# How to Use Relational Databases: Data Retrieval with Structured Query Language



By Diane Dolezel, MSCS, RHIA

The ability to generate timely, accurate reports is essential to quality management and clinical documentation improvement (CDI) efforts.[1,2] Health information management (HIM) professionals' knowledge of health record content is vital to CDI efforts to ensure the availability of accurate and detailed documentation for treatment, coding, and reimbursement.

As data governance facilitators, the e-HIM professionals' job functions include creating and assessing database reports.[3] However, many e-HIM professionals do not have sufficient training in database reporting. This article explains basic database concepts, then provides step-by-step directions for creating a standard query language (SQL) report.

## Importance of Database Reporting

Reports summarize and present data for easier visualization.[4] Today's Big Data revolution is a driving force for database reporting.[5] Typical facility data repositories contain clinical trial, performance measurement, reimbursement, health services, health information exchange, vital statistics, and health record data generated from clinical, administrative, and financial systems.[6] Data governance activities, such as reconciliation of electronically transmitted data discrepancies, require report creation.[7]

Queries are created to retrieve and display data stored in database reports. The Centers for Medicare and Medicaid Services (CMS) requirements for CDI necessitate retrieving and examining present on admission, diagnosis related group, and case management data for compliance.[8] Payer mixer reports enable case mix management to facilitate reimbursement.[9] Within the HIM arena, routine queries generate incomplete chart, remittance, coding accuracy, and case mix reports.

## Relational Database Concepts

Microsoft Access is a relational database that improves data quality, access, and sharing while providing a rich toolset for creating, managing, and querying data.[10] A relational database is a collection of data organized in tables that contain data for only one entity, such as a person, place, or event. Each table has columns, or fields, which define the attributes (i.e., characteristics) of the entity, and rows that record the attribute data. Tables have primary keys that uniquely identify each row.[11] For example, because the medical record number (MRN) column is the primary key for the patient table, duplicate MRNs cannot be added to the table, which is a common registration error.[12]

Tables are joined by primary and foreign key relationships to enforce database constraints, and reduce data inconsistencies. When the MRN field in the patient table is related to the MRN field in the visit table, by creating a relationship in Access software, a primary-foreign key relationship exists.[13] Once created, a primary-foreign key relationship facilitates reporting, reduces data redundancy, and prevents accidental data deletion of linked data. Data inconsistencies, resulting from data being edited in multiple places, are prevented by the primary and foreign key relationships.

## SQL Concepts

The SQL programming language for relational database was developed in the 1970s at IBM.[14] SQL commands are used to define a table, manipulate data, or control data access. The basic syntax for an SQL Select command, used to retrieve and display table data, is:[15,16]

```
Select [columname1, columname2, etc.]
From [tablename1, tablename2, etc.]
```

Where [condition]
[other optional conditions];
E.g. Select MRN, lname, fname, age from Patient where lname =”Smith”, age > 20, and zipcode=”1234”
displays all patients over 20 years with last name Smith and zipcode of 1234.

Complex queries can be created from several tables by referencing the linked columns in the primary and foreign key relationship, and data can be grouped and ordered.

## Reporting Skills Not Optional

Database reporting is a now a mandatory skill for e-HIM professionals, especially with the emerging fields of clinical documentation improvement, information governance, and data analytics. HIM professionals must be able to use SQL to define database tables, create reports, and monitor ongoing processes in order to function as the gatekeepers of health record data.

## Steps for SQL Reporting

Suppose you are an e-HIM manager who needs to report on patient and visit data. The steps for creating an SQL report are:

1. Install Microsoft Access
2. Create the database
3. Link tables
4. Create a report using SQL

### Install Microsoft Access Reporting Tool

Microsoft Access software is available in specific Microsoft Office software packages.[17] The process to obtain and install Access is:

1. Open a web browser and navigate to this Microsoft website: www.microsoftstore.com/store/msusa/en_US/cat/categoryID.69403900?s_kwcid=AL!4249!10!26448943831840244259358ef_id=VVOqXwAAANfb9jdJ:20150825211647:s.
2. Examine the details for Office 365 Home, Personal, and University Versions to determine which version is the best fit. For example, the University version requires validation from a university e-mail, and the Personal version only works on one tablet or PC.
3. Download and install a copy of Office 365 that includes Access on your MS Windows system.

