Databases and SQL

Databases – What are they and why are they important?

A database is defined as an organized collection of information.

Why use databases?

- Centralizes systems (keep data in one location)
- Keep track of products, customers, testing, etc.
- Easy to access information via the network

Databases Vs. Spreadsheets

Use databases when you are trying to organize various data of a certain type (such as date, number, text, etc.).

Spreadsheets will do the same thing, but spreadsheets are typically used when you want to perform calculations and analyze numerical data.

Databases allow you to restrict the data that the user enters, such as putting a date in the format of MM-DD-YYYY. It also allows the user to link related data across multiple tables. Another advantage is that it can restrict access so data remains confidential.

Microsoft Access and Other Databases

Microsoft Access is the most used database desktop application in Windows. It saves files as .accdb or .mdb (old extensions saved as .mde).

It allows communication using Structured Query Language (SQL). SQL is the most common method to programmatically writing data to a database.

We’ll use Access in this course since it’s readily available with the Microsoft Office package.

The most commonly used databases in order include Oracle, MySQL, Microsoft SQL Server, PostgreSQL, MongoDB, IBM DB2, and Access.

Structured Query Language (SQL)

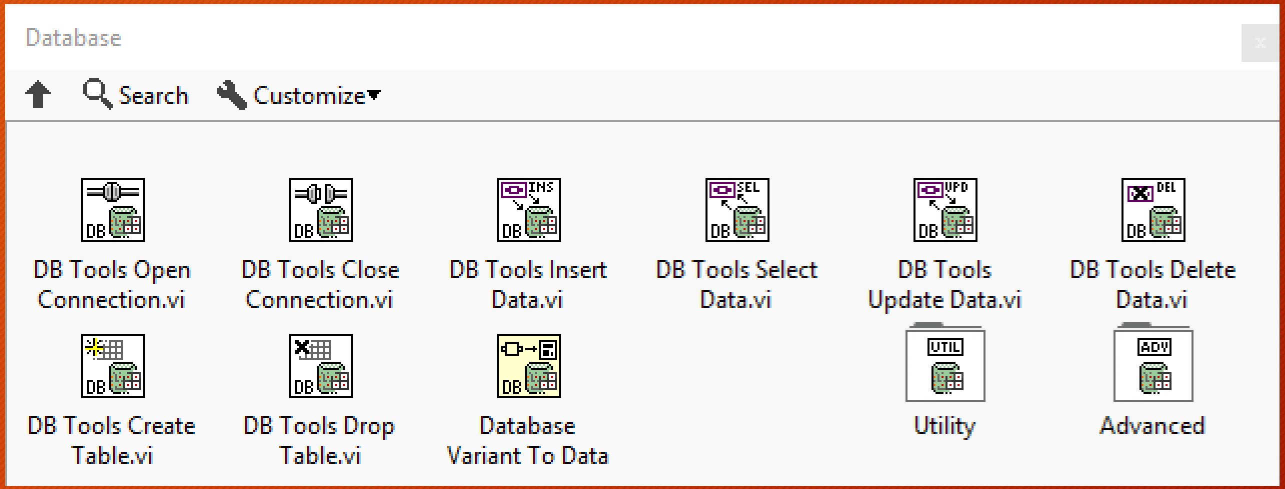
SQL is a language that allows you to work with a database. You can insert records, update records, and delete records. You can also create tables which are collections of related data within the database. Each table is kind of like a new tab in Excel.

SQL is an ANSI standard, but there are many different versions of SQL used by different database vendors. To be in compliance with ANSI, these vendors must allow major commands such as Delete, Insert, Update, Where, etc.

LabVIEW and DBTools

DBTools is in the Connectivity > Database palette in LabVIEW that has the tools needed to communicate with databases. The functions have the SQL commands built into them so you don't have to learn the SQL language.

DBTools is automatically provided in the LabVIEW 32-bit Professional edition. It is not available for 64-bit editions of LabVIEW.



UDL Files

NI’s Database Connectivity Toolkit requires a Microsoft Data Link file (.udl) in order to interface with a Microsoft Access database.

