MBS REALbasic SQL Plugin Documentation

Christian Schmitz

January 17, 2011

0.1 Introduction

This is the PDF version of the documentation for the REALbasic Plug-in from Monkeybread Software Germany. Plugin part: MBS REALbasic SQL Plugin

0.2 Content

Chapter 1

List of Topics

Chapter 2

SQL

2.1 class SQLParamMBS

class SQLParamMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The SQL class for parameters.
Notes: Subclass of the SQLValueMBS class.

2.1.1 Methods

Name as string

method from class SQLParamMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The name of the parameter.

Option(name as string) as string

method from class SQLParamMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The string value of a specific parameter option.
Notes:

see also:
http://www.sqlapi.com/OnLineDoc/Param_ Option.html
(Read and Write computed property)

ParamDirType as integer

method from class SQLParamMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The direction type of parameter (input, output, etc.).
Notes:

Use the kParamDirType* constants.

Usually the Library automatically detects parameter's direction type and implicitly creates an appropriate SAParam object. But not all of DBMS clients/servers provide complete parameters information. In that situation programmer need to describe parameter's direction type explicitly. See Server specific notes for details.

http://www.sqlapi.com/OnLineDoc/Param_ ParamDirType.html
(Read and Write computed property)

ParamNativeType as integer

method from class SQLParamMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The native type code of the parameter.
Notes: (Read and Write computed property)

ParamPrecision as integer

method from class SQLParamMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The precision of the parameter value (the total number of allowable digits).
Notes: (Read and Write computed property)

ParamScale as integer

method from class SQLParamMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The scale of the parameter value (the number of digits to the right of the decimal point).
Notes: (Read and Write computed property)

ParamSize as integer

method from class SQLParamMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The parameter's data size.
Notes: (Read and Write computed property)

ParamType as integer

method from class SQLParamMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The parameter's data type.
Notes:

See the kDataType constants.
(Read and Write computed property)

ReadLongOrLob(consumer as SQLDataConsumerMBS, BlockSize as integer)

method from class SQLParamMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The Long or Lob data reading mode.
Notes:

SQLAPI++ Library provides two ways to read Long or BLob(CLob) object's value (usually SQLField or SQLParam objects):

1. reading of Long or Lob data at once into an internal buffer (like ordinary string or binary values);
2. piecewise reading of Long or Lob data using user defined callback.
kLongOrLobReaderDefault reading mode used by default.

If you want to control piecewise reading of Long or BLob(CLob) data you should set LongOrLobReaderMode and use kLongOrLobReaderManual reading mode for Long or BLob(CLob) parameters or fields you want to process with your data consumer. After that each fetch will skip reading Long and BLob(CLob) parameters that you set to be read manually. To read field or parameter defined to be read manually you should call ReadLongOrLob method for each of them after the fetch. ReadLongOrLob method will repeatedly call the data consumer Write event.

2.1.2 Constants

kParamDirTypeInput=0

const from class SQLParamMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the parameter direction type constants.
Notes: Input parameter.

kParamDirTypeInputOutput=1

const from class SQLParamMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the parameter direction type constants.
Notes: Input/output parameter.

kParamDirTypeOutput=2

const from class SQLParamMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the parameter direction type constants.
Notes: Output parameter.

kParamDirTypeReturn=3

const from class SQLParamMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the parameter direction type constants.
Notes: Returning parameter.

2.2 class SQLNumericMBS

class SQLNumericMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The class for numeric values.

2.2.1 Methods

Constructor

method from class SQLNumericMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates an empty numeric object.
See also:

- 2.2.1 Constructor(value as double) 18
- 2.2.1 Constructor(value as string) 18

Constructor(value as double)

method from class SQLNumericMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new numeric object based on the given double value.
See also:

- 2.2.1 Constructor 18
- 2.2.1 Constructor(value as string) 18

Constructor(value as string)

method from class SQLNumericMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new numeric object based on the given string.
See also:

- 2.2.1 Constructor 18
- 2.2.1 Constructor(value as double) 18

DoubleValue as double

method from class SQLNumericMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The double value for this number.

Int64Value as Int64

method from class SQLNumericMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The number value as an int64.

precision as integer

method from class SQLNumericMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The maximum number of digits in base 10.

scale as integer

method from class SQLNumericMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The number of digits to the right of the decimal point.

sign as integer

method from class SQLNumericMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The sign: 1 for positive numbers, 0 for negative numbers.

StringValue as string

method from class SQLNumericMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The string value of this number.

UInt64Value as UInt64

method from class SQLNumericMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The number value as an uint64.

2.3 class SQLNullMBS

class SQLNullMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The class used internally for null values.

2.4 class SQLLongOrLobMBS

class SQLLongOrLobMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The super class for Long Binary/Text and BLOB/CLOB classes.
Notes: Subclass of the SQLStringMBS class.

2.5 class SQLNotInitializedExceptionMBS

class SQLNotInitializedExceptionMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The exception raised if you call a method on an object which was not properly initialized.
Notes: Subclass of the SQLErrorExceptionMBS class.