### Create the database

Initially, a new database must be created, and tables must be defined and loaded with data. The process will be illustrated with pictures and data created on the computer desktop of the author of this article for illustration.

1. Start Access and click on the Blank Desktop Database icon to create a new database. In the right pane, under “Blank Database Desktop,” change the default file name to “Reporting Database.” (See Figure 1.)
2. Click “Create.” Access creates a database, an empty table in Datasheet view.
3. Click on the Table 1 tab. Change Table Name to “Patient.” (See Figure 2.)
4. Select the Patient, click the right mouse button, and Select “Design View.”
5. Enter the Patient Fields listed in Figure 3.
6. Return to the Datasheet View.
7. Type in the Patient data, as illustrated in Figure 4.
8. Close the table.
9. In Design View, create a second table named “Visit” with the column (field) names shown in Figure 5.
10. Add the data shown in Figure 6 to the Visit table. Note: Visit.MRN must be a number datatype because it will be linked to Patient.MRN, which is a number.

**Link tables**

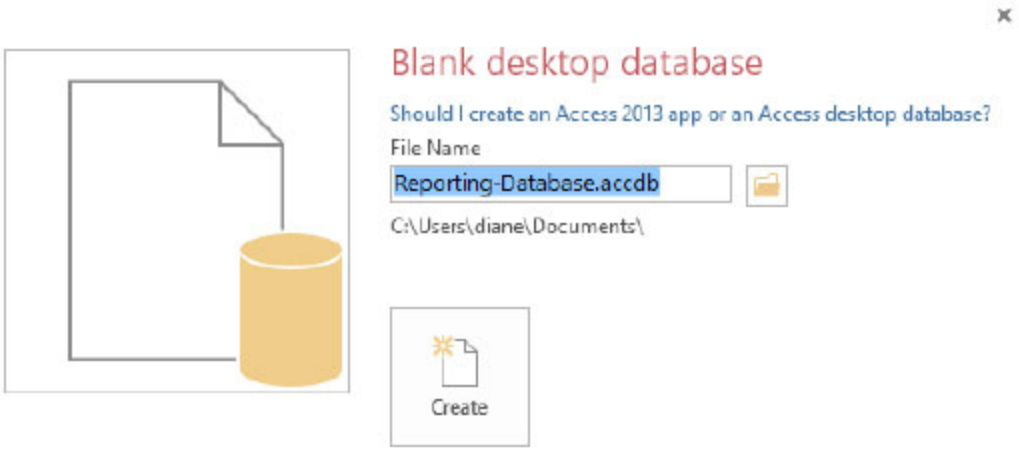
1. Select tab Database Tools > Relationships. Right mouse click and select SHOW TABLES.
2. From the pop-up, select table Patient > Click Add. Repeat to add Visit table.
3. Select the Patient.PTMRN and drag and drop it onto the Visit.MRN. (See Figure 7.)
4. In the pop-up, select Enforce Referential Integrity. Click "Create."
5. A line appears linking Patient.PTMRN and Visit.MRN in a primary and foreign key relationship.

**Create a report using SQL**

To create an Access report, the user can utilize the Query Wizard or type the SQL commands into the interface. It is recommended that the wizard be used initially to join the tables and generate the basic query, which can be modified.

