



---

# **Developer Documentation**

 **DBQT**  
**Database Query Tool**

---

## Table of Contents

<b>1. INTRODUCTION .....</b>	<b>3</b>
<b>2. INTERFACE ICONNECTIONUSERCONTROL.....</b>	<b>4</b>
2.1 INTERFACE DESCRIPTION .....	4
2.2 EXAMPLE.....	5
2.2.1 <i>Relevant source code of the example User Control.....</i>	<i>6</i>



## **1. Introduction**

The Database Query Tool, from now on referred to as DBQT, has been developed for advanced users who want to analyze their databases with the help of SQL (Standard Query Language). It supports all common subsets of SQL such as DML (Data Manipulating Language) and DDL (Data Definition Language). With DBQT you can connect to nearly any kind of database system as for example:

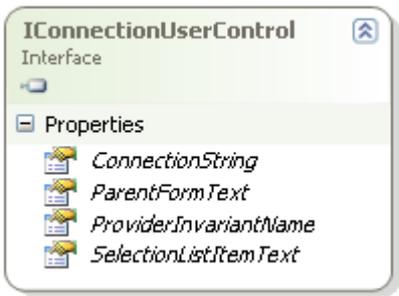
- Microsoft SQL Server
- Microsoft Access
- Oracle
- MySQL
- etc.

By default you can create a connection to any database which there is an OLE DB Provider installed for on your computer. Such an OLE DB connection will be created with the help of the "Data Link Properties" dialog that is an integral part of the Windows operation system. Moreover DBQT provides an interface that can be used to implement user-specific user controls for accessing any kind of database, on condition that there exists a .NET Framework data provider for it. The implementation of such a user control can easily be done by any .NET developer. That way it is possible to provide the possibility to create SQL Server connections using the native SqlClient data provider or it is possible to access MySQL databases with the help of the MySQL Connector/Net 5.1 database driver as well as it possible to attach Oracle databases using the OracleClient data provider for .NET.

## 2. Interface IConnectionUserControl

To extend DBQT with possibilities to connect to any database for which there is a .NET Framework data provider ("managed provider") a .NET developer can implement a custom user control that on the one hand must inherit from the "System.Windows.Forms.UserControl" and on the other hand must implement the interface named "IConnecitonUserControl". This interface is defined within the assembly "DBQueryTool.Interfaces.dll" that can be found within the application directory of DBQT.

### 2.1 Interface description

	<p>This interface can be implemented to create user controls that will be loaded into the memory by DBQT at runtime and then will be embedded into a "CustomConnectionForm" dialog that can be called by the user to create a connection to the database. The interface is quite small so it should be no problem to understand how it works.</p>
---	---

Properties	
String : ConnectionString	The connection-string that will be used by DBQT to create the database connection. An empty string (string.empty) will be interpreted as a „Cancel“ of the operation and DBQT will not try to create a connection.
String : ProviderInvariantName	The name of the managed .NET data provider that will be used to create the database connection.
String : SelectionListItemText	Text that will be displayed in the drop-down list from where the user can select the dialog.
String : ParentFormText	Text that will be displayed as the caption of the dialog the user control is placed upon.

Once this interface has been implemented, it must be compiled into an assembly (DLL) that must be placed into the application directory of DBQT. From there, it will be loaded by reflection whenever DBQT is started.

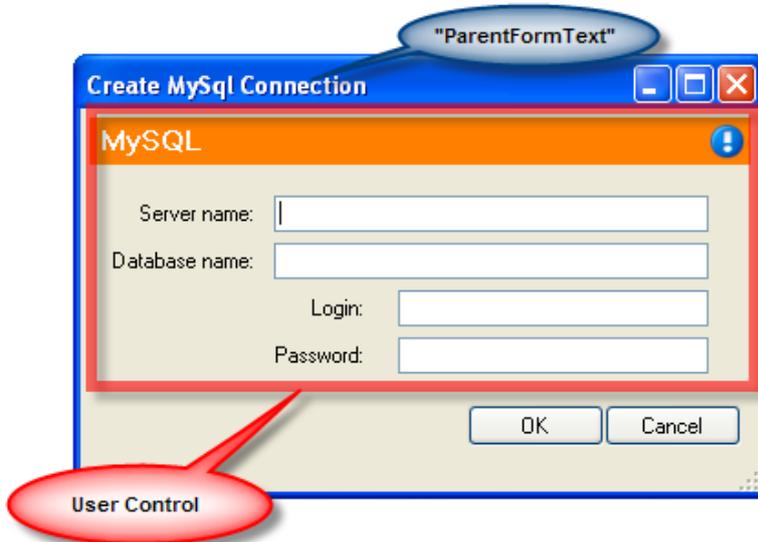
## 2.2 Example

Let's take a look at a user control that is used to create a connection to a MySQL database as an example. This user control is implemented within the assembly "DBQueryTool.MySqlConnectionControl.dll" that is distributed with the DBQT setup. It can be found within the application directory.

The first picture shows "Connect..." drop-down list with the text "MySQL..." that is taken from the "SelectedItemText" property.



The next picture shows the custom User Control (within the red frame) that implements "IConnectionUserControl" and also the caption of the dialog that comes from the property "ParentFormText".





### 2.2.1 Relevant source code of the example User Control

```
namespace DBQT.ConnectionControlLibrary.MySql
{
    public partial class MySqlConnectionControl : UserControl , IConnectionUserControl
    {
        public MySqlConnectionControl()
        {
            InitializeComponent();
        }

        #region IConnectionUserControl Members

        public string ConnectionString
        {
            get
            {
                return SetConnectionString();
            }
        }

        public string ProviderInvariantName
        {
            get
            {
                return "MySql.Data.MySqlClient";
            }
        }

        public string ParentFormText
        {
            get
            {
                return "Create MySql Connection";
            }
        }

        public string SelectionListItemText
        {
            get
            {
                return "MySql...";
            }
        }

        #endregion

        #region methods

        private string SetConnectionString()
        {
            string user = LoginNameTextBox.Text;
            string pwd = PasswordTextBox.Text;
            string server = ServerNameTextBox.Text;
            string db = DatabaseNameTextBox.Text;

            string conString = "server=" + server + ";database=" + db + ";user id="
                + user + ";password="
                + pwd + ";pooling=false";

            return conString;
        }

        #endregion
    }
}
```