figure 1.2 shows a sample of XML based on the book list in Figure 1.1.



```
<?xml version="1.0" encoding="UTF-8"?>
<dataroot xmlns:od="urn:schemas-microsoft-com:officedata"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Books.xsd" generated="2013-04-09T13:35:19">
<Books>
<ID>1</ID>
<Title>Stormy Weather</Title>
<Author>Carl Hiassen</Author>
<Year>1995</Year>
<Pages>335</Pages>
</Books>
<Books>
<ID>2</ID>
<Title>Dune</Title>
<Author>Frank Herbert</Author>
<Year>1965</Year>
<Pages>528</Pages>
</Books>
<Books>
<ID>3</ID>
<Title>The Stand</Title>
<Author>Stephen King</Author>
<Year>1978</Year>
<Pages>1200</Pages>
</Books>
</dataroot>
```

*Figure 1.2 - XML can be used to store and transfer data in a structured format.*

XML gets its name from the fact that the various fields are marked with the tags that you see indicating the field names such as title and author. The word "extensible" means that there is no limitation on what tags can be used. Unlike HTML, the markup language used to create web pages, the tags in an XML file are determined by the user based on the needs of the data. Extra documentation might be included in an accompanying XSD file to provide more information on the structure of the data.

XML has a couple of advantages over CSV. First, it can handle more complex data such as categories and sub-categories of report data. For example if you had a list of employees and each employee had a list of absentee days, that data could be exported to one XML file and then imported to another program with the link between each employee and his or her specific absences intact. Unlike some other data formats, XML is also readable by both human and machine although it can look a little confusing to a person who is not familiar with the initial data. The final advantage is that when transmitting over the Internet, XML can go through security precautions such as firewalls and e-mail filters without posing any risk to the system receiving it because it's simply a text file. Other 'binary' formats like spreadsheet and word processing files that could include machine-readable programming code are often blocked by filtering systems because of their potential to carry viruses.

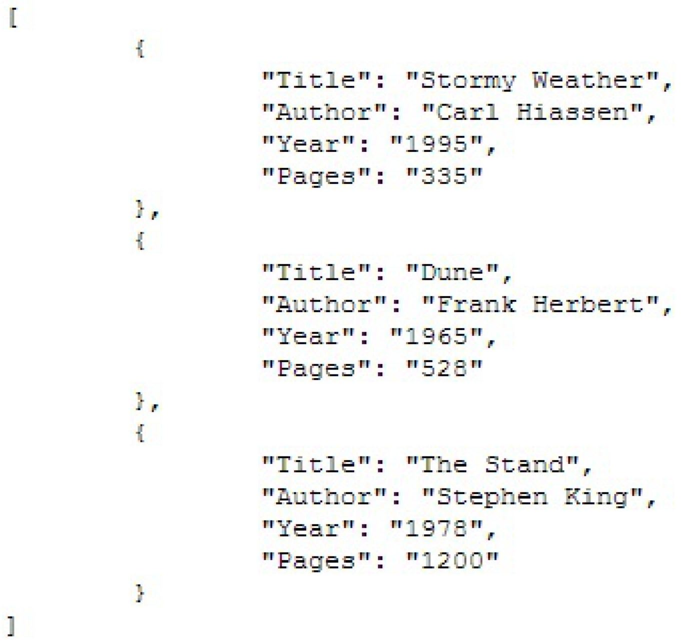
XML really shines when it comes to transmitting continuous record-based data that needs to be regularly updated and synchronized. Years ago, I worked with an automated system that maintained an inventory of the computer workstations in the company. Every time a user turned on their computer, an XML file would be generated by a small client program on their machine with their computer name and current specifications and sent to the main program on the network. The main program would then update its database of workstations with this record. When we found that the program wasn't doing such a great job of updating the inventory, it wasn't difficult for me to design another database program that would import the XML files and maintain a better inventory.

XML is also used to transmit patient data between healthcare providers. Under the HL7 (Health Level 7) standards, a doctor's office can receive a regular stream of XML files containing individual patient records and can automatically import those records into its own system in preparation for a patient visit or relay them to another doctor's office as needed.

