

ADO Connection String Samples

This page contains sample ADO connection strings for ODBC DSN / DSN-Less, OLE DB Providers, Remote Data Services (RDS), MS Remote, and MS DataShape.

Also included are ADO.NET connection strings for MySQL, ODBC, OLE DB, Oracle, and SQL Server .NET Data Providers.

These sample connection strings are compiled by Carl Prothman, a [Microsoft ASP.NET MVP](#) and [Microsoft Certified Professional \(MCP\)](#)

If you have an ADO or ADO.NET connection string that is not listed below, or you see an connection string that does not have the correct setting, please send an email to [Carl Prothman](#). Thanks!

Table of Contents

- **[.NET Data Provider Connections](#)**
 - [MySQL .NET Native Provider](#)
 - [ODBC .NET Data Provider](#)
 - [OLE DB .NET Data Provider](#)
 - [Oracle .NET Data Provider](#)
 - [SQL Server .NET Data Provider](#)
- **[ODBC DSN Connections](#)**
 - [DSN](#)
 - [File DSN](#)
- **[ODBC DSN-Less Connections](#)**
 - [ODBC Driver for Access](#)
 - [ODBC Driver for AS/400](#)
 - [ODBC Driver for dBASE](#)
 - [ODBC Driver for Excel](#)
 - [ODBC Driver for Informix](#)
 - [ODBC Driver for Interbase \(from Easysoft\)](#)
 - [ODBC Driver for Interbase \(from InterSolv\)](#)
 - [ODBC Driver for Lotus Notes](#)
 - [ODBC Driver for MySQL](#)
 - [ODBC Driver for Oracle \(from Microsoft\)](#)
 - [ODBC Driver for Oracle \(from Oracle\)](#)
 - [ODBC Driver for Paradox](#)
 - [ODBC Driver for SQL Server](#)
 - [ODBC Driver for Sybase](#)
 - [ODBC Driver for Sybase SQL Anywhere](#)
 - [ODBC Driver for Text](#)
 - [ODBC Driver for Teradata](#)
 - [ODBC Driver for Visual FoxPro](#)
- **[OLE DB Data Link Connections](#)**
 - [Data Link File \(UDL\)](#)

- **OLE DB Data Provider Connections**
 - [OLE DB Provider for Active Directory Service](#)
 - [OLE DB Provider for Advantage](#)
 - [OLE DB Provider for AS/400 \(from IBM\)](#)
 - [OLE DB Provider for AS/400 and VSAM \(from Microsoft\)](#)
 - [OLE DB Provider for Commerce Server](#)
 - [OLE DB Provider for DB2](#)
 - [OLE DB Provider for DTS Packages](#)
 - [OLE DB Provider for Exchange](#)
 - [OLE DB Provider for Excel](#)
 - [OLD DB Provider for Internet Publishing](#)
 - [OLE DB Provider for Index Server](#)
 - [OLE DB Provider for Microsoft Jet](#)
 - [OLE DB Provider for Microsoft Project](#)
 - [OLE DB Provider for MySQL](#)
 - [OLE DB Provider for ODBC Databases](#)
 - [OLE DB Provider for OLAP Services](#)
 - [OLE DB Provider for Oracle \(from Microsoft\)](#)
 - [OLE DB Provider for Oracle \(from Oracle\)](#)
 - [OLE DB Provider for Pervasive](#)
 - [OLE DB Provider for Simple Provider](#)
 - [OLE DB Provider for SQLBase](#)
 - [OLE DB Provider for SQL Server](#)
 - [OLE DB Provider for SQL Server via SQLXMOLEDB](#)
 - [OLE DB Provider for Sybase Adaptive Server Anywhere](#)
 - [OLE DB Provider for Sybase Adaptive Server Enterprise](#)
 - [OLE DB Provider for Text Files](#)
 - [OLE DB Provider for UniData and UniVerse](#)
 - [OLE DB Provider for Visual FoxPro](#)
 - **ADO URL Connections**
 - [ADO Recordset](#)
 - **Remote Data Service (RDS) Connections**
 - [RDS Data Control - Connect Property](#)
 - [RDS Data Control - URL Property](#)
 - **MS Remote Provider Connections**
 - [MS Remote - Access \(Jet\)](#)
 - [MS Remote - SQL Server](#)
 - **Data Shape Provider Connections**
 - [MS DataShape - SQL Server](#)
-

ODBC DSN Connections

Using an ODBC DSN (Data Source Name) is a two step process.

1) You must first create the DSN via the "ODBC Data Source Administrator" program found in your computer's Control Panel (or Administrative Tools menu in Windows 2000). Make sure to create a

SYSTEM DSN (not a USER DSN) when using ASP. You can also [create](#) the DSN via Visual Basic code.

2) Then use the following connection string - with your own DSN name of course.

- **ODBC - DSN**

```
oConn.Open "DSN=mySystemDSN;" & _  
           "Uid=myUsername;" & _  
           "Pwd=myPassword"
```

- **ODBC - File DSN**

```
oConn.Open "FILEDSN=c:\somepath\mydb.dsn;" & _  
           "Uid=myUsername;" & _  
           "Pwd=myPassword"
```

For more information, see: [About ODBC data sources](#) and [How to Use File DSNs and DSN-less Connections](#)

Note: The problem with DSN is that Users can (and will) modify or delete them by mistake, then your program won't work so well. So it's better to use a DSN-Less or OLE DB Provider connection string - with a Trusted Connection if possible!

ODBC DSN-Less Connections

- **ODBC Driver for Access**

For Standard Security:

```
oConn.Open "Driver={Microsoft Access Driver (*.mdb)};" & _  
           "Dbq=c:\somepath\mydb.mdb;" & _  
           "Uid=admin;" & _  
           "Pwd="
```

If you are using a Workgroup (System database):

```
oConn.Open "Driver={Microsoft Access Driver (*.mdb)};" & _  
           "Dbq=c:\somepath\mydb.mdb;" & _  
           "SystemDB=c:\somepath\mydb.mdw;", _  
           "myUsername", "myPassword"
```

If want to open up the MDB exclusively

```
oConn.Open "Driver={Microsoft Access Driver (*.mdb)};" & _  
           "Dbq=c:\somepath\mydb.mdb;" & _
```

```
"Exclusive=1;" & _
"Uid=admin;" & _
"Pwd="
```

If MDB is located on a Network Share

```
oConn.Open "Driver={Microsoft Access Driver (*.mdb)};" & _
"Dbq=\\myServer\myShare\myPath\myDb.mdb;" & _
"Uid=admin;" & _
"Pwd="
```

If MDB is located on a remote machine

- Or use an XML Web Service via [SOAP Toolkit](#) or [ASP.NET](#)
- Or upgrade to SQL Server and use an [IP connection string](#)
- Or use an [ADO URL](#) with a remote ASP web page
- Or use a [MS Remote](#) or [RDS](#) connection string

If you don't know the path to the MDB (using ASP)

```
<% ' ASP server-side code
oConn.Open "Driver={Microsoft Access Driver (*.mdb)};" & _
"Dbq=" & Server.MapPath(".") & "\myDb.mdb;" & _
"Uid=admin;" & _
"Pwd="
%>
```

This assumes the MDB is in the same directory where the ASP page is running. Also make sure this directory has Write permissions for the user account.