2.6 class SQLStringMBS

class SQLStringMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The class for strings in this plugin.
Example:

```
dim s as new SQLStringMBS("Hello ")

MsgBox "Characters: "+str(s.GetLength)+" Bytes: "+str(s.GetBinaryLength)

dim a as string= s.CopyBinaryData
dim b as string= s.CopyText

MsgBox a // RB shows garbage as it tries to display bytes as UTF8 which does not work
MsgBox b // displays correct
```

Notes: A string can be text (with text encoding) or bytes (raw binary data).

2.6.1 Methods

Compare(text as SQLStringMBS) as integer

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Compares this string object with another string.
Notes:

Function performs a case-sensitive comparison of the strings, and is not affected by locale.

Returns zero if the strings are identical, <0 if this string object is less than text, or >0 if this string object is greater than text.
See also:

- 2.6.1 Compare(text as string) as integer 22

Compare(text as string) as integer

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Compares this string object with another string.
Notes:

Function performs a case-sensitive comparison of the strings, and is not affected by locale.

Returns zero if the strings are identical, <0 if this string object is less than text, or >0 if this string object is greater than text.
See also:

- 2.6.1 Compare(text as SQLStringMBS) as integer 22

CompareNoCase(text as SQLStringMBS) as integer

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Compares this string object with another string.
Notes:

Function performs a case-insensitive comparison of the strings, and is not affected by locale.

Returns zero if the strings are identical (ignoring case), <0 if this string object is less than text (ignoring case), or >0 if this string object is greater than text (ignoring case).
See also:

CompareNoCase(text as string) as integer

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Compares this string object with another string.
Notes:

Function performs a case-insensitive comparison of the strings, and is not affected by locale.

Returns zero if the strings are identical (ignoring case), <0 if this string object is less than text (ignoring case), or >0 if this string object is greater than text (ignoring case).
See also:

Constructor

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new empty string object.
See also:

Constructor(Data as string, isText as boolean=true)

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new string object with data or text copied from the data string.
Notes: If isText is true, the data is interpreted as text and string encoding conversion may modify it. If isText is false the bytes are copied raw.
See also:

- 2.6.1 Constructor 23
- 2.6.1 Constructor(other as SQLStringMBS) 24

Constructor(other as SQLStringMBS)

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new string object with data copied from the other string object.
See also:

- 2.6.1 Constructor 23
- 2.6.1 Constructor(Data as string, isText as boolean=true) 24

CopyBinaryData as string

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Copies the bytes from the internal buffer ignoring any text encoding.

CopyText as string

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Copies the characters of this string as text.
Notes: Text encoding conversion may happen.

Empty

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Forces a string to have 0 length.

GetBinaryLength as UInt32

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns a count of the bytes in the binary data buffer.

GetLength as UInt32

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the number of characters in a SAString object.
Notes: For multibyte character sets, GetLength counts each 8-bit character; that is, a lead and trail byte in one multibyte character are counted as two bytes.

IsEmpty as boolean

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Tests whether a String object contains no characters.
Notes: Returns true if length is zero.

Left(count as integer) as SQLStringMBS

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Extracts the left part of a string.

MakeLower

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Changes all characters in the string to lower case.

MakeUpper

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Changes all characters in the string to upper case.

Mid(first as integer) as SQLStringMBS

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Extracts the middle part of a string.
Notes: first: The zero-based index of the first character in this string object that is to be included in the extracted substring.
See also:

- 2.6.1 Mid(first as integer, Count as integer) as SQLStringMBS 27

Mid(first as integer, Count as integer) as SQLStringMBS

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Extracts the middle part of a string.
Notes:

first: The zero-based index of the first character in this string object that is to be included in the extracted substring.
count: The number of characters to extract from this string object. If this parameter is not supplied, then the remainder of the string is extracted.
See also:

- 2.6.1 Mid(first as integer) as SQLStringMBS 26

Right(count as integer) as SQLStringMBS

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Extracts the right part of a string.

TrimLeft

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Trim leading whitespace characters from the string.

TrimRight

method from class SQLStringMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Trim trailing whitespace characters from the string.

2.7 class SQLPositionMBS

class SQLPositionMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The class for a position value.

2.7.1 Methods

Constructor(withID as integer)

method from class SQLPositionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new Position value with an ID.
See also:

- 2.7.1 Constructor(withName as string) 28

Constructor(withName as string)

method from class SQLPositionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new Position value with a name.
See also:

- 2.7.1 Constructor(withID as integer) 28

2.8 class SQLIntervalMBS

class SQLIntervalMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The class in the SQL Plugin for an interval.