### JSON

The *JSON* format is the newest of the three, having been developed starting in 2001 by Douglas Crockford at his company, State Software, Inc.. It was developed as a data interchange format which would be easily readable and writeable by both humans and computers and is used by some applications as an alternative to XML. The name stands for *JavaScript Object Notation* as the format is based on the JavaScript web programming language. Unlike XML, JSON recognizes actual data types such as number, string and true/false values. The format can specify individual named values as well as arrays of values and enables highly structured data to be written to a plain text format. Figure 1.3 shows a sample of JSON based on the previous list of books.

As database formats, all three of the above formats are limited by the fact that they are stored within text files. The program generally has to import the entire file to work with the data, there is no security on a text file meaning that it can be read by anyone and if it's deleted or corrupted, the data is lost. For these reasons, most programs that work with more than small amounts of data use other formats.



```
[
	{
		"Title": "Stormy Weather",
		"Author": "Carl Hiassen",
		"Year": "1995",
		"Pages": "335"
	},
	{
		"Title": "Dune",
		"Author": "Frank Herbert",
		"Year": "1965",
		"Pages": "528"
	},
	{
		"Title": "The Stand",
		"Author": "Stephen King",
		"Year": "1978",
		"Pages": "1200"
	}
]
```

*Figure 1.3 - A sample of JSON used to store book information*

## Mobile Databases - Smartphones, Tablets and the Web

Most sophisticated computer programs, including those on your smartphone, need something more than a text file to store their data. At a certain point, the data needs to conform to a structure and a set of rules. Relationships between different categories must be enforced, the data must be easily indexed and searched and the program must be able to efficiently access the right data without having to sift through an entire collection. These requirements are common enough to different programs that they can be delegated to a separate database software which will manage the database and provide access to the data by the program. This software is often written independently from the programs that you use on a daily basis and, if the program you're using is well designed, you should never even be aware of the existence of the database manager itself.

As a programmer, if I was designing software that depended on data such as a music sharing program or an appointment scheduler, I would research the available database management software from companies such as Microsoft and Oracle and then decide which one would work best based on the needs of the program I was writing and how I wanted to distribute it. I would use that software to design a database to support my program and then write my program in whatever language I was using to read and write to the database as necessary. An installation program would then enable users to install both my program and the database on their computers or devices. This is a simplified description of what is called *multi-tier design* with the program code and the database existing in separate pieces and communicating with each other. While the specifics might differ, the process is basically the same for programs written for your personal computer, your smartphone and websites.

There are many database software titles out there. Here are just a couple with examples of where they're being used. See the links section at the end of the chapter for the websites associated with the software titles listed here.

### SQLite

SQLite (pronounced SEE-kwel LITE) is a public domain database software promoted for its small size, speed and reliability. These are definitely considerations for programmers who want to design small, efficient programs and make their users happy. Since it's public domain, programmers can design their programs around an SQLite database without having to pay any licensing fees. It's used by such well known programs as Mozilla Firefox and McAfee anti-virus.

### Microsoft SQL Server Compact and Express Editions

Microsoft's SQL Server is one of the big names in the database world with various editions of the software being used for everything from small mobile apps to giant business systems. Their Compact and Express editions are designed specifically for smaller applications with the Compact edition focusing on mobile apps and the Express edition being used for desktop and website applications. These editions of the software are also free for programmers to use and are sometimes bundled with other programming tools. If you are developing websites on a hosting service or network powered by Microsoft Windows, there's a good chance that you're also using SQL Server Express for your data needs.