If you don't know the path to the MDB (using VB)

```
oConn.Open "Driver={Microsoft Access Driver (*.mdb)};" & _
"Dbq=" & App.Path & "\myDb.mdb;" & _
"Uid=admin;" & _
"Pwd="
```

This assumes the MDB is in the same directory where the application is running.

For more information, see: [Microsoft Access Driver Programming Considerations](#)

To view Microsoft KB articles related to Microsoft Access Driver, [click here](#)

• ODBC Driver for AS/400 (from IBM)

```
oConn.Open "Driver={Client Access ODBC Driver (32-bit)};" & _
           "System=myAS400;" & _
           "Uid=myUsername;" & _
           "Pwd=myPassword"
```

For more information, see: [A Fast Path to AS/400 Client/Server](#)

• ODBC Driver for dBASE

```
oConn.Open "Driver={Microsoft dBASE Driver (*.dbf)};" & _
           "DriverID=277;" & _
           "Dbq=c:\somepath"
```

Then specify the filename in the SQL statement:

```
oRs.Open "Select * From user.dbf", oConn, , ,adCmdText
```

Note: MDAC 2.1 (or greater) requires the Borland Database Engine (BDE) to [update](#) dBase DBF files. ([Q238431](#)).

For more information, see: [dBASE Driver Programming Considerations](#)

To view Microsoft KB articles related to Microsoft dBASE Driver, [click here](#)

• ODBC Driver for Excel

```
oConn.Open "Driver={Microsoft Excel Driver (*.xls)};" & _
           "DriverId=790;" & _
           "Dbq=c:\somepath\mySpreadsheet.xls;" & _
           "DefaultDir=c:\somepath"
```

For more information, see: [Microsoft Excel Driver Programming Considerations](#)

To view Microsoft KB articles related to Microsoft Excel Driver, [click here](#)

• ODBC Driver for Informix

If using INFORMIX 3.30 ODBC Driver

```
oConn.Open "Dsn='';" & _
           "Driver={INFORMIX 3.30 32 BIT};" & _
           "Host=myHostname;" & _
           "Server=myServerName;" & _
           "Service=myServiceName;" & _
           "Protocol=olsoctcp;" & _
           "Database=myDbName;" & _
```

```
"UID=myUsername;" & _
"PWD=myPassword" & _
```

' Or

```
oConn.Open "Dsn=myDsn;" & _
           "Host=myHostname;" & _
           "Server=myServerName;" & _
           "Service=myServiceName;" & _
           "Protocol=onsoctcp;" & _
           "Database=myDbName;" & _
           "Uid=myUsername;" & _
           "Pwd=myPassword"
```

If using INFORMIX-CLI 2.5 ODBC Driver

```
oConn.Open "Driver={Informix-CLI 2.5 (32 Bit)};" & _
           "Server=myServerName;" & _
           "Database=myDbName;" & _
           "Uid=myUsername;" & _
           "Pwd=myPassword" & _
```

For more information, see: [Informix Developer Zone](#), [Connection to ODBC Data Source](#),

• ODBC Driver for Interbase (from Easysoft)

For the local machine

```
oConn.Open "Driver={Easysoft IB6 ODBC};" & _
           "Server=localhost;" & _
           "Database=localhost:C:\Home\Data\Mydb.gdb;" & _
           "Uid=myUsername;" & _
           "Pwd=myPassword"
```

For a remote machine

```
oConn.Open "Driver={Easysoft IB6 ODBC};" & _
           "Server=myMachineName;" & _
           "Database=myMachineName:C:\Home\Data\Mydb.gdb;" & _
           "Uid=myUsername;" & _
           "Pwd=myPassword"
```

For more information, see: [Connecting to InterBase](#) and [Easysoft](#)

• ODBC Driver for Interbase (from InterSolv)

For the local machine

```
oConn.Open "Driver={INTERSQLV InterBase ODBC Driver (*.gdb)};" & _
           "Server=localhost;" & _
```

```
"Database=localhost:C:\Home\Data\Mydb.gdb;" & _  
"Uid=myUsername;" & _  
"Pwd=myPassword"
```

For a remote machine

```
oConn.Open "Driver={INTERSOLV InterBase ODBC Driver (*.gdb)};" & _  
"Server=myMachineName;" & _  
"Database=myMachineName:C:\Home\Data\Mydb.gdb;" & _  
"Uid=myUsername;" & _  
"Pwd=myPassword"
```

For more information, see: [Google Search](#) (if you know a direct URL [email me](#))

• ODBC Driver for Lotus Notes

```
oConn.Open "Driver={Lotus NotesSQL 3.01 (32-bit) ODBC DRIVER (*.nsf)};" & _  
"Server=myServerName;" & _  
"Database=mydir\myDbName.nsf;" & _  
"Uid=myUsername;" & _  
"Pwd=myPassword" & _
```

For more information, see: [Connection keywords](#)

• ODBC Driver for MySQL (via MyODBC)

To connect to a local database

```
oConn.Open "Driver={mysql};" & _  
"Server=MyServerName;" & _  
"Option=16834;" & _  
"Database=mydb"
```

To connect to a remote database

```
oConn.Open "Driver={mysql};" & _  
"Server=db1.database.com;" & _  
"Port=3306;" & _  
"Option=131072;" & _  
"Stmt=;" & _  
"Database=mydb;" & _  
"Uid=myUsername;" & _  
"Pwd=myPassword"
```

For more information, see: [Programs Known to Work with MyODBC](#)

- **ODBC Driver for Oracle (from Microsoft)**

For the current Oracle ODBC Driver from Microsoft

```
oConn.Open "Driver={Microsoft ODBC for Oracle};" & _  
           "Server=OracleServer.world;" & _  
           "Uid=myUsername;" & _  
           "Pwd=myPassword"
```

For the older Oracle ODBC Driver from Microsoft

```
oConn.Open "Driver={Microsoft ODBC Driver for Oracle};" & _  
           "ConnectionString=OracleServer.world;" & _  
           "Uid=myUsername;" & _  
           "Pwd=myPassword"
```

For more information, see: [Connection String Format and Attributes](#)

To view Microsoft KB articles related to Microsoft ODBC for Oracle, [click here](#)

- **ODBC Driver for Oracle (from Oracle)**

```
oConn.Open "Driver={Oracle ODBC Driver};" & _  
           "Dbq=myDBName;" & _  
           "Uid=myUsername;" & _  
           "Pwd=myPassword"
```

Where: The DBQ name must be defined in the tnsnames.ora file

For more information, see: [Oracle8 ODBC Driver Help](#), [Oracle ODBC FAQs](#), [\[asporacle\] listserv FAQs](#), and [ASPDB Oracle](#)

- **ODBC Driver for Paradox**

```
oConn.Open "Driver={Microsoft Paradox Driver (*.db )};" & _  
           "DriverID=538;" & _  
           "Fil=Paradox 5.X;" & _  
           "DefaultDir=c:\dbpath\" & _  
           "Dbq=c:\dbpath\" & _  
           "CollatingSequence=ASCII"
```

Note: MDAC 2.1 (or greater) requires the Borland Database Engine (BDE) to [update](#) Paradox ISAM fDBF files. ([Q230126](#)).