2.8.1 Methods

Constructor

method from class SQLIntervalMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new zero interval.
See also:

- 2.8.1 Constructor(days as integer, hours as integer, minutes as integer, seconds as integer) 29
- 2.8.1 Constructor(value as double) 30

Constructor(days as integer, hours as integer, minutes as integer, seconds as integer)

method from class SQLIntervalMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new interval with the given values.
See also:

- 2.8.1 Constructor 29
- 2.8.1 Constructor(value as double) 30

Constructor(value as double)

method from class SQLIntervalMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new interval with the given time delta.
Example:

```
dim n as new SQLIntervalMBS(5)

MsgBox n.StringValue // shows "120:00:00" for 120 hours
```

See also:

- 2.8.1 Constructor 29
- 2.8.1 Constructor(days as integer, hours as integer, minutes as integer, seconds as integer) 29

Dec(interval as SQLIntervalMBS)

method from class SQLIntervalMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Decrements the interval.

DoubleValue as double

method from class SQLIntervalMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The value of this interval.
Example:

```
dim n as new SQLIntervalMBS(1,2,3,4)

MsgBox str(n.DoubleValue) // shows "1.085463" for 1 day, 2 hours, 3 minutes and 4 seconds
```

GetDays as integer

method from class SQLIntervalMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The days in this interval.

GetHours as integer

method from class SQLIntervalMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The hours value.

GetMinutes as integer

method from class SQLIntervalMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The minutes value.

GetSeconds as integer

method from class SQLIntervalMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The seconds value.

GetTotalDays as double

method from class SQLIntervalMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The total days value.
Example:

```
dim n as new SQLIntervalMBS(1,2,3,4)

MsgBox str(n.GetTotalDays) // shows "1"
```

GetTotalHours as double

method from class SQLIntervalMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The total hours value.
Example:

```
dim n as new SQLIntervalMBS(1,2,3,4)

MsgBox str(n.GetTotalHours) // shows "26"
```

GetTotalMinutes as double

method from class SQLIntervalMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The total minutes value.
Example:

```
dim n as new SQLIntervalMBS(1,2,3,4)

MsgBox str(n.GetTotalMinutes) // shows "1563"
```

GetTotalSeconds as double

method from class SQLIntervalMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The total seconds value.
Notes:

dim n as new SQLIntervalMBS(1,2,3,4)

MsgBox str(n.GetTotalSeconds) // shows "93784"

Inc(interval as SQLIntervalMBS)

method from class SQLIntervalMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Increments the interval.

SetInterval(days as integer, hours as integer, minutes as integer, seconds as integer)

method from class SQLIntervalMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets the interval values.

StringValue as string

method from class SQLIntervalMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The interval as a string.
Example:

```
dim n as new SQLIntervalMBS(5)

MsgBox n.StringValue // shows "120:00:00" for 120 hours
```

2.9 class SQLGlobalsMBS

class SQLGlobalsMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The class for the global methods of the SQL plugin.

2.9.1 Methods

GetVersionBuild as integer

shared method from class SQLGlobalsMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The build number of the SQLAPI library.

GetVersionMajor as integer

shared method from class SQLGlobalsMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The major version number of the SQLAPI library.

GetVersionMinor as integer

shared method from class SQLGlobalsMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The minor version number of the SQLAPI library.

SetLicenseCode(n as string, enddate as integer, v1 as integer, v2 as integer)

shared method from class SQLGlobalsMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Registeres the SQL plugin and library.
Notes: Once you ordered a license, you receive details on how to call this method.

Setlocale(category as integer, locale as string)

shared method from class SQLGlobalsMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.5, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets the locale to use.
Notes:

The Setlocale function sets the C library's notion of natural language formatting style for particular sets of routines. Each such style is called a 'locale' and is invoked using an appropriate name passed as a C string.

The setlocale() function recognizes several categories of routines. These are the categories and the sets of routines they select:

LocaleAll	Set the entire locale generically.
LocaleCollate	Set a locale for string collation routines. This controls alphabetic ordering in strcoll() and strxfrm().
LocaleCType	Set a locale for the ctype and multibyte functions. This controls recognition of upper and lower case, alphabetic or non-alphabetic characters, and so on.
LocaleMessages	Set a locale for message catalogs, see catopen function.
LocaleMonetary	Set a locale for formatting monetary values; this affects the localeconv() function.
LocaleNumeric	Set a locale for formatting numbers. This controls the formatting of decimal points in input and output of floating point numbers in functions such as printf() and scanf(), as well as values returned by localeconv().
LocaleTime	Set a locale for formatting dates and times using the strftime() function.

Only three locales are defined by default: the empty string "" (which denotes the native environment) and the "C" and "POSIX" locales (which denote the C-language environment). By default, C programs start in the "C" locale.

2.9.2 Constants

LocaleAll=0

const from class SQLGlobalsMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the locale category constants for SetLocale.
Notes: Set the entire locale generically.

LocaleCollate=1

const from class SQLGlobalsMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the locale category constants for SetLocale.
Notes: Set a locale for string collation routines. This controls alphabetic ordering in strcoll() and strxfrm().