### Oracle Database Express Edition

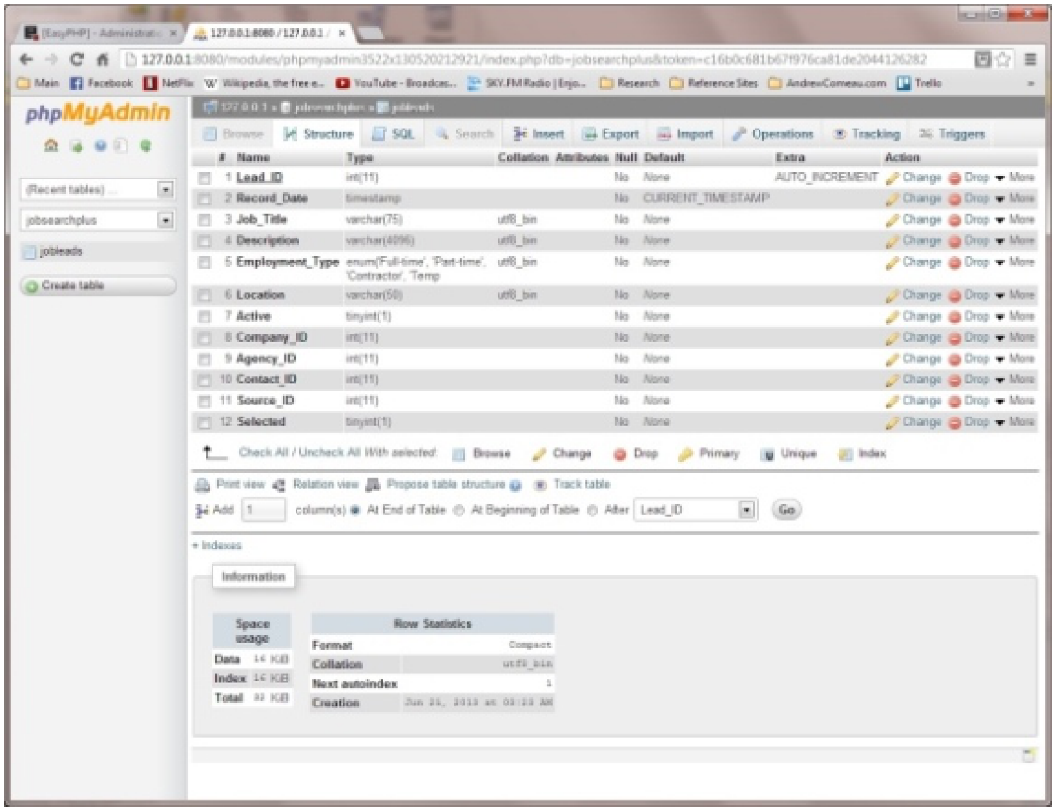
Oracle is another heavyweight in the database field. The Express edition of their self-titled database software can be used to create software and for training and software testing by programmers. Like SQL Server Express, it has limitations on the amount of data it can store and the amount of computer memory it can access but for the average developer or user with a single machine installation, it's enough to work with.

### MySQL

Also from Oracle, MySQL (pronounced MY SEE-kwel) is a database software which is popular for use with web applications, especially on non-Windows machines. It's an *open source* software which means that the source code is available for independent programmers to modify as needed and also very often means that the software is free. In my experience, MySQL and Microsoft's SQL Server Express are the two primary database systems for use with web applications. If you build your own website and sign up for a website hosting account, you'll often see one or both of these offered for use in storing whatever data your site needs to access. WordPress, the popular blogging tool, uses a MySQL database in order to store all data for the blog including posts and other content. MySQL does not have the limitations on database size that SQL Server Express and Oracle Express do and the databases can run into the terabytes (*trillions* of bytes).

## Desktop Database Software - Local Analysis

Another type of database software enables individual, non-technical users to create databases and analyze data. Several software titles offer powerful data analysis and reporting features that can be run straight from the desktop. SQL Server Express and Oracle Express might qualify for the advanced user as they can be installed locally and used to create databases but others are designed for the average business user or even the home user and are often packaged as part of a larger suite of applications. This includes programs like Microsoft Access and OpenOffice Base. These programs enable the user to create databases in individual files on the user's computer, fill them with data, manipulate and analyze the data in various ways, create reports and even link to other analysis programs like Microsoft Excel to pull additional data into the analyses. Some desktop databases include advanced features such as embedded programming languages and the ability to package user-designed databases to be run on remote machines and in Web applications.



