

<https://www.halvorsen.blog>

# LabVIEW SQL Toolkit

Hans-Petter Halvorsen



# LabVIEW SQL Toolkit

- LabVIEW Database Connectivity Toolkit is included with LabVIEW professional, but it is cumbersome to use.
- It is recommended that you use the LabVIEW SQL Toolkit instead. This Toolkit is very simple to use.
- With LabVIEW SQL Toolkit you can communicate with SQL Databases from LabVIEW
- Download and use it for free from my Blog
- <https://www.halvorsen.blog/>
- [https://www.halvorsen.blog/documents/technology/database/database\\_labview.php](https://www.halvorsen.blog/documents/technology/database/database_labview.php)
- Follow the installation instructions in the ReadMe.txt file inside the Zip file.

# LabVIEW SQL Toolkit

Blog

Database Communication in LabVIEW

## Database Communication in LabVIEW

Hans-Petter Halvorsen

### What is LabVIEW?

LabVIEW is a graphical programming language, and it has powerful features for simulation, control and DAQ applications.

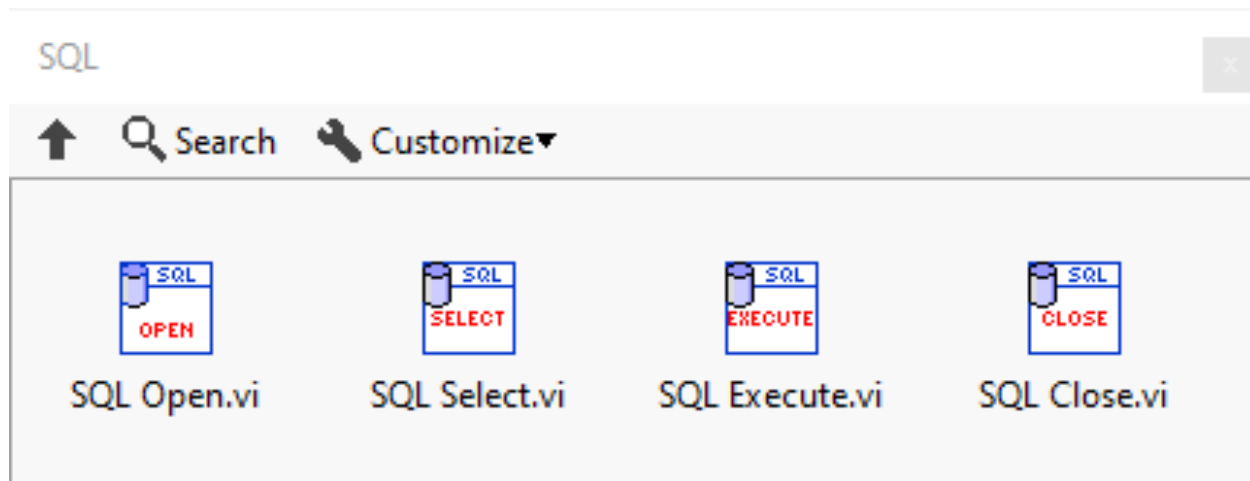
You can also use LabVIEW to communicate with databases.

Database Communication in LabVIEW:

[https://www.halvorsen.blog/documents/technology/database/database\\_labview.php](https://www.halvorsen.blog/documents/technology/database/database_labview.php)

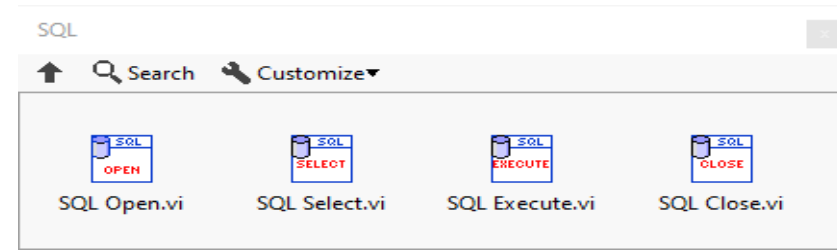


# LabVIEW SQL Toolkit



# Installation

- Copy "SQLToolkit.mnu" to the folder:  
C:\Program Files (x86)\National Instruments\LabVIEW 20xx\menus\Categories\
- Copy "SQLToolkit.llb" to the folder:  
C:\Program Files (x86)\National Instruments\LabVIEW 20xx\vi.lib\
- In the Functions palette in LabVIEW a new palette named "SQL" will appear.