Note: There is an extra space after "db" in the Paradox Driver name

For more information, see: [Paradox Driver Programming Considerations](#)

To view Microsoft KB articles related to Microsoft Paradox Driver, [click here](#)

- **ODBC Driver for SQL Server**

For Standard Security

```
oConn.Open "Driver={SQL Server};" & _  
           "Server=MyServerName;" & _  
           "Database=myDatabaseName;" & _  
           "Uid=myUsername;" & _  
           "Pwd=myPassword"
```

For Trusted Connection security

```
oConn.Open "Driver={SQL Server};" & _  
           "Server=MyServerName;" & _  
           "Database=myDatabaseName;" & _  
           "Uid=" & _  
           "Pwd="
```

' or

```
oConn.Open "Driver={SQL Server};" & _  
           "Server=MyServerName;" & _  
           "Database=myDatabaseName;" & _  
           "Trusted_Connection=yes"
```

To Prompt user for username and password

```
oConn.Properties("Prompt") = adPromptAlways  
oConn.Open "Driver={SQL Server};" & _  
           "Server=MyServerName;" & _  
           "DataBase=myDatabaseName"
```

To connect to SQL Server running on the same computer

```
oConn.Open "Driver={SQL Server};" & _  
           "Server=(local);" & _  
           "Database=myDatabaseName;" & _  
           "Uid=myUsername;" & _  
           "Pwd=myPassword"
```

To connect to SQL Server running on a remote computer (via an IP address)

```
oConn.Open "Driver={SQL Server};" & _  
           "Server=xxx.xxx.xxx.xxx;" & _  
           "Address=xxx.xxx.xxx.xxx,1433;" & _  
           "Network=DBMSSOCN;" & _
```

```
"Database=myDatabaseName;" & _  
"Uid=myUsername;" & _  
"Pwd=myPassword"
```

Where:

- xxx.xxx.xxx.xxx is an IP address
- 1433 is the default port number for SQL Server.
- "Network=DBMSSOCN" tells ODBC to use TCP/IP rather than Named Pipes ([Q238949](#))

For more information, see: [SQLDriverConnect \(ODBC\)](#)

To view Microsoft KB articles related to ODBC Driver for SQL Server, [click here](#)

- **ODBC Driver for Sybase**

If using the Sybase System 12 (or 12.5) Enterprise Open Client ODBC Driver

```
oConn.Open "Driver={SYBASE ASE ODBC Driver};" & _  
"Srvr=myServerName;" & _  
"Uid=myUsername;" & _  
"Pwd=myPassword"
```

If using the Sybase System 11 ODBC Driver

```
oConn.Open "Driver={SYBASE SYSTEM 11};" & _  
"Srvr=myServerName;" & _  
"Uid=myUsername;" & _  
"Pwd=myPassword"
```

If using the Intersolv 3.10 Sybase ODBC Driver

```
oConn.Open "Driver={INTERSOLV 3.10 32-BIT Sybase};" & _  
"Srvr=myServerName;" & _  
"Uid=myUsername;" & _  
"Pwd=myPassword"
```

For more information, see: [Sybase System 10 ODBC Driver Reference Guide](#)

To view Microsoft KB articles related to ODBC Driver for Sybase, [click here](#)

- **ODBC Driver for Sybase SQL Anywhere**

```
oConn.Open "ODBC; Driver=Sybase SQL Anywhere 5.0;" & _
    "DefaultDir=c:\dbpath\" & _
    "Dbf=c:\sqlany50\mydb.db;" & _
    "Uid=myUsername;" & _
    "Pwd=myPassword;" & _
    "Dsn="""
```

Note: Including the DSN tag with a null string is absolutely critical or else you get the dreaded -7778 error.

For more information, see: [Sybase SQL Anywhere User Guide](#)

• ODBC Driver for Teradata

```
oConn.Open "Provider=Teradata;" & _
    "DBCName=MyDbcName;" & _
    "Database=MyDatabaseName;" & _
    "Uid=myUsername;" & _
    "Pwd=myPassword"
```

For more information, see [Teradata ODBC Driver](#)

• ODBC Driver for Text

```
oConn.Open _
    "Driver={Microsoft Text Driver (*.txt; *.csv)};" & _
    "Dbq=c:\somepath\" & _
    "Extensions=asc, csv, tab, txt"
```

Then specify the filename in the SQL statement:

```
oRs.Open "Select * From customer.csv", _
    oConn, adOpenStatic, adLockReadOnly, adCmdText
```

Note: If you are using a Tab delimited file, then make sure you create a [schema.ini](#) file, and include the "Format=TabDelimited" option.

For more information, see: [Text File Driver Programming Considerations](#)

To view Microsoft KB articles related to Microsoft Text Driver, [click here](#)

• ODBC Driver for Visual FoxPro

With a database container

```
oConn.Open "Driver={Microsoft Visual FoxPro Driver};" & _
    "SourceType=DBC;" & _
    "SourceDB=c:\somepath\mySourceDb.dbc;" & _
```

```
"Exclusive=No"
```

Without a database container (Free Table Directory)

```
oConn.Open "Driver={Microsoft Visual FoxPro Driver};" & _  
           "SourceType=DBF;" & _  
           "SourceDB=c:\somepath\mySourceDbFolder;" & _  
           "Exclusive=No"
```

For more information, see: [Visual FoxPro ODBC Driver](#) and [Q165492](#)

To view Microsoft KB articles related to ODBC Driver for Visual FoxPro, [click here](#)

OLE DB Data Link Connections

- **Data Link File (UDL)**

For Absolute Path

```
oConn.Open "File Name=c:\somepath\myDatabaseName.udl"
```

For Relative Path

```
oConn.Open "File Name=myDatabaseName.udl"
```

For more information, see: [HOWTO: Use Data Link Files with ADO](#)

Note: Windows 2000 no longer contains the "New | Microsoft Data Link" menu anymore. You can add the Data Link menu back in the menu list by running the "[C:\Program Files\Common Files\System\Ole DB\newudl.reg](#)" reg file, then right-click on the desktop and select "New | Microsoft Data Link" menu.

Or you can also create a Data Link file by creating a text file and change it's file extension to ".udl", then double-click the file.

To view Microsoft KB articles related to Data Link File, [click here](#)

OLE DB Provider Connections

- **OLE DB Provider for Active Directory Service**

```
oConn.Open "Provider=ADSDS00bject;" & _  
           "User Id=myUsername;" & _  
           "Password=myPassword"
```

For more information, see: [Microsoft OLE DB Provider for Microsoft Active Directory Service](#)

To view Microsoft KB articles related to Data Link File, [click here](#)

- **OLE DB Provider for Advantage**

```
oConn.Open "Provider=Advantage OLE DB Provider;" & _  
           "Data source=c:\myDbfTableDir;" & _  
           "ServerType=ADS_LOCAL_SERVER;" & _  
           "TableType=ADS_CDX"
```

For more information, see: [Advantage OLE DB Provider \(for ADO\)](#)

- **OLE DB Provider for AS/400 (from IBM)**

```
oConn.Open "Provider=IBMDA400;" & _  
           "Data source=myAS400;" & _  
           "User Id=myUsername;" & _  
           "Password=myPassword"
```

For more information, see: [A Fast Path to AS/400 Client/Server](#)