| # | Name | Type | Collation | Attributes | Null | Default | Extra | Action |
|---|---|---|---|---|---|---|---|---|
| 1 | Lead_ID | int(11) | | | No | None | AUTO_INCREMENT | Change Drop More |
| 2 | Record_Date | timestamp | | | No | CURRENT_TIMESTAMP | | Change Drop More |
| 3 | Job_Title | varchar(75) | utf8_bin | | No | None | | Change Drop More |
| 4 | Description | varchar(4096) | utf8_bin | | No | None | | Change Drop More |
| 5 | Employment_Type | enum('Full-time', 'Part-time', 'Contractor', 'Temp | utf8_bin | | No | None | | Change Drop More |
| 6 | Location | varchar(50) | utf8_bin | | No | None | | Change Drop More |
| 7 | Active | tinyint(1) | | | No | None | | Change Drop More |
| 8 | Company_ID | int(11) | | | No | None | | Change Drop More |
| 9 | Agency_ID | int(11) | | | No | None | | Change Drop More |
| 10 | Contact_ID | int(11) | | | No | None | | Change Drop More |
| 11 | Source_ID | int(11) | | | No | None | | Change Drop More |
| 12 | Selected | tinyint(1) | | | No | None | | Change Drop More |

Add 1 column(s) ● At End of Table ○ At Beginning of Table ○ After Lead_ID

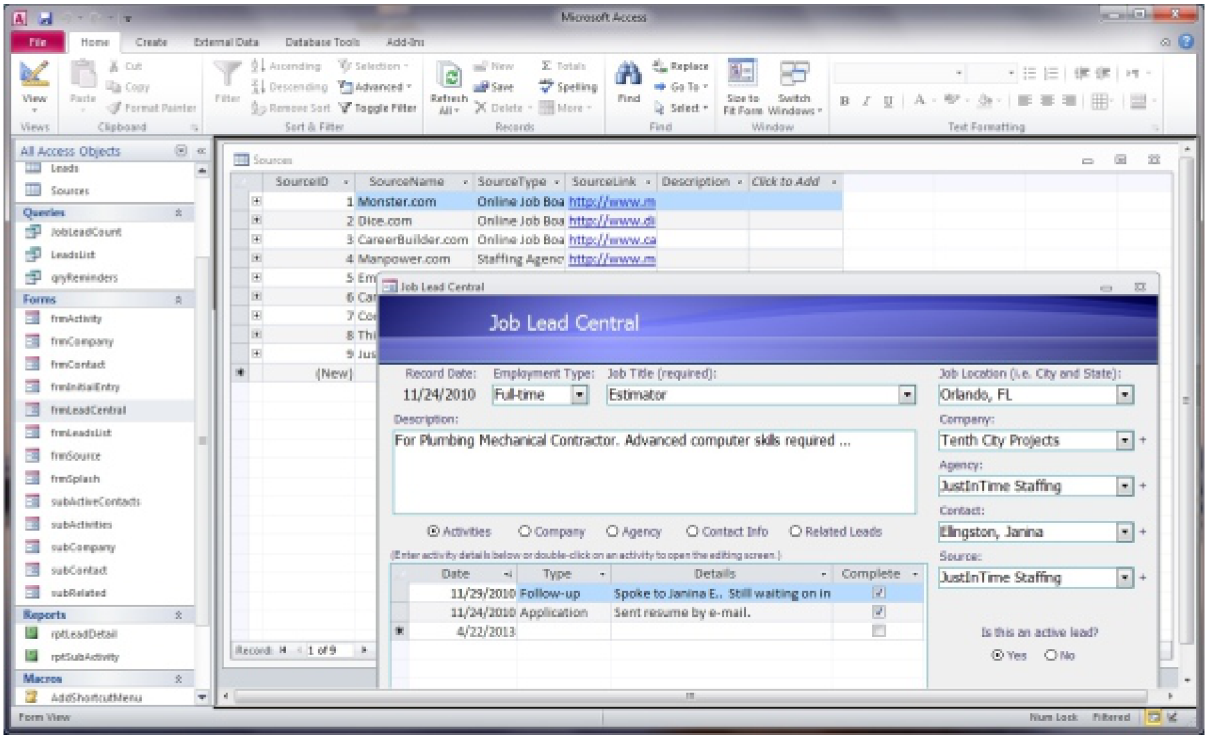
| Space usage | |
|---|---|
| Data | 16 KiB |
| Index | 16 KiB |
| Total | 32 KiB |