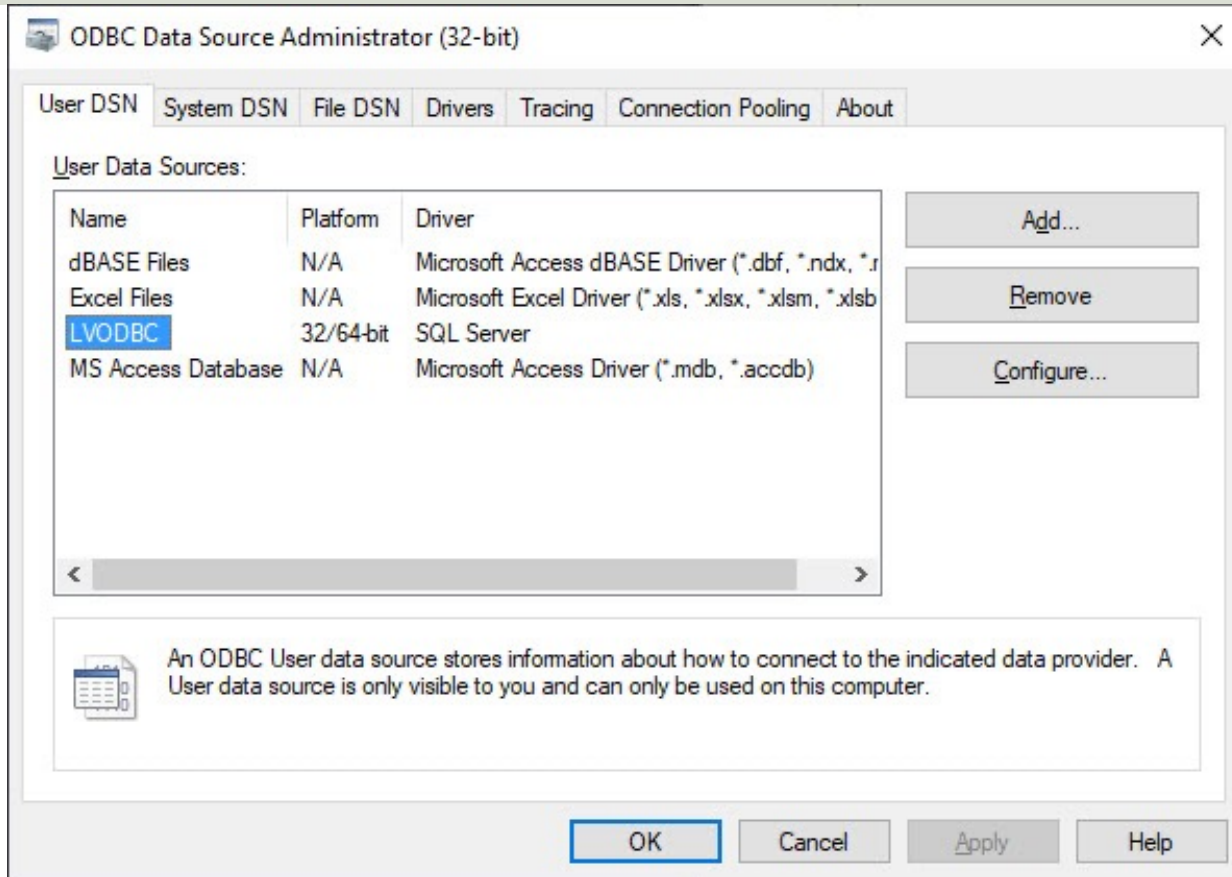


Note! LabVIEW 32bit: Program Files (x86)\  
LabVIEW 64bit: Program Files\

# ODBC

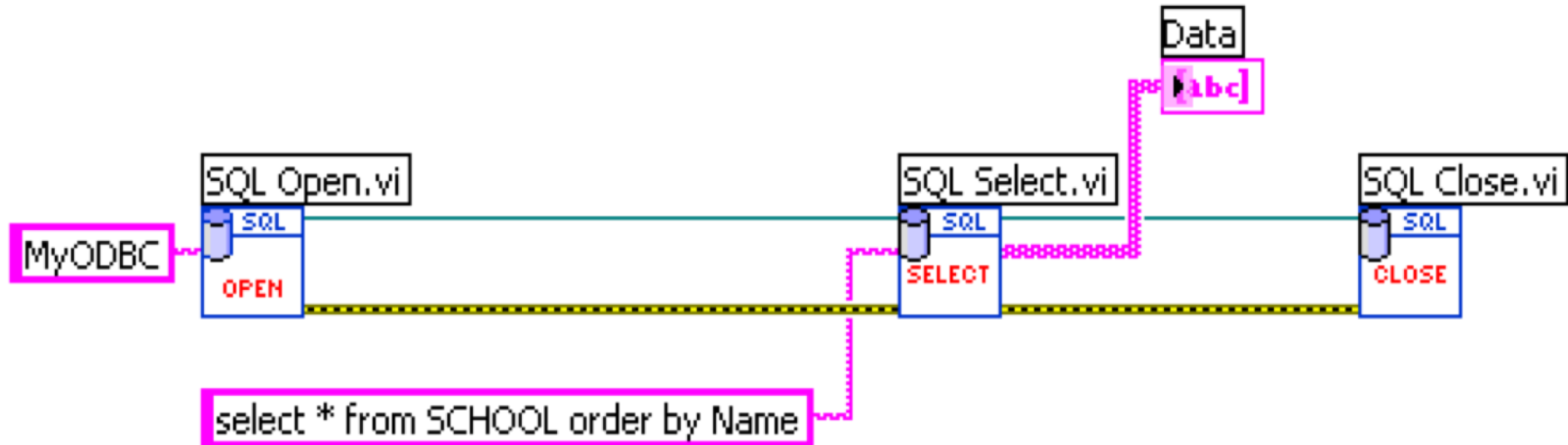
- ODBC (Open Database Connectivity) is a standardized interface (API) for accessing the database from a client.
- You can use this standard to communicate with databases from different vendors, such as Oracle, SQL Server, etc.
- The designers of ODBC aimed to make it independent of programming languages, database systems, and operating systems.
- In Windows 10 you find the "ODBC Data Source Administrator" tool here:  
Control Panel → Administrative Tools → Data Sources (ODBC).
- Then click the "Add..." button in order to create an ODBC connection to your database