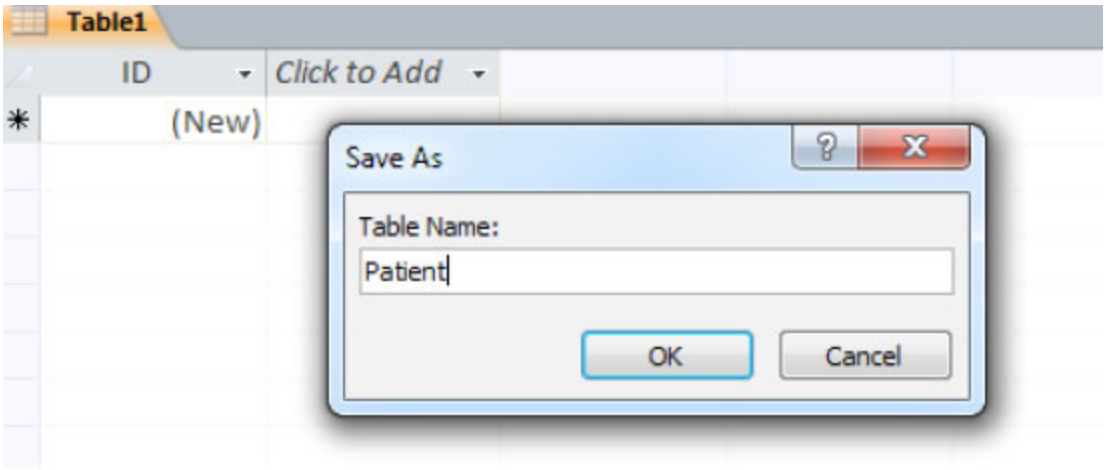
1. To use the wizard, select Create Tab > Query Wizard. Select "Simple Query."
2. From the Tables/Queries dropdown select Patient table, then select the available fields by clicking ">>." Repeat with the Visit table, adding all the fields. (See Figure 8.)
3. Click next and accept all defaults. Click "FINISH."
4. Double click on "Patient Query;" it will run and display the results in datasheet view, as shown in Figure 9.
5. Right mouse click on the Patient Query tab, select SQL View to display user's SQL. This can be edited manually to add a zipcode and run by selection the "!" symbol, as shown in Figure 10.
6. Return to design view, drag and drop Patient.PTZipcode to the field list, and it will be added to the query. This method, called Query by Example, allows the user to add multiple select criteria to the query, as shown in Figure 11.

## Figure 1: Create Blank Database




---

## Figure 2: Create Patient Table




## Figure 3: Patient Fields

| Field Name | Field Size |
|---|---|
| PTMRN | AutoNumber, Primary Key |
| PTFNAME | Text |
| PTLNAME | Text |
| PTMI | Text |
| PTSTREET | Text |
| PTCITY | Text |
| PTSTATE | Text |
| PTZIPCODE | Text |
| PTPHONENUM | Text |

## Figure 4: Patient Data

| PTFNAME | PTLNAME | PTMI | PTSTREET | PTCITY | PTSTATE | PTZIPCODE | PTPHONENUM |
|---|---|---|---|---|---|---|---|

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Jane | Allen | A. | 12 First St. | San Marcos | TX | 78666 | 5122458397 |
| John | Brown | B. | 22 Congress | San Marcos | TX | 78666 | 5122457654 |
| Sue | Cotton | C. | 32 Riverview | Austin | TX | 73301 | 361222444 |
| Kim | Daniel | D. | 44 South St | New Braunsfel | TX | 78130 | 2101231122 |
| Danette | Edwards | E. | 55 Marshall Rd | New Braunsfel | TX | 78132 | 2106250365 |
| Susan | Frank | F. | 62 Congress | Austin | TX | 78789 | 5124713434 |
| Tom | Green | G. | 72 Riverview | San Marcos | TX | 78667 | 5123962220 |
| Sam | Harrrington | H. | 802 South St | San Marcos | TX | 78667 | 5123961212 |
| Diane | Isles | I. | 901 South St | San Marcos | TX | 78667 | 5123964555 |
| Ronald | Jones | J. | 2222 Congress | Austin | TX | 78789 | 5124441234 |