| Row Statistics | |
|---|---|
| Format | Compact |
| Collation | utf8_bin |
| Next autoindex | 1 |
| Creation | Jun 21, 2013 at 03:23 AM |

*Figure 1.4 - MySQL is a popular database for use with web applications and is often administered in a web browser with the phpMyAdmin web application.*

Early in my programming experience, I used desktop database software including Microsoft Access and Borland Paradox to design database applications for the company where I was working. These applications handled a variety of data including manufacturing statistics, employee and payroll information and customer communications. By using these software tools, I was able to save the company hundreds and maybe even thousands of dollars in software purchases and man hours while providing the other employees with quick access to the information they needed. The availability of such potentially useful and powerful tools is another reason it's good for you to know how to work with databases.

Unlike the mobile databases mentioned in the last section, desktop databases represent a single-tier design. A program like OpenOffice Base is the database management software that a user installs on their machine in order to create the database. Whatever forms, reports and other interface features the user creates around the database are stored within the database file and an installation of the management software is usually required in order to work with the database. Some desktop database systems such as Microsoft Access can allow for a basic multi-tier arrangement by splitting the forms and reports into a separate file called a *front end* that is linked to the tables in another file called the *back end*. Multiple copies of the front end can then be distributed to users to enable the database to be more efficiently accessed by multiple people.



Sources

| | SourceID | SourceName | SourceType | SourceLink | Description | Click to Add |
|---|---|---|---|---|---|---|
| ⊞ | 1 | Monster.com | Online Job Boa | http://www.m | | |
| ⊞ | 2 | Dice.com | Online Job Boa | http://www.di | | |
| ⊞ | 3 | CareerBuilder.com | Online Job Boa | http://www.ca | | |
| ⊞ | 4 | Manpower.com | Staffing Agenc | http://www.m | | |
| ⊞ | 5 | Em | | | | |
| ⊞ | 6 | Ca | | | | |
| ⊞ | 7 | Co | | | | |
| ⊞ | 8 | Thi | | | | |
| ⊞ | 9 | Jus | | | | |
| * | (New) | | | | | |

Record: 1 of 9

Job Lead Central

Job Lead Central

Record Date: 11/24/2010
Employment Type: Full-time
Job Title (required): Estimator
Job Location (i.e. City and State): Orlando, FL
Description: For Plumbing Mechanical Contractor. Advanced computer skills required ...
Company: Tenth City Projects
Agency: JustInTime Staffing
Contact: Ellingston, Janina
Source: JustInTime Staffing

● Activities ○ Company ○ Agency ○ Contact Info ○ Related Leads

(Enter activity details below or double-click on an activity to open the editing screen.)

| Date | Type | Details | Complete |
|---|---|---|---|
| 11/29/2010 | Follow-up | Spoke to Janina E.. Still waiting on in | ☑ |
| 11/24/2010 | Application | Sent resume by e-mail. | ☑ |
| 4/22/2013 | | | □ |