- **OLE DB Provider for AS/400 and VSAM (from Microsoft)**

```
oConn.Open "Provider=SNAOLEDB;" & _  
           "Data source=myAS400;" & _  
           "User Id=myUsername;" & _  
           "Password=myPassword"
```

For more information, see: [Connection](#) and [ConnectionString Property](#)

To view Microsoft KB articles related to OLE DB Provider for AS/400 and VSAM, [click here](#)

- **OLE DB Provider for Commerce Server**

For Data Warehouse

```
oConn.Open "Provider=Commerce.DSO.1;" & _
    "Data Source=miscop://InProcConn/Server=mySrvName:" & _
    "Catalog=DWSchema:Database=myDBname:" & _
    "User=myUsername:Password=myPassword:" & _
    "FastLoad=True"

' Or

oConn.Open "URL=miscop://InProcConn/Server=myServerName:" & _
    "Database=myDBname:Catalog=DWSchema:" & _
    "User=myUsername:Password=myPassword:" & _
    "FastLoad=True"
```

For Profiling System

```
oConn.Open "Provider=Commerce.DSO.1;" & _
    "Data Source=miscop://InProcConn/Server=mySrvName:" & _
    "Catalog=Profile Definitions:Database=myDBname:" & _
    "User=myUsername:Password=myPassword"

' Or

oConn.Open _
    "URL=miscop://InProcConnect/Server=myServerName:" & _
    "Database=myDBname:Catalog=Profile Definitions:" & _
    "User=myUsername:Password=myPassword"
```

For more information, see: [OLE DB Provider for Commerce Server, DataWarehouse](#), and [Profiling System](#)

To view Microsoft KB articles related to OLE DB Provider for Commerce Server, [click here](#)

- **OLE DB Provider for DB2 (from Microsoft)**

For TCP/IP connections

```
oConn.Open = "Provider=DB2OLEDB;" & _
    "Network Transport Library=TCPIP;" & _
    "Network Address=xxx.xxx.xxx.xxx;" & _
    "Initial Catalog=MyCatalog;" & _
    "Package Collection=MyPackageCollection;" & _
    "Default Schema=MySchema;" & _
    "User ID=MyUsername;" & _
    "Password=MyPassword"
```

For APPC connections

```
oConn.Open = "Provider=DB2OLEDB;" & _
    "APPC Local LU Alias=MyLocalLUAlias;" & _
```

```
"APPC Remote LU Alias=MyRemoteLUAlias;" & _  
"Initial Catalog=MyCatalog;" & _  
"Package Collection=MyPackageCollection;" & _  
"Default Schema=MySchema;" & _  
"User ID=MyUsername;" & _  
"Password=MyPassword"
```

For more information, see: [Connection](#), [ConnectionString Property](#), and [Q218590](#)

To view Microsoft KB articles related to OLE DB Provider for DB2, [click here](#)

• OLE DB Provider for DTS Packages

The Microsoft OLE DB Provider for DTS Packages is a read-only provider that exposes Data Transformation Services Package Data Source Objects.

```
oConn.Open = "Provider=DTSPackageDSO;" & _  
"Data Source=mydatasource"
```

For more information, see: [OLE DB Providers Tested with SQL Server](#)

To view Microsoft KB articles related to OLE DB Provider for DTS Packages, [click here](#)

• OLE DB Provider for Exchange

```
oConn.Provider = "EXOLEDB.DataSource"  
oConn.Open = "http://myServerName/myVirtualRootName"
```

For more information, see: [Exchange OLE DB Provider](#), [Messaging](#), [Calendaring](#), [Contacts](#), and [Exchange using ADO objects](#)

To view Microsoft KB articles related to OLE DB Provider for Exchange, [click here](#)

• OLE DB Provider for Excel

Actually there is no OLE DB Provider for Excel. However, you can use the [OLE DB Provider for JET](#) to read and write data in Microsoft Excel workbooks. Or you can use the [ODBC Driver for Excel](#).

• OLE DB Provider for Index Server

```
oConn.Open "Provider=MSIDXS;" & _  
           "Data source=MyCatalog"
```

For more information, see: [Microsoft OLE DB Provider for Microsoft Indexing Service](#)

To view Microsoft KB articles related to OLE DB Provider for Index Server, [click here](#)

• OLE DB Provider for Internet Publishing

```
oConn.Open "Provider=MSDAIPP.DSO;" & _  
           "Data Source=http://mywebsite/myDir;" & _  
           "User Id=myUsername;" & _  
           "Password=myPassword"
```

' Or

```
oConn.Open "URL=http://mywebsite/myDir;" & _  
           "User Id=myUsername;" & _  
           "Password=myPassword"
```

For more information, see: [Microsoft OLE DB Provider for Internet Publishing](#) and [Q245359](#)

To view Microsoft KB articles related to OLE DB Provider for Internet Publishing, [click here](#)

• OLE DB Provider for Microsoft Jet

For standard security

```
oConn.Open "Provider=Microsoft.Jet.OLEDB.4.0;" & _  
           "Data Source=c:\somepath\myDb.mdb;" & _  
           "User Id=admin;" & _  
           "Password="
```

If using a Workgroup (System Database)

```
oConn.Open "Provider=Microsoft.Jet.OLEDB.4.0;" & _  
           "Data Source=c:\somepath\mydb.mdb;" & _  
           "Jet OLEDB:System Database=MySystem.mdw", _  
           "myUsername", "myPassword"
```

Note, remember to convert both the MDB and the MDW to the 4.0 database format when using the 4.0 OLE DB Provider.

If MDB has a database password

```
oConn.Open "Provider=Microsoft.Jet.OLEDB.4.0;" & _
           "Data Source=c:\somepath\mydb.mdb;" & _
           "Jet OLEDB:Database Password=MyDbPassword", _
           "myUsername", "myPassword"
```

If want to open up the MDB exclusively

```
oConn.Mode = adModeShareExclusive
oConn.Open "Provider=Microsoft.Jet.OLEDB.4.0;" & _
           "Data Source=c:\somepath\myDb.mdb;" & _
           "User Id=admin;" & _
           "Password="
```

If MDB is located on a network share

```
oConn.Open "Provider=Microsoft.Jet.OLEDB.4.0;" & _
           "Data Source=\\myServer\myShare\myPath\myDb.mdb"
```

If MDB is located on a remote machine

- Or use an XML Web Service via [SOAP Toolkit](#) or [ASP.NET](#)
- Or upgrade to SQL Server and use an [IP connection string](#)
- Or use an [ADO URL](#) with a remote ASP web page
- Or use a [MS Remote](#) or [RDS](#) connection string

If you don't know the path to the MDB (using ASP)

```
<% ' ASP server-side code
oConn.Open "Provider=Microsoft.Jet.OLEDB.4.0;" & _
           "Data Source=" & Server.MapPath(".") & "\myDb.mdb;" & _
           "User Id=admin;" & _
           "Password="
%>
```

This assumes the MDB is in the same directory where the ASP page is running. Also make sure this directory has Write permissions for the user account.

If you don't know the path to the MDB (using VB)

```
oConn.Open "Provider=Microsoft.Jet.OLEDB.4.0;" & _
           "Data Source=" & App.Path & "\myDb.mdb;" & _
           "User Id=admin;" & _
           "Password="
```

This assumes the MDB is in the same directory where the application is running.