**Figure 5: Visit Fields**

| **Field Name** | **Field Size** |
|---|---|
| VISIT_ID | AutoNumber, Primary Key |
| VISIT_DATE | Text |
| VISIT_REASON | Text |
| MRN | Number |

**Figure 6: Visit Data**

| VisitDate | VisitReason | MRN |
|---|---|---|
| 15-Oct-10 | Check up | 1 |
| 15-Oct-10 | X-ray | 2 |
| 16-Feb-11 | Flu | 3 |
| 24-Feb-11 | Allergies | 4 |
| 10-Feb-11 | Cough | 5 |
| 17-Feb-11 | Cold | 6 |
| 15-Feb-11 | Fever | 7 |
| 15-Feb-11 | Nausea | 8 |
| 09-Feb-11 | Headache | 9 |

**Figure 7: Linked Tables**

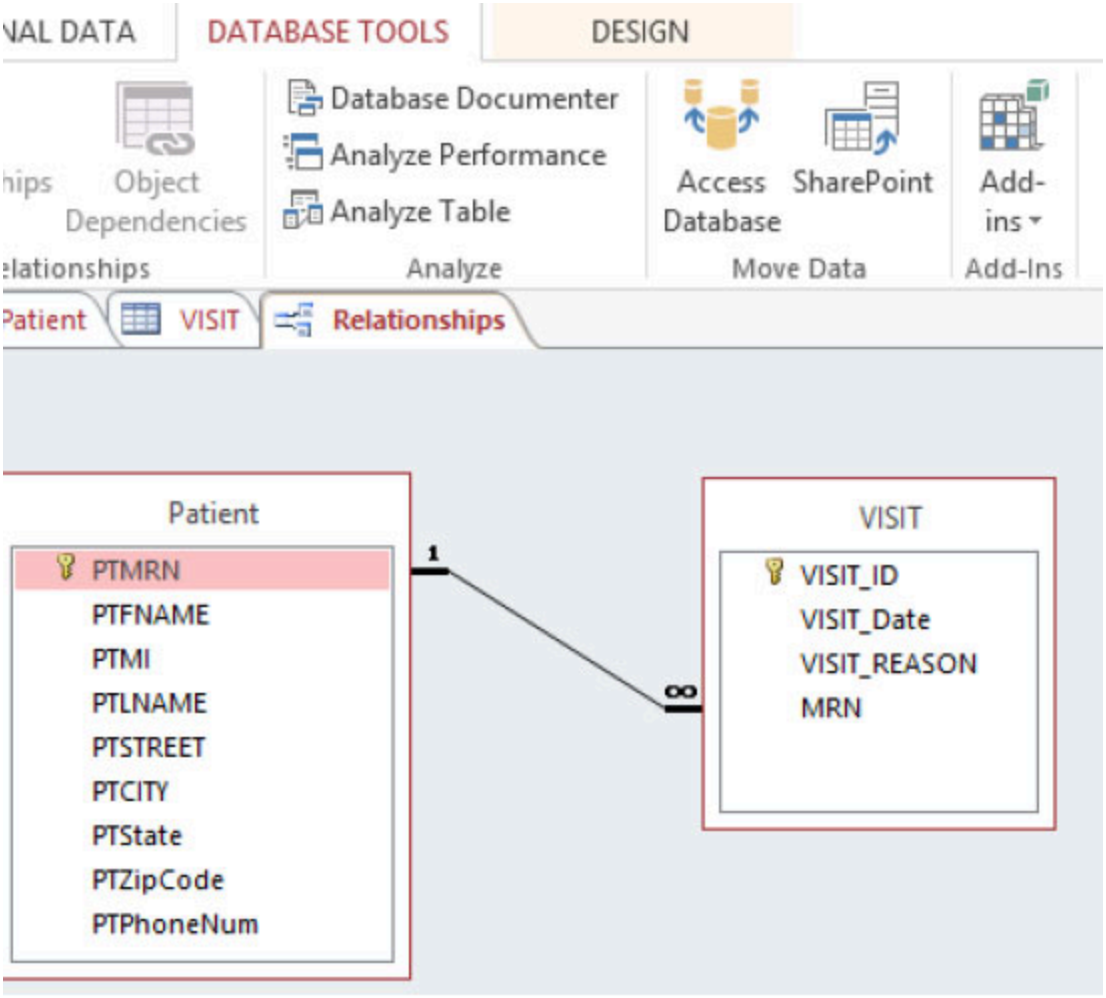
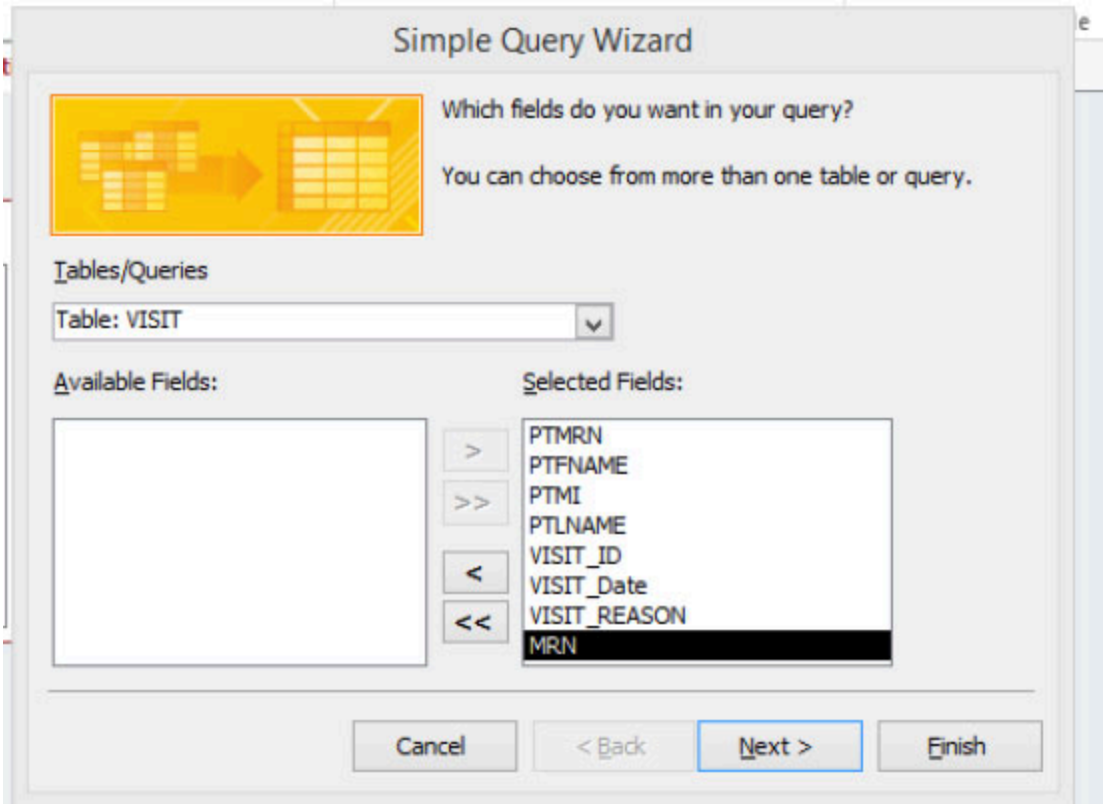
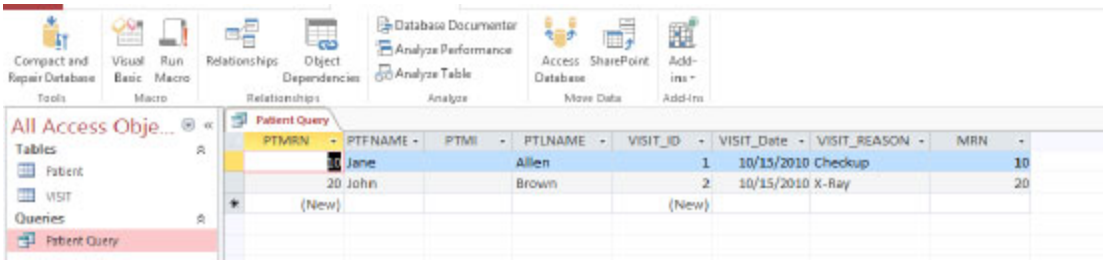