LocaleCType=2

const from class SQLGlobalsMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the locale category constants for SetLocale.
Notes: et a locale for the ctype(3) and multibyte(3) functions. This controls recognition of upper and lower case, alphabetic or non-alphabetic characters, and so on.

LocaleMessages=6

const from class SQLGlobalsMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the locale category constants for SetLocale.
Notes: Set a locale for message catalogs, see catopen(3) function.

LocaleMonetary=3

const from class SQLGlobalsMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the locale category constants for SetLocale.
Notes: Set a locale for formatting monetary values; this affects the localeconv() function.

LocaleNumeric=4

const from class SQLGlobalsMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the locale category constants for SetLocale.
Notes: Set a locale for formatting numbers. This controls the formatting of decimal points in input and output of floating point numbers in functions such as printf() and scanf(), as well as values returned by localeconv().

LocaleTime=5

const from class SQLGlobalsMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the locale category constants for SetLocale.
Notes: Set a locale for formatting dates and times using the strftime() function.

2.10 class SQLite3MBS

class SQLite3MBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.5, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The class for the native SQLite API.
Example:

```
dim con as new SQLConnectionMBS
dim path as string = "/tmp/test.db" // change path for Windows!

con.Connect(path,"","",SQLConnectionMBS.kSQLiteClient)

dim api as SQLAPIMBS = con.NativeAPI
if api isa SQLite3MBS then
dim s as SQLite3MBS = SQLite3MBS(api)
MsgBox s.Version
end if
```

Notes: Subclass of the SQLAPIMBS class.

2.10.1 Methods

Version as string

method from class SQLite3MBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.5, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The version string of the SQLite library.

VersionNumber as integer

method from class SQLite3MBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.5, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The version number of the SQLite library.

2.11 class SQLLongCharMBS

class SQLLongCharMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: A class for the long character data type.
Notes:

Basicly this is a SQLStringMBS which is always marked to contain text. You only need this class to use the constructor with dataprovider to stream data to the database.
Subclass of the SQLLongOrLobMBS class.

2.11.1 Methods

Constructor(data as SQLStringMBS)

method from class SQLLongCharMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new long character object from a string object.
See also:

Constructor(dataProvider as SQLDataProviderMBS, BlockSize as UInt32)

method from class SQLLongCharMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new long character object from a data provider.
Notes:

The blocksize specifies in which sizes data is requested from the provider.
You must make sure that the data provider and this new long character object life long enough. Because the actual data is requested later when you do the update on the database.

If BlockSize is 0, the default block size is used.
See also:

- 2.11.1 Constructor(data as SQLStringMBS) 39

2.12 class SQLLongBinaryMBS

class SQLLongBinaryMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: A class for a long binary object.
Notes:

Basicly this is a SQLStringMBS which is always marked to contain binary data. You only need this class to use the constructor with dataprovider to stream data to the database.
Subclass of the SQLLongOrLobMBS class.

2.12.1 Methods

Constructor(data as SQLStringMBS)

method from class SQLLongBinaryMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new long binary object from a string object.
See also:

- 2.12.1 Constructor(dataProvider as SQLDataProviderMBS, BlockSize as UInt32) 40

Constructor(dataProvider as SQLDataProviderMBS, BlockSize as UInt32)

method from class SQLLongBinaryMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new long binary object from a data provider.
Notes:

The blocksize specifies in which sizes data is requested from the provider.
You must make sure that the data provider and this new blob object life long enough. Because the actual data is requested later when you do the update on the database.

If BlockSize is 0, the default block size is used.
See also:

- 2.12.1 Constructor(data as SQLStringMBS) 40

2.13 class SQLValueReadMBS

class SQLValueReadMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The class used in the SQL Plugin for value objects which can be read.

2.13.1 Methods

asBLob as SQLStringMBS

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the value as BLob (SQLString) data.
Notes:

If the value of current object is NULL, asBLob method returns an empty string. Use isNull method to make sure if the value is NULL or not.

If the value's type of current object is string (kDataTypeString), bytes (kDataTypeBytes), long binary (kDataTypeLongBinary), long character (kDataTypeLongChar), BLob (kDataTypeBLob) or CLob (kDataType-

CLob), asBLob method returns the object's value as SQLString object.

If the value's type of current object is bool (kDataTypeBool), short (kDataTypeShort), long (kDataTypeLong), double (kDataTypeDouble), numeric (kDataTypeNumeric), date-time (kDataTypeDateTime) or cursor (kDataTypeCursor), the result is undefined and debug version asserts.

Use DataType method to get the value's type of SQLValueRead object.

asBool as boolean

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the value of current object as bool data.
Notes:

f the value of current object is NULL, asBool method returns false. Use isNull method to make sure if the value is NULL or not.

If the value's type of current object is bool (kDataTypeBool), asBool method returns the original value with no conversion.

If the value's type of current object is short (kDataTypeShort), long (kDataTypeLong) or double (kDataTypeDouble), asBool method converts it to bool data type. Conversion returns false if the value is 0; true otherwise.

If the value's type of current object is any data type except the described above, the result is undefined and debug version asserts.