For more information, see: [OLE DB Provider for Microsoft Jet, Q191754](#), and [Q225048](#)

Note: [Microsoft.Jet.OLEDB.3.51](#) only gets installed by MDAC 2.0.
[Q197902](#)

Note: MDAC 2.6 and 2.7 do not contain any of the JET components.
[Q271908](#) and [Q239114](#)

To view Microsoft KB articles related to OLE DB Provider for Microsoft JET, [click here](#)

You can also open an Excel Spreadsheet using the JET OLE DB Provider

```
oConn.Open "Provider=Microsoft.Jet.OLEDB.4.0;" & _  
           "Data Source=c:\somepath\mySpreadsheet.xls;" & _  
           "Extended Properties=(""Excel 8.0;HDR=Yes"")"
```

Where "HDR=Yes" means that there is a header row in the cell range (or named range), so the provider will not include the first row of the selection into the recordset. If "HDR=No", then the provider will include the first row of the cell range (or named range) into the recordset.

For more information, see: [Q278973](#)

You can also open a Text file using the JET OLE DB Provider

```
oConn.Open "Provider=Microsoft.Jet.OLEDB.4.0;" & _  
           "Data Source=c:\somepath\" & _  
           "Extended Properties=(""text;HDR=Yes;FMT=Delimited"")"
```

'Then open a recordset based on a select on the actual file

```
oRs.Open "Select * From MyTextFile.txt", oConn, _  
        adOpenStatic, adLockReadOnly, adCmdText
```

For more information, see: [Q262537](#)

• OLE DB Provider for Microsoft Project

```
oConn.Open "Provider=Microsoft.Project.OLEDB.9.0;" & _  
           "Project Name=c:\somepath\myProject.mpp"
```

For more information, see: [Microsoft Project 2000 OLE DB Provider Information](#)

To view Microsoft KB articles related to OLE DB Provider for Microsoft

Project, [click here](#)

- **OLE DB Provider for mySQL**

```
oConn.Open "Provider=MySQLProv;" & _  
           "Data Source=mysqlDB;" & _  
           "User Id=myUsername;" & _  
           "Password=myPassword"
```

For more information, see: [API - OLE DB](#), [SWSOft](#), and [Snippet](#)

- **OLE DB Provider for ODBC Databases**

WARNING: This OLE DB Provider is considered [obsolete](#) by Microsoft

For Access (Jet)

```
oConn.Open "Provider=MSDASQL;" & _  
           "Driver={Microsoft Access Driver (*.mdb)};" & _  
           "Dbq=c:\somepath\mydb.mdb;" & _  
           "Uid=myUsername;" & _  
           "Pwd=myPassword"
```

For SQL Server

```
oConn.Open "Provider=MSDASQL;" & _  
           "Driver={SQL Server};" & _  
           "Server=myServerName;" & _  
           "Database=myDatabaseName;" & _  
           "Uid=myUsername;" & _  
           "Pwd=myPassword"
```

For more information, see: [Microsoft OLE DB Provider for ODBC](#)

To view Microsoft KB articles related to OLE DB Provider for ODBC, [click here](#)

- **OLE DB Provider for OLAP Services**

Microsoft OLE DB for Online Analytical Processing (OLAP) is a set of objects and interfaces that extends the ability of OLE DB to provide access to multidimensional data stores.

For ADOMD.Catalog

```
oCat.ActiveConnection = _  
    "Provider=MSOLAP;" & _
```

```
"Data Source=myOLAPServerName;" & _
"Initial Catalog=myOLAPDatabaseName"
```

For ADOMD.Catalog (with URL)

```
oCat.ActiveConnection = _
    "Provider=MSOLAP;" & _
    "Data Source=http://myServerName/;" & _
    "Initial Catalog=myOLAPDatabaseName"
```

For Excel PivotTable

```
With ActiveWorkbook.PivotCaches.Add(SourceType:=xlExternal)
    .Connection = "OLEDB;" & _
        "Provider=MSOLAP;" & _
        "Location=myServerDataLocation;" & _
        "Initial Catalog=myOLAPDatabaseName"
    .MaintainConnection = True
    .CreatePivotTable TableDestination:=Range("A1"), _
        TableName:= "MyPivotTableName"
End With
```

For more information, see: [OLE DB for OLAP](#), [Catalog Object](#), [PivotTable](#), [Connecting Using HTTP](#)

To view Microsoft KB articles related to OLE DB Provider for OLAP Services, [click here](#)

• OLE DB Provider for Oracle (from Microsoft)

```
oConn.Open "Provider=msdaora;" & _
    "Data Source=MyOracleDB;" & _
    "User Id=myUsername;" & _
    "Password=myPassword"
```

For more information, see: [Microsoft OLE DB Provider for Oracle](#)

To view Microsoft KB articles related to OLE DB Provider for Oracle, [click here](#)

• OLE DB Provider for Oracle (from Oracle)

For Standard Security

```
oConn.Open "Provider=OraOLEDB.Oracle;" & _
    "Data Source=MyOracleDB;" & _
    "User Id=myUsername;" & _
    "Password=myPassword"
```

For a Trusted Connection

```
oConn.Open "Provider=OraOLEDB.Oracle;" & _
           "Data Source=MyOracleDB;" & _
           "User Id=/" & _
           "Password="
' Or
```

```
oConn.Open "Provider=OraOLEDB.Oracle;" & _
           "Data Source=MyOracleDB;" & _
           "OSAuthent=1"
```

Note: "Data Source=" must be set to the appropriate Net8 name which is known to the naming method in use. For example, for Local Naming, it is the alias in the tnsnames.ora file; for Oracle Names, it is the Net8 Service Name.

For more information, see: [Oracle Provider for OLE DB Developer's Guide](#)

- **OLE DB Provider for Pervasive**

```
oConn.Open "Provider=PervasiveOLEDB;" & _
           "Data Source=C:\PervasiveEB"
```

For more information, see: [OLE DB - ADO](#)

- **OLE DB Provider for Simple Provider**

The Microsoft OLE DB Simple Provider (OSP) allows ADO to access any data for which a provider has been written using the [OLE DB Simple Provider Toolkit](#). Simple providers are intended to access data sources that require only fundamental OLE DB support, such as in-memory arrays or XML documents.

OSP in MDAC 2.6 has been enhanced to support opening hierarchical ADO Recordsets over arbitrary XML files. These XML files may contain the ADO XML persistence schema, but it is not required. This has been implemented by connecting the OSP to the MSXML2.DLL, therefore MSXML2.DLL or newer is required.

```
oConn.Open "Provider=MSDAOSP;" & _
           "Data Source=MSXML2.DSOControl.2.6"
```

```
oRS.Open "http://WebServer/VirtualRoot/MyXMLFile.xml", oConn
```

For more information, see: [Microsoft OLE DB Simple Provider](#) and [Q272270](#)

To view Microsoft KB articles related to OLE DB Provider for Simple Provider, [click here](#)

- **OLE DB Provider for SQLBase**

```
oConn.Open "Provider=SQLBaseOLEDB;" & _
           "Data source=mySybaseServer;" & _
           "Location=mySybaseDB;" & _
           "User Id=myUserName;" & _
           "Password=myUserPassword"
```

For more information, see: [Books on-line](#) There is a one-time free sign-up, then select "SQLBase OLE DB Data Provider User's Guide for v7.5 (20-6220-0001)", then download the zip file and extract the document.

