# XOGetProperty Function

## Description

Retrieves a Connection Object property; the Connection Object is specified by the passed handle.

## Syntax

flag = **XOGetProperty**(hXO, property, value, arg)

## Parameters

The XOGetProperty function has the following parameters:

| Parameter | Description |
|---|---|
| HXO | Handle to the Connection Object returned by XOInstance. |
| Property | See properties below. |
| Value | Property specific; see properties below. |
| Arg | Property specific; see Properties below. |

| Parameter | Description | ***hXO*** **[in]** | ***value*** **[out]** | ***arg*** **[in]** |
|---|---|---|---|---|
| XO_DSLIST$ | Returns a list of the DataSets that use the specified Connection. | Connection Object handle. | @VM-delimited list of DataSet handles. | |
| XO_FORMATDATETIME$ | Returns the format control string for date, time, and date/time types. The format control string uses the C runtime library's strftime() notation with one addition: "%f" denotes a decimal point and up to four digits of the fractional portion of seconds, based on the precision of the time or date/time value. For more information on the format control strings, see the notes in the insert DSXO_API. | | An @fm-delimited string containing the date format, the time format, and the date/time format in strftime() notation. | |
| XO_INTERNALHANDLES$ | returns the data source specific handle for the connection. | | | |
| XO_FORMATDATETIME$ | Returns the format control string for date, time, and date/time types. The format control string uses the C runtime library's strftime() notation with one addition: "%f" denotes a decimal point and up to four digits of the fractional portion of seconds, based on the precision of the time or date/time value. For more information on the format control strings, see the notes in the insert DSXO_API. | | An @fm-delimited string containing the date format, the time format, and the date/time format in strftime() notation. | |
| XO_INTERNALHANDLES$ | returns the data source specific handle for the connection. | Connection Object handle. | if applicable, the internal handle used by the connection object to communicate with the data source. For example, the HCON value for ODBC. | |
| XO_NULLVALUE$ | Returns the literal value used to specify NULLs for the database. Usually, this is the literal NULL, although for data source which may be supported in the future, it may differ. | | The literal value used to specify NULL values. | |
| XO_OPTIONS$ | returns the connection options used to create the connection. | Connection Object handle. | Connection options used by XOInstance(). | |
| XO_QRYLIST$ | returns a list of the Queries that use the specified Connection. | Connection Object handle. | @VM-delimited list of Query handles. | |
| XO_QUOTECHAR$ | Returns the character used to quote values in scripts. Usually this is the single quote ('), although for data source which may be supported in the future, it may differ. | | The character used to quote values in scripts that are sent to the database. | |
| XO_QUOTEDQUOTE$ | Returns the escaped quote character. In other words, if the character returned by the XO_QUOTECHAR$ property is within a value that must be quoted, the XO_QUOTEDQUOTE$ property returns the value which the quote character must be changed to. For example, ODBC requires that the string "Sam's Bar and Grill" be passed as "'Sam*s Bar and Grill'".* | | The escaped quote character, (the quote character as it appears when it is quoted). | |

| | | | | |
|---|---|---|---|---|
| XO_SCOPE$ | returns the scope used to create the connection. | Connection Object handle. | The connection scope. (See XOInstance.) | |
| XO_SOURCENAME$ | returns the Data Source Name used to open the Connection. | | | |
| XO_SOURCETYPE$ | returns the Data Source Type of the Connection Object. | Connection Object handle | Data Source Type. | |
| XO_TABLEDESCRIPT$ | returns the column structure of the specified table. | Connection Object handle | Name, type, precision, scale, nullable, OI type for each column; attributes are @FM-delimited and columns are @VM-delimited. | Table name. |
| XO_TABLELIST$ | returns list of tables (data objects). | Connection Object handle. | @FM-delimited list of tables. | Table types. |
| XO_TRANSACTIONS$ | Checks if transaction processing is enabled. | Connection Object handle. | True if transactions specified on connection. | |
| XO_TYPEQUOTED$ | Returns information on which types are quoted and which types are not for a specific connection. Value is returned as an @fm-delimited array of boolean values specifying which types are quoted. The positions in the Value array are defined by the DT_ equates in the insert record DSXO_API. For example, Value<DT_DATETIME$> is true if date/time values are quoted when they are sent to the database as part of a script. | An @fm-delimited array of boolean values specifying which types are quoted. The types are defined in the insert DSXO_API and are all prefixed with "DT_". For example, if Value<DT_DATETIME$> is true, then date/time values are quoted when they are sent to the database in a script. | | |
| XO_VALID$ | checks if the passed Connection Object handle is valid. | Connection Object handle. | TRUE$ if the passed Connection Object handle is valid. | |

## Returns

TRUE$ for success or FALSE$ for failure.

## See also

XOSetProperty()