Use DataType method to get the value's type of SQLValueRead object.

asBytes as SQLStringMBS

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the value as binary string data (SQLString).
Notes:

If the value of current object is NULL, asBytes method returns an empty string. Use isNull method to make sure if the value is NULL or not.

If the value's type of current object is string (kDataTypeString), bytes (kDataTypeBytes), long binary (kDataTypeLongBinary), long character (kDataTypeLongChar), BLob (kDataTypeBLob) or CLob (kDataTypeCLob), asBytes method returns the object's value as SQLString object.

If the value's type of current object is bool (kDataTypeBool), short (kDataTypeShort), long (kDataTypeLong), double (kDataTypeDouble), numeric (kDataTypeNumeric) or date-time (kDataTypeDateTime), asBytes method returns a block of data with size sizeof(value's type) as SQLString object.

If the value's type of current object is cursor (kDataTypeCursor), the result is undefined and debug version asserts.

Use DataType method to get the value's type of SQLValueRead object.

asCLob as SQLStringMBS

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the value as CLob (SQLString) data.
Notes:

If the value of current object is NULL, asCLob method returns an empty string. Use isNull method to make sure if the value is NULL or not.

If the value's type of current object is string (kDataTypeString), bytes (kDataTypeBytes), long binary (kDataTypeLongBinary), long character (kDataTypeLongChar), BLob (kDataTypeBLob) or CLob (kDataType-

CLob), asCLob method returns the object's value as SQLString object.

If the value's type of current object is bool (kDataTypeBool), short (kDataTypeShort), long (kDataTypeLong), double (kDataTypeDouble), numeric(kDataTypeNumeric), date-time (kDataTypeDateTime) or cursor (kDataTypeCursor), the result is undefined and debug version asserts.

Use DataType method to get the value's type of SQLValueRead object.

asDateTime as SQLDateTimeMBS

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: returns the value as SQLDateTime data.
Notes:

If the value of current object is NULL, asDateTime method returns an empty SQLDateTime object. Use isNull method to make sure if the value is NULL or not.

If the value's type of current object is kDataTypeDateTime, asDateTime method returns SQLDateTime object.

If the value's type of current object is any data type except kDataTypeDateTime, the result is undefined and debug version asserts.

Use DataType method to get the value's type of SQLValueRead object.

asDouble as double

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: returns the value as double data.
Notes:

If the value of current object is NULL, asDouble method returns 0. Use isNull method to make sure if the value is NULL or not.

If the value's type of current object is double (kDataTypeDouble), asDouble method returns the original value with no conversion.

If the value's type of current object is bool (kDataTypeBool), short (kDataTypeShort), long (kDataTypeLong) or numeric (kDataTypeNumeric), asDouble method converts it to double (kDataTypeDouble) data type.

If the value's type of current object is string (kDataTypeString), asDouble method tries to convert it to double value. If the conversion is possible and correct, asDouble returns kDataTypeDouble value. If conversion is incorrect asDouble method throws an exception.

If the value's type of current object is kDataTypeDateTime, asDouble method converts it to standard double representation. Days are represented by whole number increments starting with 30 December 1899, midnight as time zero. Hour values are expressed as the absolute value of the fractional part of the number. See SQLDateTime::operator double() for more detailes.

If the value's type of current object is any data type except the described above, the result is undefined and debug version asserts.

Use DataType method to get the value's type of SQLValueRead object.

asInterval as SQLIntervalMBS

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the value as interval (SQLIntervalMBS).

asLong as integer

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the value as long data.
Notes:

If the value of current object is NULL, asLong method returns 0. Use isNull method to make sure if the value is NULL or not.

If the value's type of current object is long (kDataTypeLong), asLong method returns the original value with no conversion.

If the value's type of current object is bool (kDataTypeBool), short (kDataTypeShort), double (kDataTypeDouble) or numeric (kDataTypeNumeric), asLong method converts it to long data type. Note, that in this case the returned value can be truncated.

If the value's type of current object is string (kDataTypeString), asLong method tries to convert it to long (kDataTypeLong) value. If the conversion is possible and correct, asLong returns kDataTypeLong value. If conversion is incorrect asLong method throws an exception.

If the value's type of current object is any data type except the described above, the result is undefined and debug version asserts.

Use DataType method to get the value's type of SQLValueRead object.

asLongBinary as SQLStringMBS

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the value as long binary (SQLString) data.
Notes:

If the value of current object is NULL, asLongBinary method returns an empty string. Use isNull method to make sure if the value is NULL or not.

If the value's type of current object is string (kDataTypeString), bytes (kDataTypeBytes), long binary (kDataTypeLongBinary), long character (kDataTypeLongChar), BLob (kDataTypeBLob) or CLob (kDataTypeCLob), asLongBinary method returns the object's value as SQLString object.