UDL files are created by transforming a text file to a udl file. To make a .udl file, in Windows Explorer click New > Text File and then replace the extension to .udl (instead of .txt).

Database extensions: .accdb & .mdb

Databases for Microsoft Access can use both .accdb and .mdb. Originally, I used .accdb, but the latest articles by NI seem to use .mdb.

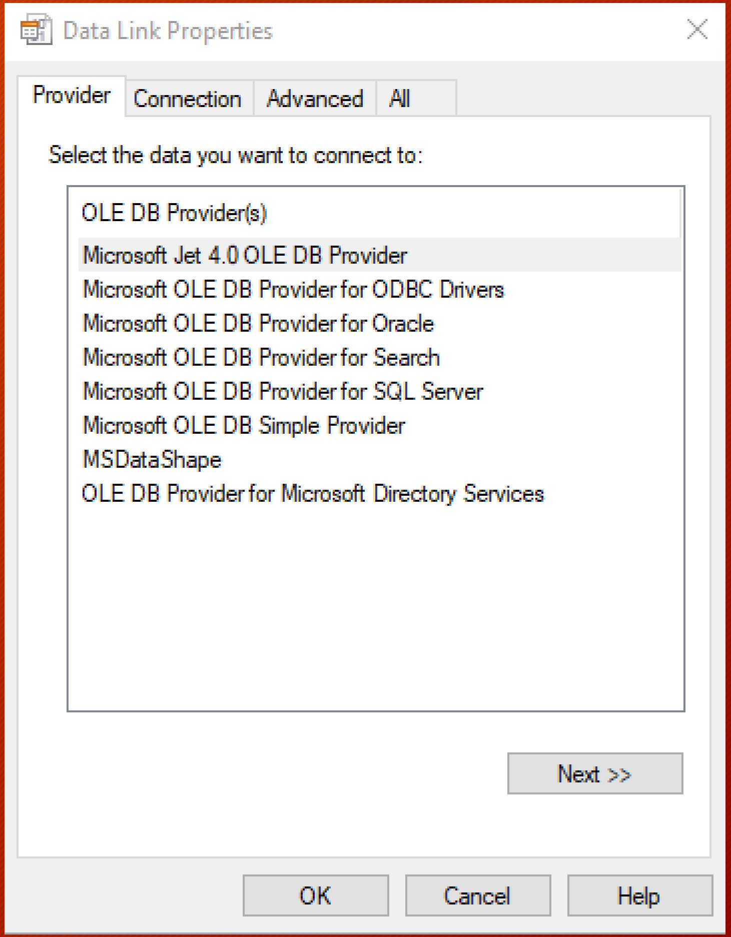
You may have to try both extensions to be able to communicate with a database.

Linking the .udl File With a .mdb File

To make things easier, be sure the .udl file is in the same file path as your Microsoft Access page. Let’s say that the database is called Brad.mdb and the udl file is called Brad.udl and they are both saved in the C: directory *(note that it’s better to use a different directory)*. The .udl file must NOT contain spaces.

~~On a 32 bit operating system *(not typical)*, double-click on the .udl file. Go to the Provider page and select “Microsoft Jet 4.0 OLE DB Provider” or similar option. Click Next.~~

On a 64 bit operating system, you won’t see the Access Ole DB Provider so you have to do some tricks. Go to windows search and type “CMD” to open up command prompt. Copy and paste everything in the following that is green *(replace the path for the UDL file with the correct path)*: C:\Windows\syswow64\rundll32.exe “C:\Program Files (x86)\Common Files\System\Ole DB\oledb32.dll”, OpenDSLFile C:\Brad.udl. A window should pop up and you should see the Access OLE DB Provider under the Provider tab.



