

Easysoft ODBC- QOnline Driver User's Guide

This manual documents version 1.0.n of the Easysoft ODBC-QOnline Driver.

Copyright © 1993-2025 Easysoft Limited.

All rights reserved.

You may not reverse engineer, decompile, or disassemble this manual. Information in this document is subject to change without notice. Companies, names, and data used in examples are fictitious unless otherwise noted.

The names of companies referred to herein, their corporate logos, the names of their hardware and software may be trade names, trademarks or registered trademarks of their respective owners.

Easysoft and the Easysoft logo are registered trademarks of Easysoft Limited.

The software described in this document is provided under a [licence agreement](#) and may be used only in accordance with the terms of that agreement.

Table of contents

Getting started	4
Installing the Easysoft ODBC-QOnline Driver	5
Installing on Windows	6
Uninstalling on Windows	8
Connecting to QuickBooks Online.	9
Connecting from Windows	9
Connection attributes	10
Logging	11
ODBC Driver Manager logging on Windows	11
Easysoft ODBC-QOnline Driver logging on Windows	12
Finding out what product version you have on Windows	13
Client applications.	14
Microsoft Access	15
Linking a table.	15
Importing a table	15
Microsoft Excel	16
Data Connection Wizard	16
Microsoft Query	16
PowerPivot	16
Microsoft Power BI	17
Microsoft SQL Server	18
Oracle	20
Connecting QuickBooks Online to Oracle on Windows	20
LibreOffice	22
Go	23
Node.js	24
Perl	26
PHP	29
Python	31
R	33
About the Easysoft ODBC-QOnline Driver.	35
ODBC API and scalar functions.	36
API functions	36
Scalar functions	38
Data type mapping	41
Finding out more about data types on Windows	42
SQL support.	43
Index	44

Getting started

The Easysoft ODBC-QOnline Driver provides real-time access to QuickBooks Online data from any application that supports ODBC.

The following section shows you how to install the Easysoft ODBC-QOnline Driver and configure the ODBC data source that stores the connection details for your QuickBooks Online file. You're then ready to work with QuickBooks Online data in your application.

- [Installing the Easysoft ODBC-QOnline Driver](#)
- [Connecting to QuickBooks Online](#)
- [Logging](#)

Installing the Easysoft ODBC-QOnline Driver

Install the Easysoft ODBC-QOnline Driver on the computer where the application you want to connect to QuickBooks Online is running.

Installing on Windows

The Windows installation can be done by anyone with local administrator privileges.

1. [Download the Easysoft ODBC-QOnline Driver installer.](#)
2. Follow the onscreen instructions to progress through the installation wizard.

Updating files that are in use

To avoid rebooting your computer, the Easysoft ODBC-QOnline Driver installer prompts you when files that it needs to update are in use by another application or service. This frees the locked files and allows the installation to complete without a system restart. The installer uses the **Restart Manager** to locate the applications that are using files that need updating. These applications are displayed in the **Files in Use** dialog box. To avoid a system restart, choose **Automatically close applications and attempt to restart them after setup is complete**. The Easysoft ODBC-QOnline Driver installer then uses **Restart Manager** to try to stop and restart each application or service in the list. If possible, **Restart Manager** restores applications to the same state that they were in before it shut them down.

Licensing

By default, the installer starts the Easysoft License Manager, because you can't use the Easysoft ODBC-QOnline Driver until you have a license. If you choose not to run Easysoft License Manager as part of the installation process, run License Manager from the **Easysoft** group in the Windows **Start** menu when you're ready to license the Easysoft ODBC-QOnline Driver. These types of license are available:

- A free time-limited trial license, which gives you free and unrestricted use of the product for a limited period (usually 14 days).
- A full license if you have purchased the product. On purchasing the product you are given an authorization code, which you use to obtain a license.

To license the Easysoft ODBC-QOnline Driver:

1. In License Manager, enter your contact details.

You **must** complete the **Name**, **E-Mail Address**, and **Company** fields.

The e-mail address **must** be the same as the one used to register at the Easysoft web site. Otherwise, you won't be able to obtain a trial license.

2. Choose **Request License**.

You're prompted to choose a license type.

3. Do one of the following:

- For a trial license, choose **Time Limited Trial**, and then choose **Next**.

-Or-

- For a purchased license, choose **Non-expiring License**, and then choose **Next**.

4. Choose your product from the drop-down list when prompted, and then choose **Next**.

5. For a purchased license, enter your authorization code when prompted, and then choose **Next**.

6. Choose how to get your license when prompted.

7. Do one of the following:

- Choose **On-line Request** if your machine is connected to the internet and can make outgoing connections to port 8884.

With this method, License Manager automatically requests and then applies your license.

-Or-

- Choose **View Request**. Then open a web browser and go to https://www.easysoft.com/support/licensing/trial_license.html or https://www.easysoft.com/support/licensing/full_license.html, as appropriate. In the web page, enter your machine number (labelled **Number** in the license request). For purchased licenses, you also need to enter your authorization code (labelled **Ref** in the license request).

We'll automatically email your license to the email address you supplied in License Manager.

-Or-

- Choose **Email Request** to email your license request to our licensing team.

Once we've processed your request, we'll email your license to the email address you supplied in License Manager.

8. Close the License Manager windows and then choose **Finish**.

If you chose either **View Request** or **Email Request**, apply your license by double-clicking the email attachment when you get the license email from us. Alternatively, start License Manager from the **Easysoft** folder in the Windows **Start** menu. Then choose **Enter License** and paste the license in the space provided.

Once you've licensed the Easysoft ODBC-QOnline Driver, the installation is complete.

Repairing the installation

The installer can repair a broken Easysoft ODBC-QOnline Driver installation. For example, you can use the installer to restore missing Easysoft ODBC-QOnline Driver files or registry keys. To do this:

1. In the Windows **Taskbar**, enter Add or remove programs in the Windows **Search** box.
2. Select Easysoft ODBC-QOnline Driver in the list, and then choose **Repair**.

Uninstalling on Windows

This section explains how to remove the Easysoft ODBC-QOnline Driver from your system.

Removing Easysoft ODBC-QOnline Driver data sources

Easysoft ODBC-QOnline Driver data sources are not removed when you uninstall the Easysoft ODBC-QOnline Driver. You don't therefore need to recreate your Easysoft ODBC-QOnline Driver data sources if you reinstall or upgrade. If you don't want to keep your Easysoft ODBC-QOnline Driver data sources, use Microsoft **ODBC Data Source Administrator** to remove them, **before** uninstalling the Easysoft ODBC-QOnline Driver:

1. In the Windows **Taskbar**, enter Run in the Windows **Search** box.
2. In the Windows **Run** dialog box, enter:

```
odbcad32.exe
```

3. Locate your data source in either the **User** or **System** tab.
4. Select the data source from the list, and then choose **Remove**.

If the **Remove** button isn't available, close **ODBC Data Source Administrator**, and then, in the Windows **Run** dialog box, enter:

```
%windir%\syswow64\odbcad32.exe
```

Repeat the previous two steps.