- **OLE DB Provider for SQL Server**

For Standard Security

```
oConn.Open "Provider=sqloledb;" & _
           "Data Source=myServerName;" & _
           "Initial Catalog=myDatabaseName;" & _
           "User Id=myUsername;" & _
           "Password=myPassword"
```

For a Trusted Connection

```
oConn.Open "Provider=sqloledb;" & _
           "Data Source=myServerName;" & _
           "Initial Catalog=myDatabaseName;" & _
           "Integrated Security=SSPI"
```

To connect to a "Named Instance"

```
oConn.Open "Provider=sqloledb;" & _
           "Data Source=myServerName\myInstanceName;" & _
           "Initial Catalog=myDatabaseName;" & _
           "User Id=myUsername;" & _
           "Password=myPassword"
```

Note: In order to connect to a SQL Server 2000 "named instance", you must have MDAC 2.6 (or greater) installed.

To Prompt user for username and password

```
oConn.Provider = "sqloledb"
oConn.Properties("Prompt") = adPromptAlways
oConn.Open "Data Source=myServerName;" & _
           "Initial Catalog=myDatabaseName"
```

To connect to SQL Server running on the same computer

```
oConn.Open "Provider=sqloledb;" & _
           "Data Source=(local);" & _
           "Initial Catalog=myDatabaseName;" & _
           "User ID=myUsername;" & _
           "Password=myPassword"
```

To connect to SQL Server running on a remote computer (via an IP address)

```
oConn.Open "Provider=sqloledb;" & _
           "Network Library=DBMSSOCN;" & _
           "Data Source=xxx.xxx.xxx.xxx,1433;" & _
           "Initial Catalog=myDatabaseName;" & _
           "User ID=myUsername;" & _
           "Password=myPassword"
```

Where:

- "Network Library=DBMSSOCN" tells OLE DB to use TCP/IP rather than Named Pipes ([Q238949](#))
- xxx.xxx.xxx.xxx is an IP address
- 1433 is the default port number for SQL Server. [Q269882](#) and [Q287932](#)
- You can also add "Encrypt=yes" for [encryption](#)

For more information, see: [Microsoft OLE DB Provider for SQL Server](#)

To view Microsoft KB articles related to OLE DB Provider for SQL Server, [click here](#)

• **OLE DB Provider for SQL Server via SQLXMLEDB**

The SQLXMLEDB provider is an OLE DB provider that exposes the Microsoft SQLXML functionality through ADO. The SQLXMLEDB provider is not a rowset provider; it can only execute commands in the "write to an output stream" mode of ADO.

```
oConn.Open "Provider=SQLXMLEDB.3.0;" & _
           "Data Provider=SQLOLEDB;" & _
           "Data Source=mySqlServerName;" & _
           "Initial Catalog=myDatabaseName;" & _
           "User Id=myUserName;" & _
           "Password=myUserPassword"
```

For more information, see: [SQLXML 3.0](#) and [A Survey of Microsoft SQL Server 2000 XML Features](#)

To view Microsoft KB articles related to OLE DB Provider for SQL Server via SQLXMLEDB, [click here](#)

- **OLE DB Provider for Sybase Adaptive Server Anywhere (ASA)**

```
oConn.Open "Provider=ASAProv;" & _  
           "Data source=myASA"
```

For more information, see: [ASA Programming Interfaces Guide](#) and [ASA User's Guide](#)

- **OLE DB Provider for Sybase Adaptive Server Enterprise (ASE)**

```
oConn.Open "Provider=Sybase ASE OLE DB Provider;" & _  
           "Data source=myASEServer"
```

' or

```
oConn.Open "Provider=Sybase.ASEOLEDBProvider;" & _  
           "Srvr=myASEServer,5000;" & _  
           "Catalog=myDBName;" & _  
           "User Id=myUserName;" & _  
           "Password=myUserPassword"
```

Where:

- The Sybase ASE OLE DB provider from the Sybase 12.5 client CD
- 5000 is the port number for Sybase.

Note: The Open Client 12 Sybase OLE DB Provider fails to work without creating a Data Source .IDS file using the Sybase Data Administrator. These .IDS files resemble ODBC DSNs.

Note: With Open Client 12.5, the server port number feature finally works, allowing fully qualified network connection strings to be used without defining any .IDS Data Source files.

For more information, see: [Sybase Advance Search](#)

- **OLE DB Provider for Text Files**

Actually there is no OLE DB Provider for Text files. However, you can use the [OLE DB Provider for JET](#) to read and write data in Text files. Or you can use the [ODBC Driver for Text](#).

- **OLE DB Provider for UniData and UniVerse**

```
oConn.Open "Provider=Ardent.UniOLEDB;" & _  
           "Data source=myServer;" & _  
           "Location=myDatabase;" & _  
           "User ID=myUsername;" & _  
           "Password=myPassword"
```

For more information, see: [Ardent Using UniOLEDB 5.1](#), [Informix Using UniOLEDB 5.2](#)

- **OLE DB Provider for Visual FoxPro**

```
oConn.Open "Provider=vfpoledb;" & _  
           "Data Source=C:\vfp7\Samples\Data\myVFPDB.dbc;" & _  
           "Mode=ReadWrite|Share Deny None;" & _  
           "Collating Sequence=MACHINE;" & _  
           "Password=''"
```

For more information, see: [Microsoft OLE DB Provider for Visual FoxPro](#)

To view Microsoft KB articles related to OLE DB Provider for Visual FoxPro, [click here](#).

Note: The Visual FoxPro OLE DB Provider is **NOT** installed by MDAC 2.x. You must install [Visual FoxPro 7.0](#) in order to get it's OLE DB Provider.

Remote Data Service (RDS) Connections

The following examples show how to connect to a remote database using the RDS Data Control. When using the RDS DataControl's Server/Connect / SQL properties, the RDS DataControl uses the RDS DataFactory on the remote server. If you use the RDS DataControl's [URL property](#), then the RDS DataFactory is not used at all.

WARNING: The RDS DataFactory can be a major security hole if not setup and configured correctly! For more information, see [RDS FAQ #24](#)

WARNING: RDS is considered [obsolete](#) by Microsoft

- **RDS DataControl - Connect Property**

With the RDS default handler disabled

```

With oRdc
    .Server = "http://myServerName"
    .Sql = "Select * From Authors Where State = 'CA'"
    .Connect = "Provider=sqloledb;" & _
               "Data Source=(local);" & _
               "Initial Catalog=pubs;" & _
               "User Id=myUsername;" & _
               "Password=myPassword"
    .Refresh
End With

```

With the RDS default handler enabled

```

With oRdc
    .Server = "http://myServerName"
    .Handler = "MSDFMAP.Handler"
    .Connect = "Data Source=MyConnectTag;"
    .Sql = "MySQLTag('CA')"
    .Refresh
End With

```

The corresponding CONNECT and SQL sections in the default handler \WINNT\MSDFMAP.INI file would be:

```

[connect MyConnectTag]
Access = ReadWrite
Connect = "Provider=sqloledb;
          Data Source=(local);
          Initial Catalog=pubs;
          User Id=sa;
          Password="                (put all of this on single line!)

[sql MySQLTag]
Sql = "Select * From Authors Where State = '?'"