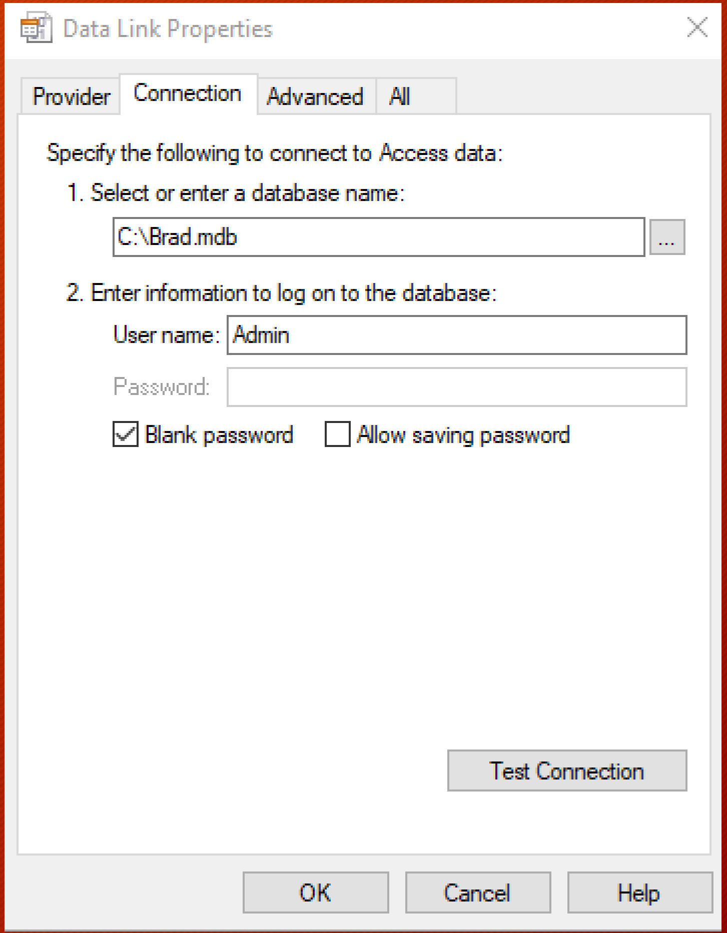
Select the data you want to connect to:

OLE DB Provider(s)
Microsoft Jet 4.0 OLE DB Provider
Microsoft OLE DB Provider for ODBC Drivers
Microsoft OLE DB Provider for Oracle
Microsoft OLE DB Provider for Search
Microsoft OLE DB Provider for SQL Server
Microsoft OLE DB Simple Provider
MSDataShape
OLE DB Provider for Microsoft Directory Services

Linking the .udl File

This brings you to the Connection tab. Click on the "…" to choose the database C:\Brad.mdb.

Click "Test Connection" and hopefully it says your connection has succeeded. If so, then you are ready to link LabVIEW to the database.



Specify the following to connect to Access data:

1. Select or enter a database name: C:\Brad.mdb

2. Enter information to log on to the database:

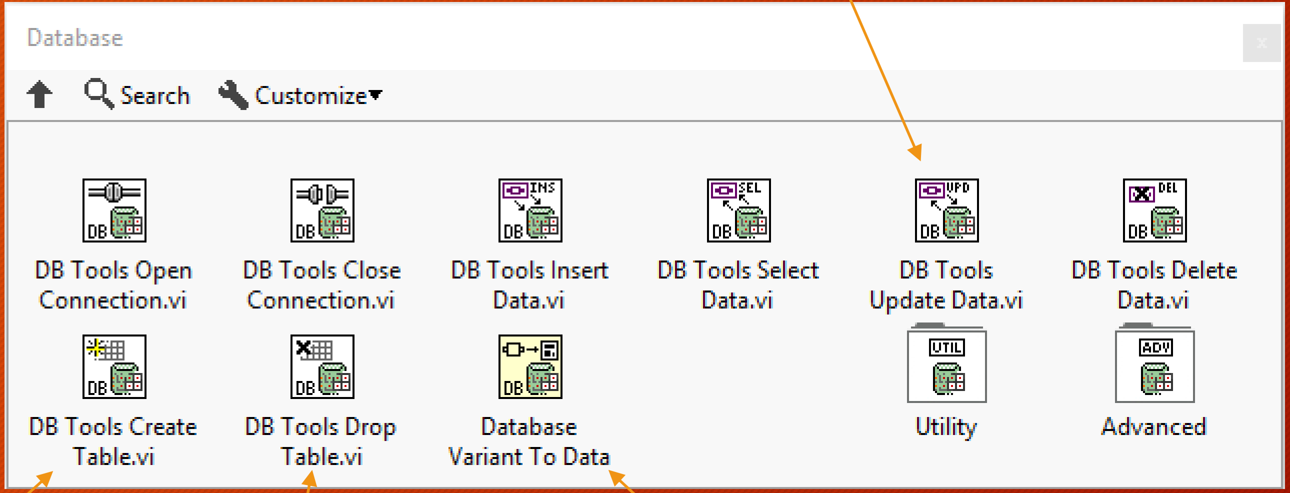
User name: Admin

Password:

☑ Blank password □ Allow saving password

LabVIEW Database Palette

Allows programmer to change data.