Removing the Easysoft ODBC-QOnline Driver

1. In the Windows **Taskbar**, enter Add or remove programs in the Windows **Search** box.
2. Select Easysoft ODBC-QOnline Driver in the list, and then choose **Uninstall**.

Note

Easysoft product licenses are stored in the Windows registry. When you uninstall, your licenses are not removed, so you do not need to relicense the product if you reinstall or upgrade.

Connecting to QuickBooks Online

Applications that support ODBC interface with an ODBC Driver Manager, which is included with the operating system, and also the Easysoft ODBC driver distribution on some platforms. One of the jobs that the ODBC Driver Manager does is to manage ODBC data sources. A data source specifies which ODBC driver to load, which data store to connect to, and how to connect to it.

Before setting up a data source, you must have [successfully installed the Easysoft ODBC-QOnline Driver](#).

Connecting from Windows

Creating an ODBC data source

1. In the Windows **Taskbar Search** box, enter "Run".
2. Do one of the following:
 - If your application is 64-bit, in the **Run** dialog box, enter:

```
odbcad32.exe
```

-Or-

- If your application is 32-bit, in the **Run** dialog box, enter:

```
%windir%\syswow64\odbcad32.exe
```

Note

If you're not sure whether your application is 32-bit or 64-bit, start your application, then in Windows **Task Manager** check whether your application's process name contains (32-bit). For example, the process name for the 32-bit version of Excel is Microsoft Excel (32-bit); the process name for the 64-bit version of Excel is Microsoft Excel. On older versions of Windows, 32-bit applications contain *32 in the process name rather than (32-bit). For applications such as Oracle or SQL Server that run as a service, check the *Background processes* list rather than the **Apps** list in **Task Manager**. If you're running a programming language from within a Windows command-line shell (for example, Command or PowerShell), in your shell, run the .exe file for the programming language. For example, run perl, php, python, or node. In **Task Manager**, expand the process list for **Windows Command Processor** or **Windows PowerShell**, as appropriate, and check whether the process for your programming language contains (32-bit).

3. Do one of the following:
 - To create a data source that only the user you're currently logged in as can access, choose the **User** tab.
If your application is a Windows service (for example, SQL Server or Oracle) creating a user data source won't work, unless the service is running as the same user you're logged in as.
 - To create a data source that all users on this computer can access, choose the **System** tab.
4. Choose **Add**.
5. In the list of ODBC drivers, select Easysoft ODBC-QOnline Driver, and then choose **Finish**.
6. Complete the Easysoft ODBC-QOnline Driver configuration dialog box.
To find out how to do this, refer to the Connection attributes section.
7. Choose the **Connect** button.

The Intuit login page displays in your default browser.

10 Connection attributes

8. Log into QuickBooks Online, and then close the browser tab when prompted.

Data source set up is now complete.

9. To test the connection to QuickBooks Online, choose **Test**.

Note that this doesn't test that the Easysoft ODBC-QOnline Driver is licensed. If you haven't yet [licensed](#) the Easysoft ODBC-QOnline Driver, this ODBC data source won't work with your application, even if the **Test** button succeeds.

Connection attributes

- [Setting on Windows](#)

Setting on Windows

The Easysoft ODBC-QOnline Driver data source configuration dialog box, accessible when you create or edit an Easysoft ODBC-QOnline Driver data source in **ODBC Data Source Administrator** contains a number of attribute fields:

Name	Value
DSN	The name of the data source. You'll need to specify this in your application. For example, your application may prompt you to choose this from a list of DSNs.
Description	Some applications display this to help users identify a particular data source.
Sandbox	Turn on this option if you're connecting to the QuickBooks Online sandbox.
Logging	Whether to turn on Easysoft ODBC-QOnline Driver logging. Normally, you'll only do this if so directed by the Easysoft support team.
Log File	<p>The file name and path of the file you want the driver to write log information to. For example:</p> <p>C:\Windows\Temp\Easysoft.log</p> <p>If the file doesn't exist, the Easysoft ODBC-QOnline Driver creates it.</p>

Logging

If you report an issue to us, we may ask you to turn on ODBC Driver Manager or Easysoft ODBC-QOnline Driver logging, to help us diagnose the cause of the issue.

To turn on logging, refer to the following sections.

Note If your application is a service (for example, Oracle or SQL Server), you may need to restart the service before enabling logging takes effect. To do this on Linux or UNIX, use `service`, `systemctl`, or a vendor-supplied script. To do this on Windows, use the Windows **Services** app.

ODBC Driver Manager logging on Windows

1. In the Windows **Taskbar Search** box, enter "Run".
2. Do one of the following:
 - If your application is 64-bit, in the **Run** dialog box, enter:

```
odbcad32.exe
```

-Or-

- If your application is 32-bit, in the **Run** dialog box, enter:

```
%windir%\syswow64\odbcad32.exe
```

Note If you're not sure whether your application is 32-bit or 64-bit, start your application, then in Windows **Task Manager** check whether your application's process name contains (32-bit). For example, the process name for the 32-bit version of Excel is Microsoft Excel (32-bit); the process name for the 64-bit version of Excel is Microsoft Excel. On older versions of Windows, 32-bit applications contain *32 in the process name rather than (32-bit). For applications such as Oracle or SQL Server that run as a service, check the *Background processes* list rather than the **Apps** list in **Task Manager**. If you're running a programming language from within a Windows command-line shell (for example, Command or PowerShell), in your shell, run the .exe file for the programming language. For example, run perl, php, python, or node. In **Task Manager**, expand the process list for **Windows Command Processor** or **Windows PowerShell**, as appropriate, and check whether the process for your programming language contains (32-bit).

3. Choose the **Tracing** tab.
4. Select **Machine-Wide tracing for all identities**.
5. Enter a log file name and path in the space provided. For example:

```
C:\Windows\Temp\SQL.log
```

6. Choose **Start Tracing Now**.

Note

With SQL Server, you may get two Driver Manager log files, we need both. The first log file is in the folder that you specify in **ODBC Data Source Administrator**. The second file's location is defined by SQL Server. Two possible locations are the top-level folder (for example, C:\SQL.log) or the SQL Server temporary folder (for example, C:\Users\MSSQL\$SQLEXPRESS\AppData\Local\Temp\SQL.log). If the Driver Manager log file isn't in these folders, search for it on the drive where SQL Server is installed.

Easysoft ODBC-QOnline Driver logging on Windows

1. In the Windows **Taskbar Search** box, enter "Run".
2. Do one of the following:
 - If your application is 64-bit, in the **Run** dialog box, enter:

```
odbcad32.exe
```

-Or-

- If your application is 32-bit, in the **Run** dialog box, enter:

```
%windir%\syswow64\odbcad32.exe
```

Note

If you're not sure whether your application is 32-bit or 64-bit, start your application, then in Windows **Task Manager** check whether your application's process name contains (32-bit). For example, the process name for the 32-bit version of Excel is Microsoft Excel (32-bit); the process name for the 64-bit version of Excel is Microsoft Excel. On older versions of Windows, 32-bit applications contain *32 in the process name rather than (32-bit). For applications such as Oracle or SQL Server that run as a service, check the *Background processes* list rather than the **Apps** list in **Task Manager**. If you're running a programming language from within a Windows command-line shell (for example, Command or PowerShell), in your shell, run the .exe file for the programming language. For example, run perl, php, python, or node. In **Task Manager**, expand the process list for **Windows Command Processor** or **Windows PowerShell**, as appropriate, and check whether the process for your programming language contains (32-bit).