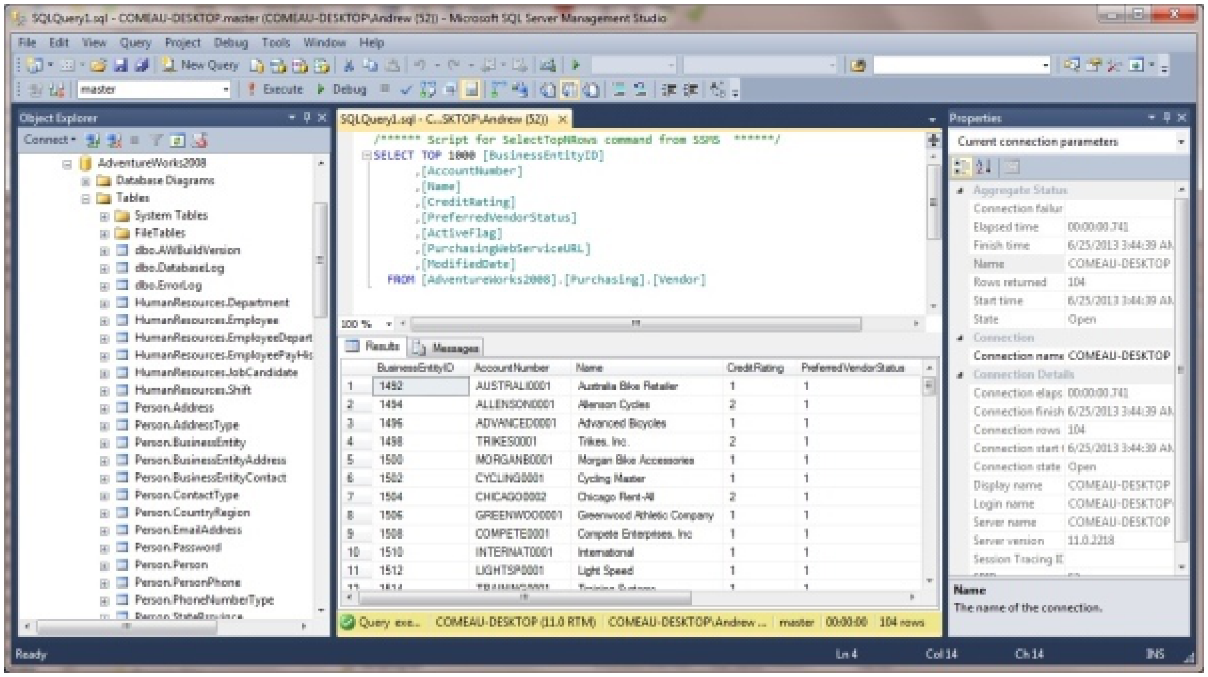
Is this an active lead?
● Yes ○ No

*Figure 1.5 - Microsoft Access and other desktop database software can be used to create sophisticated applications for analyzing data.*

## Server Databases - Organizational Data and Enterprise Applications

Most of the databases I've mentioned so far are great for moderate amounts of data, even up to a few gigabytes. They provide a structure for the data and maybe even security to determine who can access it. In the right hands, they can be used to produce very sophisticated applications. At a certain point, however, it's time to go to the next level. Mobile and desktop databases typically support a limited number of users and are also limited in the amount of data that can be stored, the security features available and other management tools that help to maintain the kind of data collections you find in companies, universities and other organizations.

Just like the other types of database software, server software can be used for small databases but is actually designed to support immense database systems containing up to hundreds of *petabytes* of data (1 petabyte = 1,000,000 gigabytes) in multiple databases, on powerful networked machines providing data access to thousands of users. These massive systems can be worlds unto themselves that contain dozens of tables, report templates and a sizeable amount of programming code used to run automatic operations on the data within the tables and provide data to the reports. These databases are used in multi-tier software systems and are accessed by multiple pieces of software operated on both PCs and mobile devices by users across a company or university campus. Very often, the databases themselves will be maintained by one or more full-time database administrators (DBAs) who will monitor the security and performance of the systems, respond to any access problems and either design new databases as needed or provide expertise for the software designers.



```
/****** Script for SelectTopNRows command from SSMS  ******/
SELECT TOP 1000 [BusinessEntityID]
      ,[AccountNumber]
      ,[Name]
      ,[CreditRating]
      ,[PreferredVendorStatus]
      ,[ActiveFlag]
      ,[PurchasingWebServiceURL]
      ,[ModifiedDate]
  FROM [AdventureWorks2008].[Purchasing].[Vendor]
```