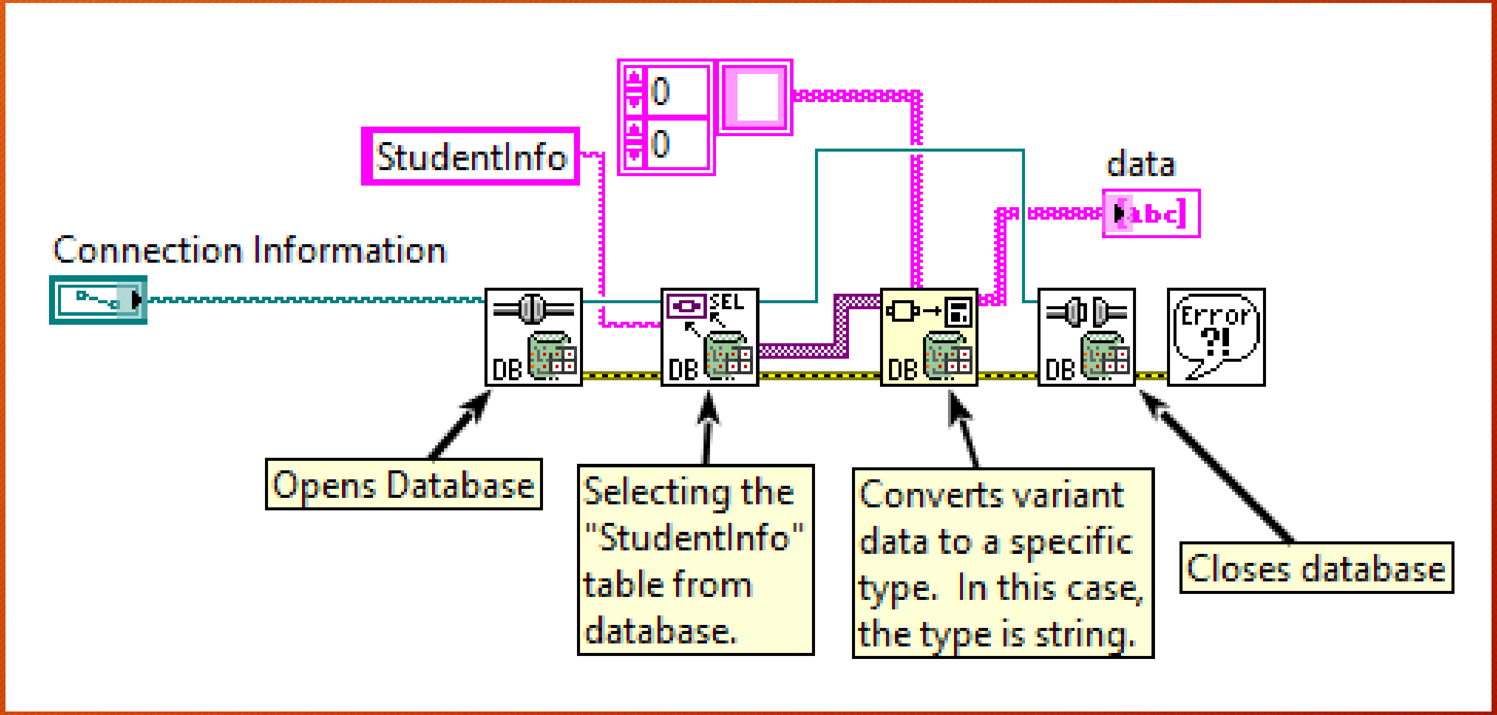


Tables are kind of like tabs in Excel. Creates a different page of information.

Deletes table

Data comes in as a variant. This allows the programmer to cast a type

Example: Reading Data from Database



Opens Database

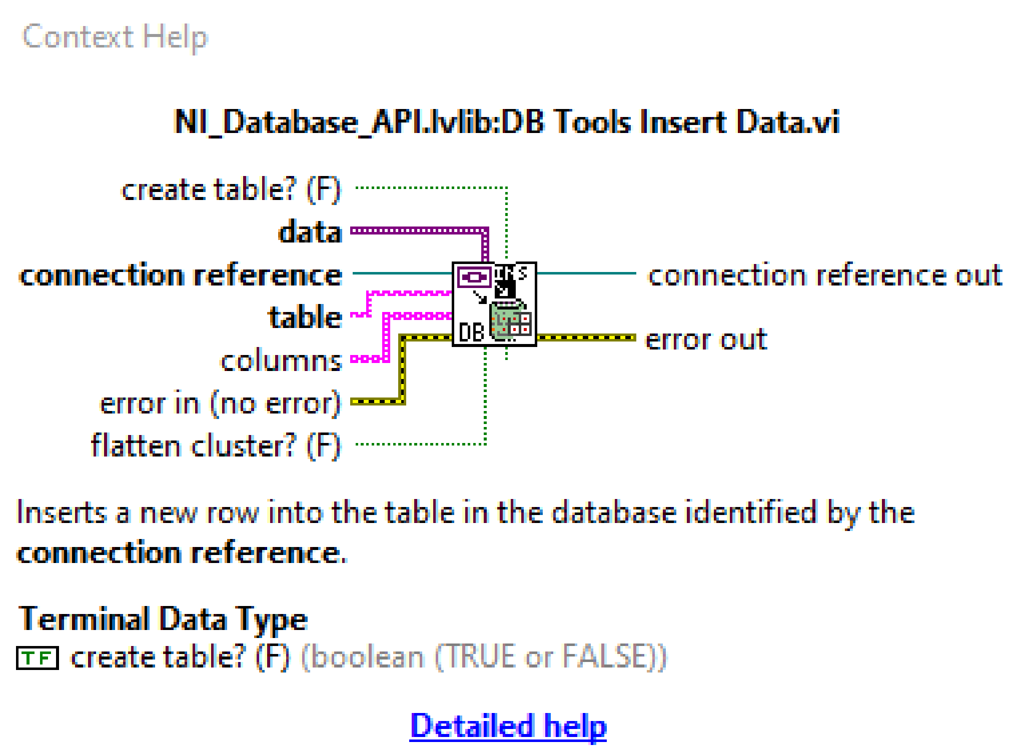
Selecting the "StudentInfo" table from database.

Converts variant data to a specific type. In this case, the type is string.

Closes database

Writing to databases

- To write to a database, you need data, the connection reference, the table name, and column names.
- When writing data to a database, you must use a cluster and the elements within the cluster must match the data type of the database. In Access, the first column in each table of a database has an ID number. When you write to a database, you need to skip this column.
- The table is just the name of the table you are using in the database. There can be lots of different tables in each database.
- Columns is the name of each column you are writing data to. Don't include the name of the ID column (first column in each table).



NI_Database_API.lvlib:DB Tools Insert Data.vi

Inserts a new row into the table in the database identified by the **connection reference**.

Terminal Data Type

TF create table? (F) (boolean (TRUE or FALSE))