3. Do one of the following:
 - If you configured a system data source, choose the **System DSN** tab.
 - Or-
 - If you configured a system data source, choose the **System DSN** tab.
4. Choose your Easysoft ODBC-QOnline Driver data source from the list, and then choose **Configure**.
5. In the Easysoft ODBC-QOnline Driver data source configuration dialog box, turn on **Driver Logging**.
6. Enter a log file name and path in the space provided. For example:

```
C:\Windows\Temp\Easysoft.log
```

Finding out what product version you have on Windows

If you have an issue with the Easysoft ODBC-QOnline Driver, we may ask you to tell us what your product version is. To find this out:

1. In the Windows **Taskbar**, enter “Add or remove programs” in the Windows **Search** box.
2. Select Easysoft ODBC-QOnline Driver in the list.

The product version displays below.

Client applications

How to work with QuickBooks Online data in some example applications and programming languages:

- [Microsoft Access](#)
- [Microsoft Excel](#)
- [Microsoft Power BI](#)
- [SQL Server](#)
- [Oracle](#)
- [LibreOffice](#)
- [Go](#)
- [Node.js](#)
- [Perl](#)
- [PHP](#)
- [Python](#)
- [R](#)

Microsoft Access

1. [Install the Easysoft ODBC-QOnline Driver](#) on same computer as Access.
2. [Configure an ODBC data source](#).
3. Choose one of the following ways to work with your QuickBooks Online data in Access.

Linking a table

1. Open your Microsoft Access database.
2. Choose **External Data**.
3. In the **New Data Source** list, choose **From Other Sources > ODBC Database**.
4. In the **Get External Data** screen, choose **Link to the data source by creating a linked table**, and choose **OK**.
5. In the **Select Data Source** dialog box, choose the **Machine Data Source** tab.
6. Choose your Easysoft ODBC-QOnline Driver ODBC data source from the **Machine Data Source** list, and then choose **OK**.
7. In the **Link Tables** dialog box, choose the tables that you want to link to, and then choose **OK**.

Importing a table

1. Open your Microsoft Access database.
2. Choose **External Data**.
3. In the **New Data Source** list, choose **From Other Sources > ODBC Database**.
4. In the **Get External Data** screen, choose **Import the source data into a new table in the current database**, and choose **OK**.
5. In the **Select Data Source** dialog box, choose the **Machine Data Source** tab.
6. Choose your Easysoft ODBC-QOnline Driver ODBC data source from the **Machine Data Source** list, and then choose **OK**.
7. In the **Import Objects** dialog box, choose the tables you want to import, and then choose **OK**.

Microsoft Excel

1. [Install the Easysoft ODBC-QOnline Driver](#) on same computer as Excel.
2. [Configure an ODBC data source](#).
3. Choose one of the following ways to work with your QuickBooks Online data in Excel.

Data Connection Wizard

1. Choose **Data > Get Data > From Other Sources > From ODBC**.
2. Choose your Easysoft ODBC-QOnline Driver data source from the list, and then choose **OK**.
3. Enter the user name and password for your data store if applicable, otherwise, enter any text string to get past this stage. Choose **Next**.
4. Choose the table that contains the data you want to retrieve, and then choose **Load**.

Microsoft Query

1. Choose **Data > Get Data > From Other Sources > From Microsoft Query**.
2. In the **Choose Data Source** dialog box, choose your QuickBooks Online data source from the list, and then choose **OK**.
3. In the **Query Wizard**, choose the columns that contain the data you want to retrieve, and then click **Next**.
4. If you want to return a subset of the data, use the **Filter Data** screen to filter the results of your query (this is the equivalent of a SQL WHERE clause), and then choose **Next**.
5. If you want to change the sort order of your data, use the **Sort Order** screen to sort the results of your query (this is the equivalent of a SQL ORDER BY clause), and then choose **Next**. Choose **Finish** to return your QuickBooks Online data to Excel.

PowerPivot

1. On the **PowerPivot** tab, choose **Manage**.
2. In the **PowerPivot** window, choose **Get External Data > From Other Sources**.
3. In the **Connect to a Data Source** list, choose **Others (OLEDB/ODBC)**.
4. In the **Specify a Connection** screen, enter a name for your connection in the space provided. Then choose **Build**.
5. In the **Data Link Properties** box, choose your Easysoft ODBC-QOnline Driver data source from the list, and then choose **OK**.
6. Choose **Next**.
7. Choose how to import your QuickBooks Online data and then choose **Finish**.
8. Choose **Close** to return the data to Excel.

Microsoft Power BI

1. [Install the Easysoft ODBC-QOnline Driver](#) on same computer as Power BI Desktop.
2. [Configure an ODBC data source](#).
3. In Power BI Desktop, choose **Get data from another source**.
4. In the **Get Data** dialog box, choose **ODBC**, and then choose **Connect**.
5. In the **From ODBC** dialog box, choose your QuickBooks Online data source, and then choose **OK**.
6. Enter your database user name and password when prompted.

If you make a mistake when entering the user name and password, cancel the connection process. Then in Power BI Desktop **Options and Settings**, edit the data source. Specify the correct user name or password in the data source credentials dialog box. Otherwise, Power BI Desktop will continue to use the cached incorrect credentials.

Note	If you do not normally need to enter a user name and password, enter some dummy strings in the spaces provided.
-------------	---

7. In the **Navigator** dialog box, choose the tables you want to analyse in Power BI Desktop, and then choose **Load**.

Your QuickBooks Online data is now available to use in Power BI visualisations.

Microsoft SQL Server

1. [Install the Easysoft ODBC-QOnline Driver](#) on same computer as SQL Server.
2. [Configure an ODBC data source](#).
3. In Microsoft SQL Server Management Studio, connect to the SQL Server instance you want to create the linked server against.

You need to log on with an account that is a member of the SQL Server sysadmin fixed server role to create a linked server.

4. Right-click **Server Objects**. From the pop-up menu choose **New > Linked Server**.
5. In the **Linked server** box, enter "QuickBooks Online".
6. From the **Provider** list, choose **Microsoft OLE DB Provider for ODBC drivers**.
7. In the **Data source** box, enter the name of your QuickBooks Online data source, and then choose **OK**.

SQL Server verifies the linked server by testing the connection.

- If you get the error "Specified driver could not be loaded due to system error 126: The specified module could not be found," choose **Yes** when prompted whether to keep the linked server. You need to restart your SQL Server instance before you can use the linked server. If SQL Server was already running when you installed the Easysoft ODBC-QOnline Driver, it will not have the latest version of the System Path environment variable. The Easysoft ODBC-QOnline Driver Setup program adds entries for the driver to the System Path. Restarting the instance makes these changes available to SQL Server, allowing it to load the Easysoft ODBC-QOnline Driver.
 - If you made a mistake when specifying the Easysoft ODBC-QOnline Driver, you get the error "Data source name not found and no default driver specified." If you get this error, choose **No** when prompted whether to keep the linked server and edit the value in the **Data source** box.
8. You can query your Easysoft ODBC-QOnline Driver data either by using a:
 - Four part table name in a distributed query.