If the value's type of current object is bool (kDataTypeBool), short (kDataTypeShort), long (kDataTypeLong), double (kDataTypeDouble), numeric (kDataTypeNumeric), date-time (kDataTypeDateTime) or cursor (kDataTypeCursor), the result is undefined and debug version asserts.

Use DataType method to get the value's type of SQLValueRead object.

asLongChar as SQLStringMBS

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the value as long character (SQLString) data.
Notes:

If the value of current object is NULL, asLongChar method returns an empty string. Use isNull method to make sure if the value is NULL or not.

If the value's type of current object is string (kDataTypeString), bytes (kDataTypeBytes), long binary (kDataTypeLongBinary), long character (kDataTypeLongChar), BLob (kDataTypeBLob) or CLob (kDataTypeCLob), asLongChar method returns the object's value as SQLString object.

If the value's type of current object is bool (kDataTypeBool), short (kDataTypeShort), long (kDataTypeLong), double (kDataTypeDouble), numeric (kDataTypeNumeric), date-time (kDataTypeDateTime) or cursor (kDataTypeCursor), the result is undefined and debug version asserts.

Use DataType method to get the value's type of SQLValueRead object.

asNumeric as SQLNumericMBS

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the value as SQLNumeric data.
Notes:

If the value of current object is NULL, asNumeric method returns 0. Use isNull method to make sure if the value is NULL or not.

If the value's type of current object is exact numeric value (kDataTypeNumeric), asNumeric method returns the original value with no conversion.

If the value's type of current object is bool (kDataTypeBool), short (kDataTypeShort), double (kDataTypeDouble) or long (kDataTypeLong), asNumeric method converts it to SQLNumeric .

If the value's type of current object is string (kDataTypeString), asNumeric method tries to convert it from SQLChar* value. If the conversion is possible and correct, asNumeric converts to SQLNumeric from SQLChar* value. If conversion is incorrect asNumeric method throws an exception.

If the value's type of current object is any data type except the described above, the result is undefined and debug version asserts.

Use DataType method to get the value's type of SQLValueRead object.

asShort as Int16

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the value as short.
Notes:

If the value of current object is NULL, asShort method returns 0. Use isNull method to make sure if the value is NULL or not.

If the value's type of current object is short, asShort method returns the original value with no conversion.

If the value's type of current object is bool (kDataTypeBool), long (kDataTypeLong), unsigned long (SkDataTypeULong), double (kDataTypeDouble) or numeric (kDataTypeNumeric), asShort method converts it to short data type. Note, that in this case the returned value can be truncated.

If the value's type of current object is string (kDataTypeString), asShort method tries to convert it to short value. If the conversion is possible and correct, asShort returns the value. If conversion is incorrect asShort method throws an exception.

If the value's type of current object is any data type except the described above, the result is undefined and debug version asserts.

Use DataType method to get the value's type of SAValueRead object.

asString as SQLStringMBS

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: returns the value as string (SQLString) data.
Notes:

If the value of current object is NULL, asString method returns an empty string. Use isNull method to make sure if the value is NULL or not.

If the value's type of current object is bool (kDataTypeBool), asString method returns "true" or "false" string (SQLString object).

If the value's type of current object is short (kDataTypeShort), asString method converts it to decimal string (SQLString object) like function printf("% hd", ...) does.

If the value's type of current object is long (kDataTypeLong), asString method converts it to decimal string (SQLString object) like function printf("% ld", ...) does.

If the value's type of current object is double (kDataTypeDouble), asString method converts it to decimal string (SQLString object) like function printf("% g", ...) does.

If the value's type of current object is numeric (kDataTypeNumeric), asString method converts it to decimal string (SQLString object) without precision loss.

If the value's type of current object is date-time (kDataTypeDateTime), asString method converts it to string (SQLString object) like function asctime(...) does.

If the value's type of current object is string (kDataTypeString, kDataTypeLongChar, kDataTypeCLob), asString method returns the original object's value as SQLString object.

If the value's type of current object is binary (kDataTypeBytes, kDataTypeLongBinary, kDataTypeBLob), asString method converts it to hexadecimal string (SQLString object).

If the value's type of current object is cursor (kDataTypeCursor), the result is undefined and debug version asserts.

Use DataType method to get the value's type of SQLValueRead object.

asStringValue as String

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: returns the value string.
Notes: Same as asString but returns a REALbasic string.

asULong as UInt32

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the value as an unsigned 32 bit integer value.

Notes:

If the value of current object is NULL, asULong method returns 0. Use isNull method to make sure if the value is NULL or not.

If the value's type of current object is long (kDataTypeLong), asULong method returns the original value with no conversion.

If the value's type of current object is bool (kDataTypeBool), short (kDataTypeShort), double (kDataTypeDouble) or numeric (kDataTypeNumeric), asULong method converts it to long data type. Note, that in this case the returned value can be truncated.

If the value's type of current object is string (kDataTypeString), asULong method tries to convert it to long (kDataTypeLong) value. If the conversion is possible and correct, asULong returns kDataTypeLong value. If conversion is incorrect asULong method throws an exception.