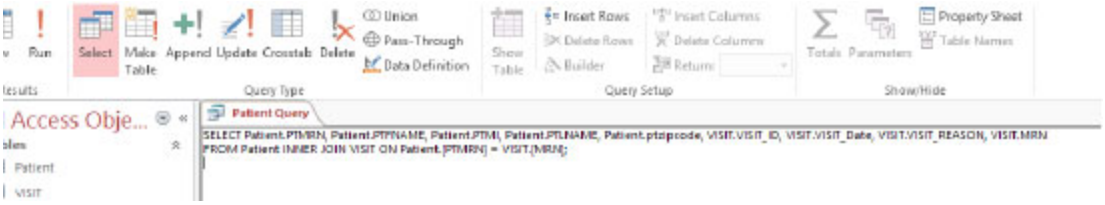

**Figure 8: Simple Query**



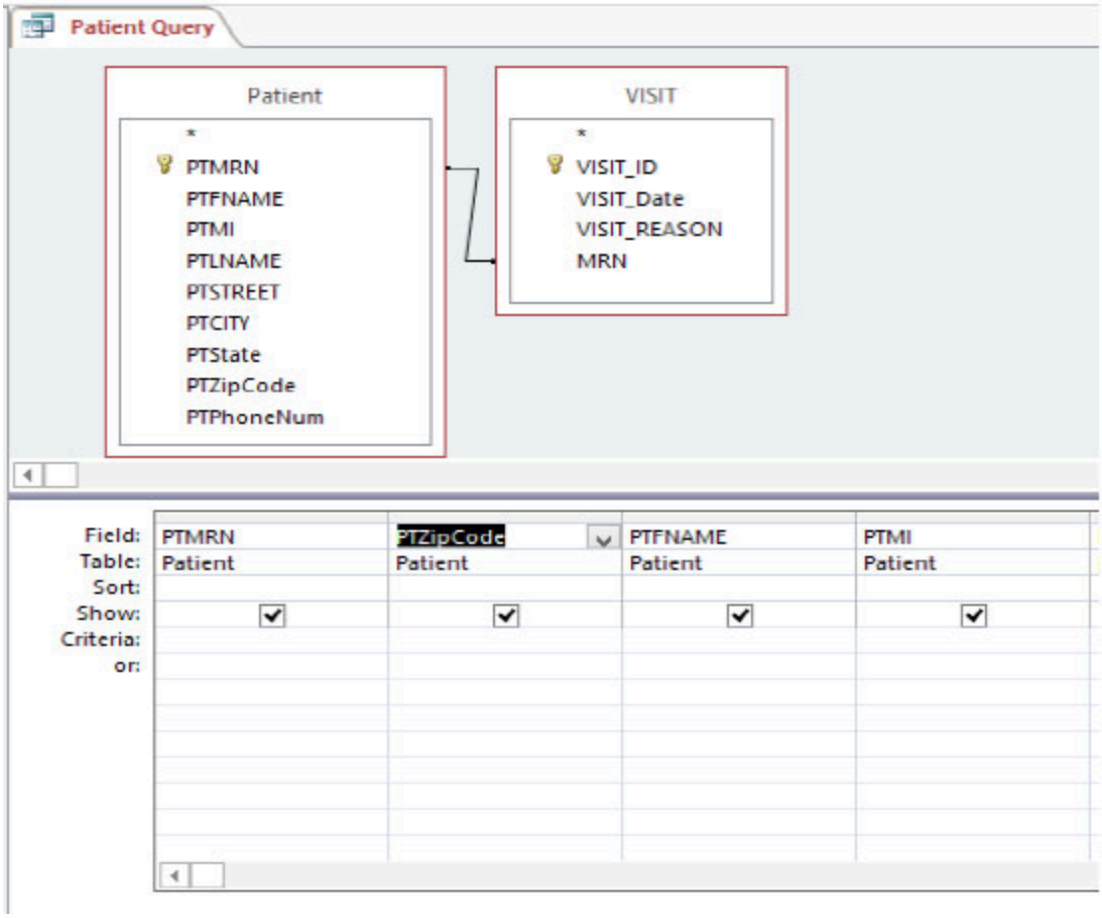
## Figure 9: Run Query




## Figure 10: Add Zipcode




## Figure 11: Query by Example




## Notes

1. AHIMA. "EHRs as the Business and Legal Records of Healthcare Organizations (Updated). Appendix A: Issues in Electronic Health Record Management." Updated November 2010.
2. AHIMA. "Clinical Documentation Improvement Toolkit." 2014.
3. AHIMA. "HIM Functions in Healthcare Quality and Patient Safety. Appendix C: HIM's Role in Data Capture, Validation, and Maintenance." *Journal of AHIMA* 82, no. 8 (Aug 2011): expanded online version.
4. Microsoft. "Database basics." https://support.office.com/en-za/article/Database-basics-a849ac16-07c7-4a31-9948-3c8c94a7c204.
5. Fernandes, Lorraine, Michelle O'Connor, and Victoria Weaver. "Big Data, Bigger Outcomes." *Journal of AHIMA* 83, no. 10 (October 2012): 38-43.
6. LaTour, K., S. E. Maki, and P.K. Oachs. *Health Information Management Concepts, Principles, and Practice* (4th edition). Chicago, IL: AHIMA Press, 2012.
7. AHIMA. "EHRs as the Business and Legal Records of Healthcare Organizations (Updated). Appendix A: Issues in Electronic Health Record Management."
8. Centers for Medicare and Medicaid Services. "Pay-for-Performance and Quality Incentives." 2005. www.cms.gov/site-search/search-results.html?q=Pay-for-Performance%20and%20Quality%20Incentives.