A four part table name has the format:

```
server_name.[database_name].[schema_name].table_name
```

For data stores where there is no database or schema, Easysoft ODBC drivers return a "dummy" value for both identifiers, because some ODBC applications expect there to be a database and a schema. To find out the identifier names, run:

```
EXEC sp_tables_ex @table_server = 'QuickBooks Online'
```

If present, include these identifiers in your SQL statements. If not present, omit them. For example:

```
SELECT * FROM [QuickBooks Online]..DBO.MyTable
```

The capitalisation of the table name must be the same as it is in the result set returned by `sp_tables_ex`.

- Pass-through query in an OPENQUERY function. For example:

```
SELECT * FROM OPENQUERY([QuickBooks Online], 'SELECT * FROM MyTable')
```

```
-- If you get an "RPC not enabled for this server" message, right-click your
linked server and choose Properties.
```

```
-- In Server Options, set both RPC and RPC Out to `True`.  
EXEC ('INSERT INTO MyTable (MyCol1, MyCol2, MyCol3, MyCol4, MyCol5)  
VALUES (''MyValue1'' , ''MyValue2'' , ''MyValue3'' , ''MyValue4'' ,  
''MyValue5'')')  
AT QuickBooks Online  
  
UPDATE OPENQUERY ([QuickBooks Online], 'SELECT MyCol1 FROM MyTable WHERE  
MyCol1 = ''MyValue1'') SET MyCol1='MyNewValue'  
DELETE OPENQUERY (QuickBooks Online, 'SELECT MyCol1 FROM MyTable WHERE MyCol1  
= ''MyValue1'')
```

SQL Server sends pass-through queries as uninterpreted query strings to the QuickBooks Online. This means that SQL Server does not apply any kind of logic to the query or try to estimate what that query will do.

Oracle

1. [Install the Easysoft ODBC-QOnline Driver](#) on same computer as Oracle.
2. [Configure an ODBC data source](#).
3. Follow the instructions for your Oracle platform.

Connecting QuickBooks Online to Oracle on Windows

1. Create a DG4ODBC init file on your Oracle machine. To do this, change to the %ORACLE_HOME%\hs\admin directory. Create a copy of the file initdg4odbc.ora. Name the new file initquickbooks-online.ora.

Note In these instructions, replace %ORACLE_HOME% with the location of your Oracle HOME directory. For example, C:\app\product\21c\homes\OraDB21Home1.

2. Ensure these parameters and values are present in your init file:

```
HS_FDS_CONNECT_INFO = "QuickBooks Online"
```

Replace QuickBooks Online with the name of your Easysoft ODBC-QOnline Driver data source.

3. Comment out the line that enables DG4ODBC tracing. For example:

```
#HS_FDS_TRACE_LEVEL = <trace_level>
```

4. Add an entry to %ORACLE_HOME%\network\admin\listener.ora that creates a SID_NAME for DG4ODBC. For example:

```
SID_LIST_LISTENER =
(SID_LIST =
(SID_DESC=
(SID_NAME=quickbooks-online)
(ORACLE_HOME=%ORACLE_HOME%)
(PROGRAM=dg4odbc)
)
)
```

5. Add a DG4ODBC entry to %ORACLE_HOME%\network\admin\tnsnames.ora that specifies the SID_NAME created in the previous step. For example:

```
quickbooks-online =
(DESCRIPTION =
(ADDRESS = (PROTOCOL = TCP)(HOST = oracle_host)(PORT = 1521))
(CONNECT_DATA =
(SID = quickbooks-online)
)
(HS = OK)
)
```

Replace oracle_host with the host name of your Oracle machine.

6. Start (or restart) the Oracle Listener:

```
cd %ORACLE_HOME%\bin
lsnrctl stop
```

```
lsnrctl start
```

7. Connect to your Oracle database in SQL*Plus.
8. In SQL*Plus, create a database link for QuickBooks Online. For example:

```
CREATE PUBLIC DATABASE LINK quickbooks-onlineLink
CONNECT TO "dbuser" IDENTIFIED BY "dbpassword"
USING 'quickbooks-online';
```

Replace dbuser and dbpassword with your backend user name and password, if applicable.

9. Try querying and updating your QuickBooks Online data. For example:

```
SELECT "MyCol1" FROM "MyTable"@quickbooks-onlineLink;

DECLARE
    num_rows integer;
BEGIN
    num_rows:=DBMS_HS_PASSTHROUGH.EXECUTE_IMMEDIATE@quickbooks-onlineLink
('INSERT INTO MyTable (MyCol1, MyCol2, MyCol3, MyCol4, MyCol5) VALUES
('MyValue1'', 'MyValue2'', 'MyValue3'', 'MyValue4'', 'MyValue5''));
END;
/

DECLARE
    num_rows integer;
BEGIN
    num_rows:=DBMS_HS_PASSTHROUGH.EXECUTE_IMMEDIATE@quickbooks-onlineLink
('UPDATE "MyTable" SET "MyCol1" = 'MyNewValue' WHERE "MyCol1" = 'MyValue1'');
END;
/

DECLARE
    num_rows integer;
BEGIN
    num_rows:=DBMS_HS_PASSTHROUGH.EXECUTE_IMMEDIATE@quickbooks-onlineLink
('DELETE from "MyTable" WHERE MyCol1 = 'MyValue1'');
END;
/
```

Notes

- If you have problems connecting to QuickBooks Online from Oracle, enable DG4ODBC tracing and check the trace files written to the %ORACLE_HOME%\hs\trace directory. To enable DG4ODBC tracing, add the line HS_FDS_TRACE_LEVEL = DEBUG to initQuickBooks Online.ora and then start or restart the Oracle listener. If the trace directory does not exist, create it.
- If you enable ODBC Driver Manager tracing, but do not get a log file in the location you specify, try looking in the top-level folder (for example, C:\SQL.log). Alternatively, in **ODBC Data Source Administrator**, change the trace file location to the Windows TEMP directory.

LibreOffice

1. [Install the Easysoft ODBC-QOnline Driver](#) on same computer as LibreOffice.
2. [Configure an ODBC data source](#).
3. Choose **File > New > Database**.
4. Choose **Connect to an existing database**.
5. Choose **ODBC** in the list, and then choose **Next**.
6. Choose **Browse**, double-click your data source, and then choose **Next**.
7. If your database requires a database user name, enter it in the **User name** box. If this user needs to supply a password choose the **Password required** check box.
8. Choose **Finish**.
9. Save the database when prompted.

The database opens in a new Base window. From here you can access your data.

10. In the left pane of the database window, choose the **Tables** icon to display a hierarchy of tables. Enter the database password if prompted, and then choose **OK**.
11. To retrieve the data in a table, in the **Tables** pane, double-click a table.
12. Choose the **Queries** icon to create a query.

Use any of the methods listed in the **Tasks** pane to create a query.