| | BusinessEntityID | AccountNumber | Name | CreditRating | PreferredVendorStatus |
|---|---|---|---|---|---|
| 1 | 1492 | AUSTRALI0001 | Australia Bike Retailer | 1 | 1 |
| 2 | 1494 | ALLENSON0001 | Allenson Cycles | 2 | 1 |
| 3 | 1496 | ADVANCED0001 | Advanced Bicycles | 1 | 1 |
| 4 | 1498 | TRIKES0001 | Trikes, Inc. | 2 | 1 |
| 5 | 1500 | MORGANB0001 | Morgan Bike Accessories | 1 | 1 |
| 6 | 1502 | CYCLING0001 | Cycling Master | 1 | 1 |
| 7 | 1504 | CHICAGO0002 | Chicago Rent-All | 2 | 1 |
| 8 | 1506 | GREENWOO0001 | Greenwood Athletic Company | 1 | 1 |
| 9 | 1508 | COMPETE0001 | Compete Enterprises, Inc | 1 | 1 |
| 10 | 1510 | INTERNAT0001 | International | 1 | 1 |
| 11 | 1512 | LIGHTSP0001 | Light Speed | 1 | 1 |

| | |
|---|---|
| Aggregate Status | |
| Connection failur | |
| Elapsed time | 00:00:00.741 |
| Finish time | 6/25/2013 3:44:39 AM |
| Name | COMEAU-DESKTOP |
| Rows returned | 104 |
| Start time | 6/25/2013 3:44:39 AM |
| State | Open |
| Connection | |
| Connection name | COMEAU-DESKTOP |
| Connection Details | |
| Connection elaps | 00:00:00.741 |
| Connection finish | 6/25/2013 3:44:39 AM |
| Connection rows | 104 |
| Connection start t | 6/25/2013 3:44:39 AM |
| Connection state | Open |
| Display name | COMEAU-DESKTOP |
| Login name | COMEAU-DESKTOP\ |
| Server name | COMEAU-DESKTOP |
| Server version | 11.0.2218 |
| Session Tracing II | |

**Name**
The name of the connection.

*Figure 1.6 - Server databases can host immense database systems that serve many hundreds or even thousands of users. This screenshot shows a view of a database maintained in SQL Server 2012.*

At the server level, the database software is no longer just a standalone program or a series of files but is actually integrated into the operating system of the network machine that it runs on. For example, Microsoft SQL Server can use Windows security to verify that a user is authorized to access specific databases. When a user logs into their Windows desktop machine and runs any program that accesses the database server, SQL Server will permit them access only to the databases for which their user account has been authorized by the person responsible for the system. This means that the user only has to enter a password once, at Windows startup, and the DBA can grant or revoke their access to specific resources with a couple clicks of the mouse. Another benefit is that database servers are often listed as network resources so that a person with enough knowledge can find the right database no matter which machine they're using on the network and without needing to know what machine it's stored on. Programmers can design applications to call a database over a company network or even the Internet so the average user can continue to be completely unaware of the database itself so long as everything is working correctly.

The two big players in the server database market are Oracle and Microsoft. As I've mentioned, Oracle has both their self-named Oracle database and MySQL while Microsoft has Microsoft SQL Server. IBM has its own database titles, DB2 and Informix. Organizations might use more than one title depending on their needs. I personally have Microsoft SQL Server Developer Edition and MySQL installed on my local machines for development and testing while the hosting service that I use for my website offers access to both SQL Server Express and MySQL for the creation of databases to support the website. Sometimes, tools like WordPress use specific database titles but it can also come down to the preference of the individual database developer or programmer. Cost can also be a factor. Depending on the needs of the organization and the applications being developed, database software might be a free download or it might cost an organization thousands of dollars in license and support fees.