9. Fahrenholz, Cheryl Gregg. "Show Me the Money... A Look at the Revenue Cycle from the Billing Perspective." 2010 AHIMA Convention Proceedings, September 2010.
10. White, Susan. *Practical Approach to Analyzing Healthcare Data* (2nd edition). Chicago, IL: AHIMA, 2013.
11. Ibid.
12. Dimick, Chris. "Exposing Double Identity at Patient Registration." *Journal of AHIMA* 80, no. 11 (November 2009): web extra.
13. Coronel, Carlos, Steven Morris, and Peter Rob. *Database Systems: Design, Implementation and Management* (9th edition). Sanford, CT: Cengage Learning, 2015.
14. Chamberlin, D. D. "Early History of SQL." *IEEE Annals of the History of Computing* 34, no. 4 (2012): 78-82. http://doi.org/10.1109/MAHC.2012.61.
15. ITBusinessEdge. "SQLCourse - Lesson 3: Selecting Data." 2015. www.sqlcourse.com/select.html.
16. McLendon, Kelly. "Technology Basics for Today's Health Information Professionals." 2010 AHIMA Convention Proceedings, September 2010.
17. Microsoft. "Database basics."


Diane Dolezel (dd30@txstate.edu) is an assistant professor in the health information management department at Texas State University in San Marcos, TX.

**Article citation**:
Dolezel, Diane. "How to Use Relational Databases: Data Retrieval with Structured Query Language" *Journal of AHIMA* 86, no.11 (November 2015): 22-27.