Go

1. [Install the Easysoft ODBC-QOnline Driver](#) on same computer as Go.
2. [Configure an ODBC data source](#).
3. Install the `odbc` package for Go:

```
go mod init test
go get github.com/alexbrainman/odbc
```

4. Create and then use Go to run this script, which retrieves some QuickBooks Online data:

```
package main

import (
    _ "github.com/alexbrainman/odbc"
    "database/sql"
    "log"
)

func main() {
    // Replace the DSN value with the name of your ODBC data source.
    db, err := sql.Open("odbc",
        "DSN=QuickBooks Online")
    if err != nil {
        log.Fatal(err)
    }

    var (
        name string
    )

    rows, err := db.Query("SELECT MyCol1 FROM MyTable")
    if err != nil {
        log.Fatal(err)
    }
    defer rows.Close()
    for rows.Next() {
        err := rows.Scan(&name)
        if err != nil {
            log.Fatal(err)
        }
        log.Println(name)
    }
    err = rows.Err()
    if err != nil {
        log.Fatal(err)
    }

    defer db.Close()
}
```

Node.js

1. [Install the Easysoft ODBC-QOnline Driver](#) on same computer as Node.js.
2. [Configure an ODBC data source.](#)
3. Install the `odbc` module for Node.js:

```
npm install odbc
```

4. Create and then use Node.js to run this script, which retrieves some QuickBooks Online data:

```
const odbc = require('odbc');
// Replace QuickBooks Online with the name of your Easysoft ODBC-QOnline Driver
// data source.
const connection = odbc.connect('DSN=QuickBooks Online', (error, connection) => {
  connection.query('SELECT MyCol1 FROM MyTable', (error, result) => {
    if (error) { console.error(error) }
    console.log(result);
  });
});
```

5. This script retrieves the tables and views in your Easysoft ODBC-QOnline Driver data source:

```
const odbc = require('odbc');
const connection = odbc.connect('DSN=QuickBooks Online', (error, connection) => {
  connection.tables(null, null, null, null, (error, result) => {
    if (error) { return; }
    const util = require('util');
    console.log(util.inspect(result, {maxLength: null, depth:null}))
  });
});
```

6. This script retrieves the names of the columns in these tables and views:

```
const odbc = require('odbc');
const connection = odbc.connect('DSN=QuickBooks Online', (error, connection) => {
  connection.columns(null, null, null, null, (error, result) => {
    if (error) { return; }
    const util = require('util');
    console.log(util.inspect(result, {maxLength: null, depth:null}))
  });
});
```

7. These scripts insert, update, and then delete some QuickBooks Online data:

```
const odbc = require("odbc");
const connection = odbc.connect("DSN=QuickBooks Online", (error, connection) => {
  connection.query("INSERT INTO
  MyTable (
    MyCol1,
    MyCol2,
    MyCol3,
    MyCol4,
    MyCol5
```



```

    )
VALUES
    (
        'MyValue1',
        'MyValue2',
        'MyValue3',
        'MyValue4',
        'MyValue5'
    ), (error, result) => {
        if (error) { console.error(error) }
        console.log(result);
    });
});

const odbc = require("odbc");
const connection = odbc.connect("DSN=QuickBooks Online", (error, connection) => {
    connection.query("UPDATE MyTable SET MyCol1 = 'MyNewValue' WHERE MyCol1 =
'MyValue1'", (error, result) => {
        if (error) { console.error(error) }
        console.log(result);
    });
});

const odbc = require("odbc");
const connection = odbc.connect("DSN=QuickBooks Online", (error, connection) => {
    connection.query("DELETE FROM MyTable WHERE MyCol1 = 'MyValue1'", (error,
result) => {
        if (error) { console.error(error) }
        console.log(result);
    });
});
});

```

Perl

1. [Install the Easysoft ODBC-QOnline Driver](#) on same computer as Perl.
2. [Configure an ODBC data source](#).
3. Check whether your Perl distribution supports ODBC:

```
perl -e 'use DBD::ODBC;'
```

4. Do one of the following:
 - If you get no output, your Perl distribution supports ODBC. Skip to the next step.
 - If you get:

```
Can't locate DBD/ODBC.pm
```

you need to [install DBD::ODBC](#) before you can use the Easysoft ODBC-QOnline Driver to connect to QuickBooks Online.

5. Create and then use Perl to run this script, which retrieves some QuickBooks Online data:

```
use strict;
use DBI;
# Replace QuickBooks Online with the name of your Easysoft ODBC-QOnline Driver
data source.
my $dbh = DBI-> connect('dbi:ODBC:QuickBooks Online');

my $sql = "SELECT MyCol1 FROM MyTable";

my $sth = $dbh->prepare($sql)
    or die "Can't prepare statement: $DBI::errstr";

$sth->execute();

my($Col);

# Fetch and display the result set values.
while(($Col) = $sth->fetchrow()){
    print("$Col\n");
}

$dbh->disconnect if ($dbh);
```

6. This script retrieves the tables and views in your Easysoft ODBC-QOnline Driver data source:

```
use strict;
use DBI;
my $dbh = DBI-> connect('dbi:ODBC:QuickBooks Online');

my $sth = $dbh->table_info()
    or die "Can't prepare statement: $DBI::errstr";

my @row;

while (@row = $sth->fetchrow_array) {
```

```

        print join(", ", @row), "\n";
    }

    $dbh->disconnect if ($dbh);

```

7. This script retrieves the names of the columns in these tables and views:

```

use strict;
use DBI;
my $dbh = DBI-> connect('dbi:ODBC:QuickBooks Online');

my $sth = $dbh->column_info('', '', '', '')
    or die "Can't prepare statement: $DBI::errstr";

my @row;
while (@row = $sth->fetchrow_array) {
    print join(", ", @row), "\n";
}

$dbh->disconnect if ($dbh);

```

8. These scripts insert, update, and then delete some QuickBooks Online data:

```

use strict;
use DBI;
my $dbh = DBI-> connect('dbi:ODBC:QuickBooks Online');

my $sth = $dbh->prepare(q/INSERT INTO MyTable (MyCol1, MyCol2, MyCol3, MyCol4,
MyCol5) VALUES (?, ?, ?, ?, ?)/)
    or die "Can't prepare statement: $DBI::errstr";

$sth->execute('MyValue1', 'MyValue2', 'MyValue3', 'MyValue4', 'MyValue5');

$dbh->disconnect if ($dbh);

use strict;
use DBI;
my $dbh = DBI-> connect('dbi:ODBC:QuickBooks Online');

my $sth = $dbh->prepare('UPDATE MyTable SET MyCol1 = \'MyNewValue\' WHERE MyCol1
= ?')
    or die "Can't prepare statement: $DBI::errstr";

$sth->execute('MyValue1');

$dbh->disconnect if ($dbh);

use strict;
use DBI;
my $dbh = DBI-> connect('dbi:ODBC:QuickBooks Online');

my $sth = $dbh->prepare('DELETE FROM MyTable WHERE MyCol1 = ?')