# ODBC



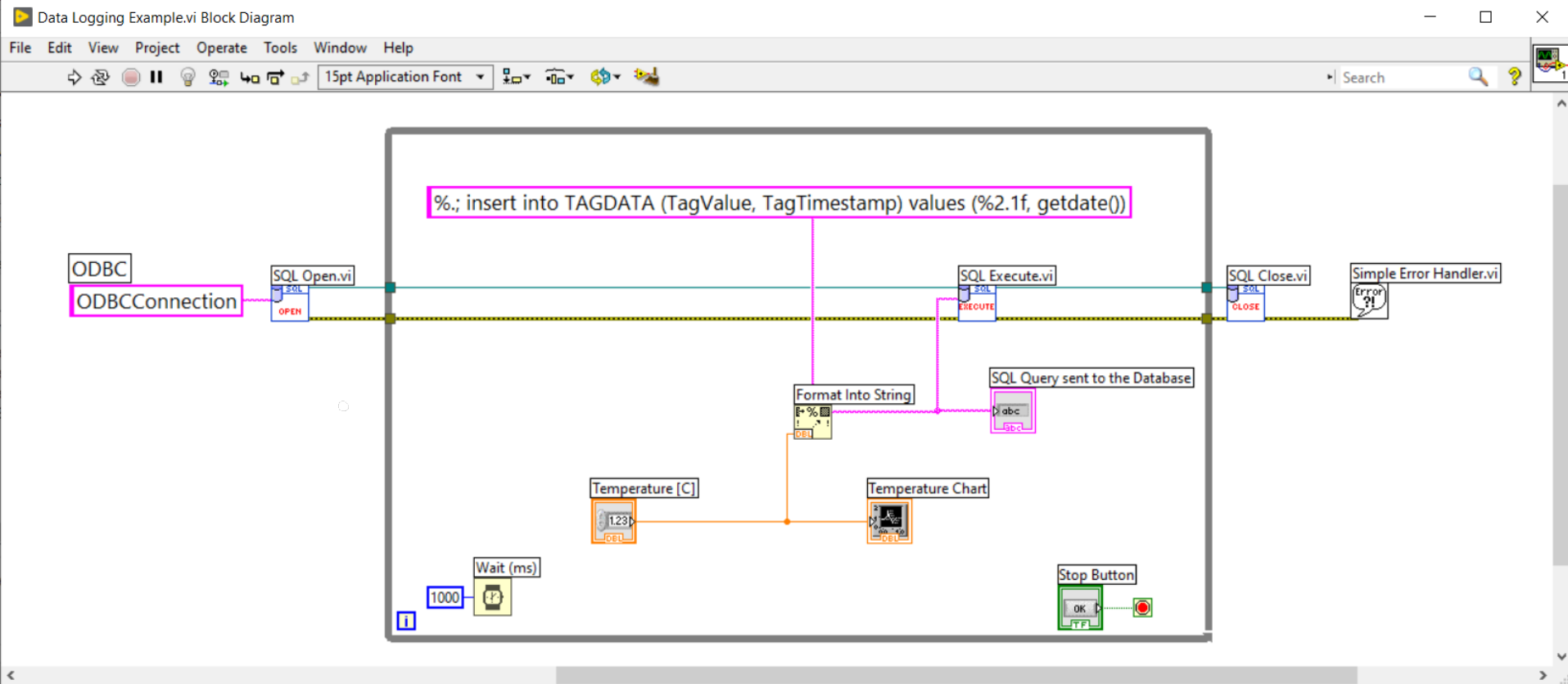
Note! Make sure to use the 32bit version of the ODBC Tool!

# SQL Select Example





# SQL Insert Example



# Comments to Code Example

- In this example the SQL Server function `getdate()` is used.
  - This function just makes sure that the current timestamp is inserted into the "TagTimeStamp" database column.
- Decimal Numbers in SQL: Using Decimal Numbers in SQL can cause problems. Assume we want to insert the Value "22,5":

```
insert into TAGDATA (TagValue, TagTimestamp) values (22,5, getdate())
```
- Here, SQL assumes you try to insert 3 values into 2 columns since SQL uses "," as a separation symbol, and you will get an error running this statement.
- If we instead write it like this:

```
insert into TAGDATA (TagValue, TagTimestamp) values (22.5, getdate())
```
- Then it will work without problems.
- Format Into String:
- Using `%.` in front of the string means that "." will be used as Decimal Point. Then we will avoid the decimal problem illustrated above.
- `%2.1f` means that this is replaced with the value that comes from the Temperature control with one decimal value. "f" means it is a floating-point value

# Hans-Petter Halvorsen

University of South-Eastern Norway

[www.usn.no](http://www.usn.no)

E-mail: [hans.p.halvorsen@usn.no](mailto:hans.p.halvorsen@usn.no)

Web: <https://www.halvorsen.blog>