If the value's type of current object is any data type except the described above, the result is undefined and debug version asserts.

Use DataType method to get the value's type of SQLValueRead object.

asUShort as UInt16

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the value as unsigned short.
Notes:

If the value of current object is NULL, asUShort method returns 0. Use isNull method to make sure if the value is NULL or not.

If the value's type of current object is unsigned short, asUShort method returns the original value with no conversion.

If the value's type of current object is bool (kDataTypeBool), long (kDataTypeLong), unsigned long (Sk-

DataTypeULong), double (kDataTypeDouble) or numeric (kDataTypeNumeric), asUShort method converts it to unsigned short data type. Note, that in this case the returned value can be truncated.

If the value's type of current object is string (kDataTypeString), asUShort method tries to convert it to unsigned short value. If the conversion is possible and correct, asUShort returns the value. If conversion is incorrect asUShort method throws an exception.

If the value's type of current object is any data type except the described above, the result is undefined and debug version asserts.

Use DataType method to get the value's type of SAValueRead object.

Constructor(DataType as integer)

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new empty value object for the given data type.
See also:

- 2.13.1 Constructor(value as SQLValueReadMBS) 52

Constructor(value as SQLValueReadMBS)

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new value object by copying the given one.
See also:

- 2.13.1 Constructor(DataType as integer) 52

DataType as integer

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns SAValueRead object's data type.
Notes: One of the kDataType constants.

isNull as boolean

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns true if the value of current object is NULL; otherwise false.

LongOrLobReaderMode as integer

method from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The Long or Lob data reading mode.
Notes:

SQLAPI++ Library provides two ways to read Long or BLob(CLob) object's value (usually SQLField or SQLParam objects):

1. reading of Long or Lob data at once into an internal buffer (like ordinary string or binary values);
2. piecewise reading of Long or Lob data using user defined callback.
kLongOrLobReaderDefault reading mode used by default.

If you want to control piecewise reading of Long or BLob(CLob) data you should set LongOrLobReaderMode and use kLongOrLobReaderManual reading mode for Long or BLob(CLob) parameters or fields you want to process with your data consumer. After that each fetch will skip reading Long and BLob(CLob) parameters that you set to be read manually. To read field or parameter defined to be read manually you should call ReadLongOrLob method for each of them after the fetch. ReadLongOrLob method will repeatedly call the data consumer Write event.
(Read and Write computed property)

2.13.2 Constants

kDataTypeBLob=14

const from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the field type constants
Notes: Data type is BLob data (SQLStringMBS).

kDataTypeBool=1

const from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the field type constants
Notes: Data type is a boolean.

kDataTypeBytes=11

const from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the field type constants
Notes: Data type is binary string (SQLStringMBS).

kDataTypeCLob=15

const from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the field type constants
Notes: Data type is CLob data (SQLStringMBS).

kDataTypeCursor=16

const from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the field type constants
Notes: Data type is Oracle REF CURSOR (SQLCommand).

kDataTypeDateTime=8

const from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the field type constants
Notes: Data type is SQLDateTime.

kDataTypeDouble=6

const from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the field type constants
Notes: This is a normal double variable.

kDataTypeInterval=9

const from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the field type constants
Notes: Data type is an interval (SQLIntervalMBS).

kDataTypeLong=4

const from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the field type constants
Notes: A 32 bit integer.

kDataTypeLongBinary=12

const from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the field type constants
Notes: Data type is long binary data (SQLStringMBS).

kDataTypeLongChar=13

const from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the field type constants
Notes: Data type is long character data (SQLStringMBS).

kDataTypeNumeric=7

const from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the field type constants
Notes: Data type is SQLNumeric (used internally).

kDataTypeShort=2

const from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the field type constants
Notes: Data type is a 16 bit signed integer.

kDataTypeSpecificToDBMS=17

const from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the field type constants
Notes: Data type is server-specific.

kDataTypeString=10

const from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the field type constants
Notes: Data type is character string (SQLString).

kDataTypeULong=5

const from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the field type constants
Notes: Data type is a 32 bit unsigned integer.

kDataTypeUnknown=0

const from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the field type constants
Notes: Data type is unknown.

kDataTypeUShort=3

const from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the field type constants
Notes: A 16 bit unsigned integer.

kLongOrLobReaderModeDefault=0

const from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the read modes.
Notes: Long or Lob(CLob) data reading mode is default (automatic).

kLongOrLobReaderModeManual=1

const from class SQLValueReadMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the read modes.
Notes: Long or Lob(CLob) data reading mode is manual.

2.14 class SQLUnsupportedExceptionMBS

class SQLUnsupportedExceptionMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The class for an exception to report that the function is not supported on this platform.
Notes:

This one raises only if the plugin is compiled for Mac OS Classic.
Subclass of the SQLErrorExceptionMBS class.

2.15 class SQLValueMBS

class SQLValueMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The SQL class for mutable values.
Notes: Subclass of the SQLValueReadMBS class.