```

For more information about the RDS Default Handler, see: [Q243245](#), [Q230680](#), and [RDS Customization Handler](#) Microsoft articles

To view Microsoft KB articles related to RDS, [click here](#)

- **RDS DataControl - URL Property**

To get records from a remote database

```

With oRdc
    .URL = "http://myServerName/AuthorsGet.asp?state=CA"
    .Refresh
End With

```

To save, set the URL property to an ASP web page

```
With oRdc
    .URL = "http://myServerName/AuthorsSave.asp"
    .SubmitChanges
End With
```

Note: You must use MDAC 2.5 (or greater) for this feature

For more information, see: [RDS URL Property](#)

To view Microsoft KB articles related to RDS, [click here](#)

MS Remote Provider Connections

The following connections strings use Microsoft's remote provider (MS Remote). The MS Remote provider tells ADO to communicate with the remote server (via the RDS DataFactory) and to use the remote provider that is installed on the remote server.

WARNING: The RDS DataFactory can be a major security hole if not setup and configured correctly! For more information, see [RDS FAQ #24](#)

WARNING: RDS is considered [obsolete](#) by Microsoft

- **MS Remote - Access (Jet)**

If you want to use an ODBC DSN on the remote machine

```
oConn.Open "Provider=MS Remote;" & _
    "Remote Server=http://myServerName;" & _
    "Remote Provider=MSDASQL;" & _
    "DSN=AdvWorks;" & _
    "Uid=myUsername;" & _
    "Pwd=myPassword"
```

If you want to use an OLE DB Provider on the remote machine

```
oConn.Open "Provider=MS Remote;" & _
    "Remote Server=http://myServerName;" & _
    "Remote Provider=Microsoft.Jet.OLEDB.4.0;" & _
    "Data Source=c:\somepath\mydb.mdb", _
    "admin", ""
```

If you want to use an OLE DB Provider on the remote machine

```
oConn.Open "Provider=MS Remote;" & _
           "Remote Server=http://myServerName;" & _
           "Handler=MSDFMAP.Handler;" & _
           "Data Source=MyAdvworksConn"
```

The corresponding entry in the [\winnt\Msdfmap.ini](#) file would be:

```
[connect MyAdvworksConn]
Access = ReadWrite
Connect = "Provider=Microsoft.Jet.OLEDB.4.0;
          Data Source=mydb.mdb;
          User Id=admin;
          Password="                (put all of this on single line!)
```

- **MS Remote - SQL Server**

If you want to use an ODBC DSN on the remote machine

```
oConn.Open "Provider=MS Remote;" & _
           "Remote Server=http://myServerName;" & _
           "Remote Provider=MSDASQL;" & _
           "DSN=myDatabaseName;" & _
           "Uid=myUsername;" & _
           "Pwd=myPassword"
```

If you want to use an OLE DB Provider on the remote machine

```
oConn.Open "Provider=MS Remote;" & _
           "Remote Server=http://myServerName;" & _
           "Remote Provider=SQLOLEDB;" & _
           "Data Source=myServerName;" & _
           "Initial Catalog=myDatabaseName;" & _
           "User ID=myUsername;" & _
           "Password=myPassword"
```

If you want to use an OLE DB Provider on the remote machine

```
oConn.Open "Provider=MS Remote;" & _
           "Remote Server=http://myServerName;" & _
           "Handler=MSDFMAP.Handler;" & _
           "Data Source=MyPubsConn"
```

The corresponding entry in the [\winnt\Msdfmap.ini](#) file would be:

```
[connect MyPubsConn]
Access = ReadWrite
Connect = "Provider=SQLOLEDB;
          Data Source=myServerName;
          Initial Catalog=myDatabaseName;
          User ID=myUsername;
          Password=myPassword"      (put all of this on single line!)
```

For more information, see: [Microsoft OLE DB Remoting Provider](#) and [Q240838](#)

To view Microsoft KB articles related to MS Remote, [click here](#)

To view Microsoft KB articles related to RDS, [click here](#)

ADO URL Connections

ADO 2.5+ allows you to open up a Recordset based on XML returned from an ASP file over HTTP. This feature doesn't use RDS at all.

- **ADO Recordset**

To get records from a remote database

```
oRs.Open "http://myServer/AuthorsGetByState.asp?state=CA",, _  
        adOpenStatic, adLockBatchOptimistic
```

To save changes

```
' Save Recordset into Stream  
Set oStm = New ADODB.Stream  
oRs.Save oStm, adPersistXML  
  
' Use MSXML's XMLHTTP object to open ASP  
Set oXMLHTTP = New MSXML2.XMLHTTP30  
oXMLHTTP.Open "POST", "http://myServerName/AuthorsSave.asp"  
oXMLHTTP.Send oStm.ReadText  
  
' If an error occurred  
If oXMLHTTP.Status = 500 Then  
    Debug.Print oXMLHTTP.statusText  
End If
```

For more information, see: [ADO Recordset's Open Method](#)

.NET Data Provider Connections

- **SQL Server .NET Data Provider (System.Data.SqlClient)**

The SQL Server .NET Data Provide allows you to connect to a Microsoft

SQL Server 7.0 or 2000 databases.

For Microsoft SQL Server 6.5 or earlier, use the [OLE DB .NET Data Provider](#) with the "SQL Server OLE DB Provider" (SQLOLEDB).

Note: The SQL Server .NET Data Provider knows which Provider it is. Hence the "provider=" part of the connection string is not needed.

Using C#:

```
using System.Data.SqlClient;
...
SqlConnection oSQLConn = new SqlConnection();
oSQLConn.ConnectionString = "Data Source=(local);" +
                           "Initial Catalog=mySQLServerDBName;" +
                           "Integrated Security=yes";

oSQLConn.Open();
```

Using VB.NET:

```
Imports System.Data.SqlClient
...
Dim oSQLConn As SqlConnection = New SqlConnection()
oSQLConn.ConnectionString = "Data Source=(local);" & _
                           "Initial Catalog=mySQLServerDBName;" & _
                           "Integrated Security=yes"

oSQLConn.Open()
```

If connection to a remote server (via IP address):

```
oSQLConn.ConnectionString = "Network Library=DBMSSOCN;" & _
                           "Data Source=xxx.xxx.xxx.xxx,1433;" & _
                           "Initial Catalog=mySQLServerDBName;" & _
                           "User ID=myUsername;" & _
                           "Password=myPassword"
```

Where:

- "Network Library=[DBMSSOCN](#)" tells SqlConnection to use TCP/IP [Q238949](#)
- [xxx.xxx.xxx.xxx](#) is an IP address.
- 1433 is the default port number for SQL Server. [Q269882](#) and [Q287932](#)
- You can also add "[Encrypt=yes](#)" for [encryption](#)

For more information, see: [System.Data.SQL Namespace](#), [Q308656](#), and [.NET Data Providers](#)

Note: Microsoft [SQLXML Managed Classes](#) exposes the functionality of SQLXML inside the Microsoft .NET Framework.

To view Microsoft KB articles related to SqlConnection, [click here](#)

- **OLE DB .NET Data Provider (System.Data.OleDb)**

The OLE DB .NET Data Provider uses native OLE DB through COM interop to enable data access.

To use the OLE DB .NET Data Provider, you must also use an OLE DB provider (e.g. SQLOLEDB, MSDAORA, or Microsoft.JET.OLEDB.4.0).