```

```
        or die "Can't prepare statement: $DBI::errstr";

$sth->execute('MyValue1');

$dbh->disconnect if ($dbh);
```

Further information

- [Perl DBI DBD::ODBC tutorial: Drivers, data sources, and connection](#)

PHP

1. [Install the Easysoft ODBC-QOnline Driver](#) on same computer as PHP.
2. [Configure an ODBC data source](#).
3. Check whether your PHP distribution supports ODBC. In php.ini, make sure there is no comment character (;) before the extension_dir and extension=odbc settings (;extension_dir=directory becomes extension_dir=directory and ;extension=odbc becomes extension=odbc).
4. Create and then use PHP to run this script, which retrieves some QuickBooks Online data:

```
<?php
// Replace QuickBooks Online with the name of your Easysoft ODBC-QOnline Driver
data source.
// If your database requires a user name and password, supply them in the
odbc_connect_call.
$con = odbc_connect("QuickBooks Online", "", "");
$stmt = odbc_exec($con, "SELECT * FROM MyTable");
// You may need to change the capitalisation of MyCol1 to all upper case or all
lower case.
while ($row = odbc_fetch_array($stmt)) {
    echo "MyCol1 = ", $row["MyCol1"], "\n";
}
odbc_close($con);
?>
```

5. This script retrieves the tables and views in your Easysoft ODBC-QOnline Driver data source:

```
<?php
$con = odbc_connect("QuickBooks Online", "", "");
$tables = odbc_tables($con);
while (($row = odbc_fetch_array($tables))) {
    print_r($row);
}
odbc_close($con);
?>
```

6. This script retrieves the names of the columns in these tables and views:

```
<?php
$con = odbc_connect("QuickBooks Online", "", "");
$columns = odbc_columns($con);
while (($row = odbc_fetch_array($columns))) {
    print_r($row);
}
odbc_close($con);
?>
```

7. These scripts insert, update, and then delete some QuickBooks Online data:

```
<?php
$cnx = odbc_connect("QuickBooks Online", "", "");
$stmt = odbc_prepare($cnx, "INSERT INTO MyTable (MyCol1, MyCol2, MyCol3,
MyCol4, MyCol5) VALUES (?, ?, ?, ?, ?)");
```

```
$success = odbc_execute($stmt, array('MyValue1', 'MyValue2', 'MyValue3',  
'MyValue4', 'MyValue5'));  
odbc_close($cnx);  
?>  
  
<?php  
$cnx = odbc_connect("QuickBooks Online", "", "");  
$stmt = odbc_prepare($cnx, "UPDATE MyTable SET MyCol1 = 'MyNewValue' WHERE  
MyCol1 = ?");  
$success = odbc_execute($stmt, array('MyValue1'));  
odbc_close($cnx);  
?>  
  
<?php  
$cnx = odbc_connect("QuickBooks Online", "", "");  
$stmt = odbc_prepare($cnx, "DELETE FROM MyTable WHERE MyCol1 = ?");  
$success = odbc_execute($stmt, array('MyValue1'));  
odbc_close($cnx);  
?>
```

Further information

- [Easysoft PHP tutorials and code samples](#)

Python

1. [Install the Easysoft ODBC-QOnline Driver](#) on same computer as Python.
2. [Configure an ODBC data source](#).
3. Check whether your Python distribution supports ODBC.

```
pip list
```

If you don't have pip installed:

```
curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py
python get-pip.py
```

4. Do one of the following:
 - If the output contains pyodbc, your Python distribution supports ODBC. Skip to the next step.
 - If the output does not contain pyodbc, use pip to install this module:

```
pip install pyodbc
```

5. Create and then use Python to run this script, which retrieves some QuickBooks Online data:

```
import pyodbc

# Replace QuickBooks Online with the name of your Easysoft ODBC-QOnline Driver
data source.
cnxn = pyodbc.connect("DSN=QuickBooks Online")
cursor = cnxn.cursor()
sql = "SELECT MyCol1 FROM MyTable"
cursor.execute(sql)
rows = cursor.fetchall()
# You may need to change the capitalisation of MyCol1 to all upper case or all
lower case.
for row in rows:
    print(row.MyCol1)
exit()
```

6. This script retrieves the tables and views in your Easysoft ODBC-QOnline Driver data source:

```
import pyodbc

# Replace QuickBooks Online with the name of your Easysoft ODBC-QOnline Driver
data source.
cnxn = pyodbc.connect("DSN=QuickBooks Online")
cursor = cnxn.cursor()
cursor.tables()
rows = cursor.fetchall()
for row in rows:
    print(row.table_name)
exit()
```

7. This script retrieves the names of the columns in these tables and views:

```
import pyodbc
```

```
# Replace QuickBooks Online with the name of your Easysoft ODBC-QOnline Driver
data source.
cnxn = pyodbc.connect("DSN=QuickBooks Online")
cursor = cnxn.cursor()
cursor.columns()
rows = cursor.fetchall()
for row in rows:
    print(row.table_name, row.column_name)
exit()
```

8. These scripts insert, update, and then delete some QuickBooks Online data:

```
import pyodbc

cnxn = pyodbc.connect("DSN=QuickBooks Online")
cursor = cnxn.cursor()
sql = "INSERT INTO MyTable (MyCol1, MyCol2, MyCol3, MyCol4, MyCol5) VALUES (?, ?, ?, ?, ?)"
cursor.execute(sql, 'MyValue1', 'MyValue2', 'MyValue3', 'MyValue4', 'MyValue5')
cursor.commit()
exit()
```

```
import pyodbc

cnxn = pyodbc.connect("DSN=QuickBooks Online")
cursor = cnxn.cursor()
sql = "UPDATE MyTable SET MyCol1 = 'MyNewValue' WHERE MyCol1 = ?"
cursor.execute(sql, 'MyValue1')
cursor.commit()
exit()
```

```
import pyodbc

cnxn = pyodbc.connect("DSN=QuickBooks Online")
cursor = cnxn.cursor()
sql = "DELETE FROM MyTable WHERE MyCol1 = ?"
cursor.execute(sql, 'MyValue1')
cursor.commit()
exit()
```

Further information

- [Easysoft Python tutorials and code samples](#)

R

1. [Install the Easysoft ODBC-QOnline Driver](#) on same computer as R.
2. [Configure an ODBC data source](#).
3. In R Console, check whether your R distribution supports ODBC.

```
library("RODBC")
```

4. Do one of the following:
 - If you get no output, you have the ODBC library for R. Skip to the next step.
 - If you get an "there is no package" error, install the ODBC library for R:

```
install.packages("RODBC")
```

5. Create and then use R to run this script, which retrieves some QuickBooks Online data:

```
library("RODBC")
# Replace QuickBooks Online with the name of your Easysoft ODBC-QOnline Driver
data source.
ch <- odbcConnect("QuickBooks Online")
sqlQuery(ch, paste("SELECT MyCol1 FROM MyTable"))
odbcClose(ch)
quit()
```

6. This script retrieves the tables and views in your Easysoft ODBC-QOnline Driver data source:

```
library("RODBC")
# Replace QuickBooks Online with the name of your Easysoft ODBC-QOnline Driver
data source.
ch <- odbcConnect("QuickBooks Online")
sqlTables(ch)
odbcClose(ch)
quit()
```