# Chapter Summary

The flow of information has always been important and in today's electronic world, it's easier to store and manage than ever. Our everyday activities including purchases, communications, work and entertainment leave a trail of electronic data that is stored in various types of computerized databases. These databases can take many forms depending on the needs of the data and the organization managing it and might range from small databases on personal smartphones to massive networked databases maintained by corporations and government agencies. Many software titles are available to assist in the cataloging of data, each having their own limitations, features and appropriate use. Understanding how databases work can provide you with a better perspective of how information is stored and used and is essential in the pursuit of both technical and non-technical careers.

# For Further Study

## Review Questions

1. What is the basic definition of a database?
2. How does information stored in a Microsoft Word file or JPG image file differ from information stored in a database?
3. Other than data storage, what is a popular use for CSV and XML data?
4. What is an advantage of the XML format over the CSV format?
5. If you were transmitting continuous information over the Internet that needed to be collected and stored on the other end, what data format would you use?
6. How does a desktop database like OpenOffice Base differ from a server database like Microsoft SQL Server.
7. If you are designing a website, which two database software titles are you likely to be using?

## Exercises

1. Examine your own daily routine for examples of information that needs to be stored somewhere and ask yourself how this might be done. How long does the data need to be kept? How much data actually needs to be stored and how might the collection of data grow over time? How confidential is the data? What would the consequences be if the data is lost or mishandled? Consider the examples of databases in everyday life from the first part of this chapter as well.
2. Using some of the links provided, check out the websites for some of the software titles mentioned in this chapter. Search for information on the abilities and limitations of each software.

## Terms to Remember

**Back end** - In software development, the database is often referred to as the back end. The use of this term implies that the program code is independent of the database itself although a database can also include various types of programming instructions that are separate from the logic and presentation code in the other software components. See *Multi-tier Design*.

**CSV -** Comma Separated Values, a plain-text file format in which information is broken down into fields and stored in order to transfer the data between programs or simply store small to moderate amounts of information on disk. Also known as delimited or tabular data.

**Database** - A collection of information, usually in electronic form, stored in a specific format for easy retrieval and use in reports and other applications.

**Front end** - In software development, the front end represents any logic and presentation code such as user forms and interfaces, reports and business logic that exists separately from the database. See *Multi-tier Design*.

**Gigabyte (GB)** - One billion characters of information. More precisely, $1024^3$ bytes (1024 x 1024 x 1024 or 1,073,741,824 bytes).

**JSON** - JavaScript Object Notation, a text file format that uses a format based on the JavaScript language to store complex data.

**Multi-tier Design** - Software design in which the program code is developed and exists independently from the database and the two communicate with each other.

**Open Source** - A software development practice under which source code is made freely available for study and modification by independent developers. This means that the software is often free as well although more advanced versions and product support might be provided as a commercial product.

**Petabyte (PB)** - One million gigabytes of information ($1024^5$ bytes).

**XML** - Extensible Markup Language, a plain-text format which uses user-defined markup tags to store structured data for transfer between systems or over the Internet.

**Zettabyte (ZB)** - One billion gigabytes of information ($1024^6$ bytes).

## Links

JSON text format official site
http://www.json.org

MySQL Database official site
http://www.mysql.com

OpenOffice Base
http://www.openoffice.org/product/base.html

SQLite home page
http://www.sqlite.org

Oracle Express 11g home page
http://www.oracle.com/technetwork/products/express-edition/overview/index.html

EMC - World Data Doubling Every Two Years
http://www.emc.com/about/news/press/2011/20110628-01.htm

If you've enjoyed this free preview, you can purchase the rest of the book online on the following sites …

**Amazon.com**

http://www.amazon.com/First-Guide-Database-Design-ebook/dp/B00FNAMMA8

**Barnes and Noble**

http://www.barnesandnoble.com/w/your-first-guide-to-database-design-andrew-comeau/1117049383?ean=2940148791621

**Scribd.com (PDF format)**

http://www.scribd.com/doc/174069216/Your-First-Guide-to-Database-Design

**Visit the book's official page at:**

http://www.andrewcomeau.com/Pages/db-design-guide.aspx