For IBM AS/400 OLE DB Provider

```
' VB.NET
Dim oOleDbConnection As OleDb.OleDbConnection
Dim sConnString As String = _
    "Provider=IBMDA400.DataSource.1;" & _
    "Data source=myAS400DbName;" & _
    "User Id=myUsername;" & _
    "Password=myPassword"
oOleDbConnection = New OleDb.OleDbConnection(sConnString)
oOleDbConnection.Open()
```

For JET OLE DB Provider

```
' VB.NET
Dim oOleDbConnection As OleDb.OleDbConnection
Dim sConnString As String = _
    "Provider=Microsoft.Jet.OLEDB.4.0;" & _
    "Data Source=C:\myPath\myJet.mdb;" & _
    "User ID=Admin;" & _
    "Password="
oOleDbConnection = New OleDb.OleDbConnection(sConnString)
oOleDbConnection.Open()
```

For Oracle OLE DB Provider

```
' VB.NET
Dim oOleDbConnection As OleDb.OleDbConnection
Dim sConnString As String = _
    "Provider=OraOLEDB.Oracle;" & _
    "Data Source=MyOracleDB;" & _
    "User ID=myUsername;" & _
    "Password=myPassword"
oOleDbConnection = New OleDb.OleDbConnection(sConnString)
oOleDbConnection.Open()
```

For SQL Server OLE DB Provider

```
' VB.NET
Dim oOleDbConnection As OleDb.OleDbConnection
Dim sConnString As String = _
    "Provider=sqloledb;" & _
    "Data Source=myServerName;" & _
```

```

        "Initial Catalog=myDatabaseName;" & _
        "User Id=myUsername;" & _
        "Password=myPassword"
oOleDbConnection = New OleDb.OleDbConnection(sConnString)
oOleDbConnection.Open()

```

For Sybase ASE OLE DB Provider

```

' VB.NET
Dim oOleDbConnection As OleDb.OleDbConnection
Dim sConnString As String = _
    "Provider=Sybase ASE OLE DB Provider;" & _
    "Data Source=MyDataSourceName;" & _
    "Server Name=MyServerName;" & _
    "Database=MyDatabaseName;" & _
    "User ID=myUsername;" & _
    "Password=myPassword"
oOleDbConnection = New OleDb.OleDbConnection(sConnString)
oOleDbConnection.Open()

```

For more information, see: [System.Data.OleDb Namespace](#) and [.NET Data Providers](#)

To view Microsoft KB articles related to OleDbConnection, [click here](#)

• ODBC .NET Data Provider (System.Data.ODBC)

The ODBC .NET Data Provider is an [add-on component](#) to the .NET Framework SDK. It provides access to native ODBC drivers the same way the OLE DB .NET Data Provider provides access to native OLE DB providers.

For SQL Server ODBC Driver

```

' VB.NET
Dim oODBCConnection As Odbc.OdbcConnection
Dim sConnString As String = _
    "Driver={SQL Server};" & _
    "Server=MySQLServerName;" & _
    "Database=MyDatabaseName;" & _
    "Uid=MyUsername;" & _
    "Pwd=MyPassword"
oODBCConnection = New Odbc.OdbcConnection(sConnString)
oODBCConnection.Open()

```

For Oracle ODBC Driver

```

' VB.NET
Dim oODBCConnection As Odbc.OdbcConnection
Dim sConnString As String = _
    "Driver={Microsoft ODBC for Oracle};" & _
    "Server=OracleServer.world;" & _

```

```

        "Uid=myUsername;" & _
        "Pwd=myPassword"
oODBCConnection = New Odbc.OdbcConnection(sConnString)
oODBCConnection.Open()

```

For Access (JET) ODBC Driver

```

' VB.NET
Dim oODBCConnection As Odbc.OdbcConnection
Dim sConnString As String = _
    "Driver={Microsoft Access Driver (*.mdb)};" & _
    "Dbq=c:\somepath\mydb.mdb;" & _
    "Uid=Admin;" & _
    "Pwd="
oODBCConnection = New Odbc.OdbcConnection(sConnString)
oODBCConnection.Open()

```

For Sybase System 11 ODBC Driver

```

// C#
string myConnStr = "Driver={Sybase System 11};" +
    "SRVR=mySybaseServerName;" +
    "DB=myDatabaseName;" +
    "UID=myUsername;" +
    "PWD=myPassword";
OdbcConnection myConnection = new OdbcConnection(myConnStr);
myConnection.Open();

```

For all other ODBC Drivers

```

' VB.NET
Dim oODBCConnection As Odbc.OdbcConnection
Dim sConnString As String = "Dsn=myDsn;" & _
    "Uid=myUsername;" & _
    "Pwd=myPassword"
oODBCConnection = New Odbc.OdbcConnection(sConnString)
oODBCConnection.Open()

```

For more information, see: [ODBC .Net Data Provider](#)

To view Microsoft KB articles related to OdbcConnection, [click here](#)

- **.NET Framework Data Provider for Oracle
(System.Data.OracleClient)**

The .NET Framework Data Provider for Oracle is an [add-on component](#) to the .NET Framework that provides access to an Oracle database using the Oracle Call Interface (OCI) as provided by Oracle Client software.

Using C#:

```
using System.Data.OracleClient;

OracleConnection oOracleConn = new OracleConnection();
oOracleConn.ConnectionString = "Data Source=Oracle8i;" +
                             "Integrated Security=yes";
oOracleConn.Open();
```

Using VB.NET:

```
Imports System.Data.OracleClient

Dim oOracleConn As OracleConnection = New OracleConnection()
oOracleConn.ConnectionString = "Data Source=Oracle8i;" & _
                             "Integrated Security=yes";
oOracleConn.Open()
```

Note: You must have the Oracle 8i Release 3 (8.1.7) Client or later installed in order for this provider to work correctly.

Note: You must have the RTM version of the .NET Framework installed in order for this provider to work correctly.

Note: There are known Oracle 7.3, Oracle 8.0, and Oracle9i client and server problems in this beta release. The server-side issues should be resolved in the final release of the product. However, Oracle 7.3 client will not be supported.

For more information, see: [.NET Data Provider for Oracle Beta 1](#)

To view Microsoft KB articles related to OracleConnection, [click here](#)

• MySQL .NET Native Provider

The MySQL .NET Native Provider is an [add-on component](#) to the .NET Framework that allows you to access the MySQL database through the native protocol, without going through OLE DB.

Using C#

```
using EID.MySqlClient;

MySQLConnection oMySQLConn = new MySqlConnection();
oMySQLConn.ConnectionString = "Data Source=localhost;" +
                             "Database=mysqlDatabase;" +
                             "User ID=myUsername;" +
                             "Password=myPassword;" +
                             "Command Logging=false";

oMySQLConn.Open();
```

Using VB.NET

```
Imports EID.MySqlClient

Dim oMySQLConn As MySqlConnection = New MySqlConnection()
oMySQLConn.ConnectionString = "Data Source=localhost;" & _
    "Database=mySQLDatabase;" & _
    "User ID=myUsername;" & _
    "Password=myPassword;" & _
    "Command Logging=false"

oMySQLConn.Open()
```

For more information, see: [EID's MySQL ADO.NET native provider](#)