7. This script retrieves the names of the columns in the specified table or view:

```
library("RODBC")
# Replace QuickBooks Online with the name of your Easysoft ODBC-QOnline Driver
data source.
ch <- odbcConnect("QuickBooks Online")
# You may need to change the capitalisation of MyTable to all upper case or all
lower case.
sqlColumns(ch, sqtable="MyTable")
odbcClose(ch)
quit()
```

8. These scripts insert, update, and then delete some QuickBooks Online data:

```
library("RODBC")
ch <- odbcConnect("QuickBooks Online")
sqlQuery(ch, paste("INSERT INTO MyTable (MyCol1, MyCol2, MyCol3, MyCol4, MyCol5)
VALUES ('MyValue1', 'MyValue2', 'MyValue3', 'MyValue4', 'MyValue5')"))
odbcClose(ch)
```

```
quit()

library("RODBC")
ch <- odbcConnect("QuickBooks Online")
sqlQuery(ch, paste("UPDATE MyTable SET MyCol1 = 'MyNewValue' WHERE MyCol1 = 'MyValue1'"))
odbcClose(ch)
quit()

library("RODBC")
ch <- odbcConnect("QuickBooks Online")
sqlQuery(ch, paste("DELETE FROM MyTable WHERE MyCol1 = 'MyValue1'"))
odbcClose(ch)
quit()
```

About the Easysoft ODBC-QOnline Driver

The Easysoft ODBC-QOnline Driver provides real-time access to QuickBooks Online data from any application that supports ODBC.

In this section:

- [ODBC API and scalar functions](#)
- [Data type mapping](#)
- [SQL support](#)

ODBC API and scalar functions

API functions

Use this table to find out what ODBC API functions the Easysoft ODBC-QOnline Driver supports:

Function	Status
SQLAllocConnect	Supported
SQLAllocEnv	Supported
SQLAllocHandle	Supported
SQLAllocStmt	Supported
SQLBindCol	Supported
SQLBindParameter	Supported
SQLBrowseConnect	Not supported
SQLBulkOperations	Not supported
SQLCancel	Supported
SQLCloseCursor	Supported
SQLColAttribute	Supported
SQLColAttributes	Supported
SQLColumnPrivileges	Not supported
SQLColumns	Supported
SQLConnect	Supported
SQLCopyDesc	Supported
SQLDisconnect	Supported
SQLDriverConnect	Supported
SQLDrivers	Supported
SQLEndTran	Supported
SQLError	Supported
SQLExecDirect	Supported
SQLExecute	Supported
SQLExtendedFetch	Supported
SQLFetch	Supported
SQLFetchScroll	Supported
SQLForeignKeys	Supported
SQLFreeConnect	Supported
SQLFreeEnv	Supported
SQLFreeHandle	Supported
SQLFreeStmt	Supported

Function	Status
SQLGetConnectAtt	Supported
SQLGetConnectOption	Supported
SQLGetCursorName	Supported
SQLGetData	Supported
SQLGetDescField	Supported
SQLGetDescRec	Supported
SQLGetDiagField	Supported
SQLGetDiagRec	Supported
SQLGetEnvAttr	Supported
SQLGetFunctions	Supported
SQLGetInfo	Supported
SQLGetStmtAttr	Supported
SQLGetStmtOption	Supported
SQLGetTypeInfo	Supported
SQLMoreResults	Supported
SQLNativeSql	Supported
SQLNumParams	Supported
SQLNumResultCols	Supported
SQLParamData	Supported
SQLParamOptions	Supported
SQLPrepare	Supported
SQLPrimaryKeys	Supported
SQLProcedureColumns	Supported
SQLProcedures	Supported
SQLPutData	Supported
SQLRowCount	Supported
SQLSetConnectAttr	Supported
SQLSetConnectOption	Supported
SQLSetCursorName	Supported
SQLSetDescField	Supported
SQLSetDescRec	Supported
SQLSetEnvAttr	Supported
SQLSetParam	Supported
SQLSetPos	Supported
SQLSetScrollOptions	Supported

Function	Status
SQLSetStmtOption	Supported
SQLSetStmtAttr	Supported
SQLStatistics	Supported
SQLTablePrivileges	Not supported
SQLTables	Supported
SQLTransact	Supported

Scalar functions

The Easysoft ODBC-QOnline Driver supports a number of scalar functions:

- [String functions](#)
- [Numeric functions](#)
- [Time, date, and interval functions](#)
- [System functions](#)
- [Conversion functions](#)

Use either the SQL-92 or ODBC syntax with scalar functions. For example:

```
SELECT
    Invoice_Id,
    Customer_Name,
    EXTRACT(YEAR FROM Due_Date) as "Year"
FROM
    Invoice

SELECT
    Invoice_Id,
    Customer_Name,
    {fn EXTRACT(YEAR FROM Due_Date)} as "Year"
FROM
    Invoice
```

String functions

The Easysoft ODBC-QOnline Driver supports these [string](#) functions:

- `ASCII(string_exp)`
- `BIT_LENGTH(string_exp)`
- `CHAR(code)`
- `CHAR_LENGTH(string_exp)`
- `CHARACTER_LENGTH(string_exp)`
- `CONCAT(string_exp1, string_exp2)`
- `DIFFERENCE(string_exp1, string_exp2)`
- `INSERT(string_exp1, start, length, string_exp2)`
- `LCASE(string_exp)`
- `LEFT(string_exp, count)`
- `LENGTH(string_exp)`
- `LOCATE(string_exp1, string_exp2[start])`
- `LTRIM(string_exp)`

- OCTET_LENGTH(*string_exp*)
- POSITION(*char_exp* IN *char_exp*)
- REPEAT(*string_exp*, *count*)
- REPLACE(*string_exp1*, *string_exp2*, *string_exp3*)
- RIGHT(*string_exp*, *count*)
- RTRIM(*string_exp*)
- SOUNDEX(*string_exp*)
- SPACE(*count*)
- UCASE(*string_exp*)

Numeric functions

The Easysoft ODBC-QOnline Driver supports these [numeric](#) functions:

- ABS(*numeric_exp*)
- ACOS(*float_exp*)
- ASIN(*float_exp*)
- ATAN(*float_exp*)
- CEILING(*numeric_exp*)
- COS(*float_exp*)
- COT(*float_exp*)
- DEGREES(*numeric_exp*)
- EXP(*float_exp*)
- FLOOR(*numeric_exp*)
- LOG(*float_exp*)
- LOG10(*float_exp*)
- MOD(*integer_exp1*, *integer_exp2*)
- PI()
- POWER(*numeric_exp*, *integer_exp*)
- RADIANS(*numeric_exp*)
- RAND([*integer_exp*])
- ROUND(*numeric_exp*, *integer_exp*)
- SIGN(*numeric_exp*)
- SIN(*float_exp*)
- SQRT(*float_exp*)
- TAN(*float_exp*)
- TRUNCATE(*numeric_exp*, *integer_exp*)

Time, date, and interval functions

The Easysoft ODBC-QOnline Driver supports these [time, date, and interval](#) functions:

- CURRENT_DATE()
- CURRENT_TIME[(*time-precision*)]
- CURRENT_TIMESTAMP[(*timestamp-precision*)]
- DAYNAME(*date_exp*)
- DAYOFMONTH(*date_exp*)
- DAYOFWEEK(*date_exp*)
- DAYOFYEAR(*date_exp*)
- EXTRACT(*extract-field* FROM *extract-source*)
- HOUR(*time_exp*)
- MINUTE(*time_exp*)
- MONTH(*date_exp*)
- MONTHNAME(*date_exp*)

- NOW()
- QUARTER(*date_exp*)
- SECOND(*time_exp*)
- TIMESTAMPADD(*interval*, *integer_exp*, *timestamp_exp*)
- TIMESTAMPDIFF(*interval*, *timestamp_exp1*, *timestamp_exp2*)
- WEEK(*date_exp*)
- YEAR(*date_exp*)

System functions

The Easysoft ODBC-QOnline Driver supports these [system](#) functions:

- DATABASE()
- USER()

Conversion functions

The Easysoft ODBC-QOnline Driver supports both the [SQL-92 CAST](#) function and the [ODBC CONVERT](#) function for conversion between compatible data types.

Data type mapping

The Easysoft ODBC-QOnline Driver maps QuickBooks Online data types to ODBC data types in this way:

QuickBooks Online data type	ODBC data type
LONG NVARCHAR	SQL_WLONGVARCHAR
NATIONAL CHARACTER VARYING	SQL_WVARCHAR
NATIONAL CHARACTER	SQL_WCHAR
BIT	SQL_BIT
TINYINT	SQL_TINYINT
BIGINT	SQL_BIGINT
LONG VARBINARY	SQL_LONGVARBINARY
VARBINARY	SQL_VARBINARY
BINARY	SQL_BINARY
LONG VARCHAR	SQL_LONGVARCHAR
CHARACTER	SQL_CHAR
NUMERIC	SQL_NUMERIC
DECIMAL	SQL_DECIMAL
INTEGER	SQL_INTEGER
INTEGER AUTONUMBER	SQL_INTEGER
SMALLINT	SQL_SMALLINT
FLOAT	SQL_FLOAT
REAL	SQL_REAL
DOUBLE PRECISION	SQL_DOUBLE
CHARACTER VARYING	SQL_VARCHAR
DATE	SQL_TYPE_DATE
TIME	SQL_TIME
TIMESTAMP	SQL_TYPE_TIMESTAMP
INTERVAL YEAR	SQL_INTERVAL_YEAR
INTERVAL MONTH	SQL_INTERVAL_MONTH
INTERVAL DAY	SQL_INTERVAL_DAY
INTERVAL HOUR	SQL_INTERVAL_HOUR
INTERVAL MINUTE	SQL_INTERVAL_MINUTE
INTERVAL SECOND	SQL_INTERVAL_SECOND
INTERVAL DAY TO HOUR	SQL_INTERVAL_DAY_TO_HOUR
INTERVAL DAY TO MINUTE	SQL_INTERVAL_DAY_TO_MINUTE
INTERVAL DAY TO SECOND	SQL_INTERVAL_DAY_TO_SECOND

QuickBooks Online data type	ODBC data type
INTERVAL HOUR TO MINUTE	SQL_INTERVAL_HOUR_TO_MINUTE
INTERVAL HOUR TO SECOND	SQL_INTERVAL_HOUR_TO_SECOND
INTERVAL MINUTE TO SECOND	SQL_INTERVAL_MINUTE_TO_SECOND

Finding out more about data types on Windows

If you need more information about a data types, for example, the precision and scale, use Microsoft's ODBC Test to do this.

1. Download the version of ODBC Test that matches your application's architecture from:

<https://www.easysoft.com/ftp/pub/utils/windows/odbc-test/>

2. Copy both files to a folder on the machine where Easysoft ODBC-QOnline Driver is installed.
3. Double-click **odbcte32.exe**.
4. Select **Con > Full Connect**.
5. Choose your Easysoft ODBC-QOnline Driver data source from the list.
6. Choose **Catalog > SQLGetTypeInfo**.
7. Either choose **SQL_ALL_TYPES=0 (1.0)** or a specific data type from the **DataType** list.
8. Choose **Results > Get Data All**.

SQL support

The Easysoft ODBC-QOnline Driver supports these SQL statements, clauses, and operators:

- SELECT
- SELECT DISTINCT
- WHERE
- ORDER BY
- AND
- OR
- NOT
- INSERT INTO
- NULL
- UPDATE
- DELETE
- TOP
- MIN
- MAX
- COUNT
- SUM
- AVG
- LIKE
- WILDCARDS
- IN
- BETWEEN
- ALIASES
- JOINS
- UNION
- GROUP BY
- HAVING
- EXISTS
- CASE

Index

A

Access

- importing QuickBooks Online data, [15](#)
- linking QuickBooks Online data, [15](#)

B

Base

- working with QuickBooks Online data, [22](#)

C

- connecting to QuickBooks Online, [9](#)

D

- data source attributes, [10](#)

data sources

- removing
 - on Windows, [8](#)

- data type mapping, [41](#)

- Description attribute, [10](#)

DG4ODBC

- working with QuickBooks Online data, [20](#)

- DSN attribute, [10](#)

E

Easysoft ODBC-QOnline Driver

- adding ODBC data sources
 - on Windows, [9](#)
- connecting to QuickBooks Online, [9](#)
- data source attributes, [10](#)
- data type mapping, [41](#)
- installing
 - on Windows, [6](#)
- logging, [11](#)
- ODBC API support, [36](#)
- scalar function support, [38](#)
- SQL support, [43](#)
- uninstalling
 - on Windows, [8](#)

Excel

- importing QuickBooks Online data with Data Connection Wizard, [16](#)
- importing QuickBooks Online data with PowerPivot, [16](#)
- importing QuickBooks Online data with Query, [16](#)

G

Go

- working with QuickBooks Online data, [23](#)

I

- installing the Easysoft ODBC-QOnline Driver, [5](#)

L

LibreOffice

- working with QuickBooks Online data, [22](#)
- linked server
 - working with QuickBooks Online data, [18](#)

- Log File attribute, [10](#)

- log files, [11](#)

- Logging attribute, [10](#)

N

Node.js

- working with QuickBooks Online data, [24](#)

O

- ODBC API function support, [36](#)

ODBC data sources

- adding
 - on Windows, [9](#)
- removing
 - on Windows, [8](#)

Oracle

- working with QuickBooks Online data, [20](#)

P

Perl

- working with QuickBooks Online data, [26](#)

PHP

- working with QuickBooks Online data, [29](#)

Power BI

- importing QuickBooks Online data, [17](#)

PowerPivot

- importing QuickBooks Online data, [16](#)

Python

- working with QuickBooks Online data, [31](#)

R

R

- working with QuickBooks Online data, [33](#)

S

- Sandbox attribute, [10](#)

- scalar function support, [38](#)

SQL Server

- working with QuickBooks Online data, [18](#)

- SQL support, [43](#)

T

- trace files, [11](#)

U

uninstalling

- on Windows, [8](#)