2.15.1 Methods

Constructor(DataType as integer)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new empty value object with the given data type.

isDefault as boolean

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns true if the plugin has been forced to use parameter's default value by calling setAsDefault method; false otherwise.

setAsBlob(data as SQLDataProviderMBS, BlockSize as UInt32)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as BLob data (SQLString)
Notes:

When you call the SQLCommandMBS.Execute method all input parameters are bound with their values, including Long and BLob(CLob) parameters.
That is the time when the data provider Read event runs to get the values in the block size you specify.

The default value for the block size is 0. If you use the default value, SQLAPI++ Library will automatically use the most appropriate size for current DBMS.
See also:

- 2.15.1 setAsBlob(data as SQLStringMBS) 60
- 2.15.1 setAsBlob(data as string) 61

setAsBlob(data as SQLStringMBS)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as BLob data (SQLString)
See also:

- 2.15.1 setAsBlob(data as SQLDataProviderMBS, BlockSize as UInt32) 60
- 2.15.1 setAsBlob(data as string) 61

setAsBlob(data as string)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as BLob data (SQLString)
See also:

- 2.15.1 setAsBlob(data as SQLDataProviderMBS, BlockSize as UInt32) 60
- 2.15.1 setAsBlob(data as SQLStringMBS) 60

setAsBool(value as boolean)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as bool data.

setAsBytes(value as SQLStringMBS)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as binary string data (SQLString).

setAsClob(data as SQLDataProviderMBS, BlockSize as UInt32)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as CLob data (SQLString)
Notes:

nWhen you call the SQLCommandMBS.Execute method all input parameters are bound with their values, including Long and BLob(CLob) parameters.
That is the time when the data provider Read event runs to get the values in the block size you specify.

The default value for the block size is 0. If you use the default value, SQLAPI++ Library will automatically use the most appropriate size for current DBMS.
See also:

- 2.15.1 setAsClob(text as SQLStringMBS) 62
- 2.15.1 setAsClob(text as string) 62

setAsClob(text as SQLStringMBS)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as CLob data (SQLString)
See also:

- 2.15.1 setAsClob(data as SQLDataProviderMBS, BlockSize as UInt32) 61
- 2.15.1 setAsClob(text as string) 62

setAsClob(text as string)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as CLob data (SQLString)
See also:

- 2.15.1 setAsClob(data as SQLDataProviderMBS, BlockSize as UInt32) 61
- 2.15.1 setAsClob(text as SQLStringMBS) 62

setAsDateTime(value as SQLDateTimeMBS)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as SQLDateTime data.

setAsDefault

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Forces to use the default parameter's value.
Notes:

Forces DBMS API to use the default parameter value as the input value for an input or input/output parameter in a procedure.

If DBMS API does not support the concept of "default parameter values" in stored procedures, this setting will be ignored.

If you set this flag for the parameter that doesn't have a default value, the effect is DBMS defined (e.g. an error can be returned or NULL can be bound).

To cancel using the default parameter value you should call any other SQLValue::setAs... method to bind a parameter value.

To check whether this flag is set or not use isDefault method.

setAsDouble(value as double)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as double data.

setAsInterval(value as SQLIntervalMBS)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets an interval value.

setAsLong(value as Int32)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as long data.

setAsLongBinary(data as SQLDataProviderMBS, BlockSize as UInt32)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as long binary data (SQLString)
Notes:

When you call the SQLCommandMBS.Execute method all input parameters are bound with their values, including Long and BLob(CLob) parameters.
That is the time when the data provider Read event runs to get the values in the block size you specify.

The default value for the block size is 0. If you use the default value, SQLAPI++ Library will automatically use the most appropriate size for current DBMS.
See also:

- 2.15.1 setAsLongBinary(data as SQLStringMBS) 64
- 2.15.1 setAsLongBinary(data as string) 65

setAsLongBinary(data as SQLStringMBS)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as long binary data (SQLString)
See also:

- 2.15.1 setAsLongBinary(data as SQLDataProviderMBS, BlockSize as UInt32) 64

setAsLongBinary(data as string)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as long binary data (SQLString)
See also:

setAsLongChar(data as SQLDataProviderMBS, BlockSize as UInt32)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as long character data (SQLString)
Notes:

When you call the SQLCommandMBS.Execute method all input parameters are bound with their values, including Long and BLob(CLob) parameters.
That is the time when the data provider Read event runs to get the values in the block size you specify.

The default value for the block size is 0. If you use the default value, SQLAPI++ Library will automatically use the most appropriate size for current DBMS.
See also:

setAsLongChar(text as SQLStringMBS)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as long character data (SQLString)
See also:

- 2.15.1 setAsLongChar(data as SQLDataProviderMBS, BlockSize as UInt32) 65
- 2.15.1 setAsLongChar(text as string) 66

setAsLongChar(text as string)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as long character data (SQLString)
See also:

- 2.15.1 setAsLongChar(data as SQLDataProviderMBS, BlockSize as UInt32) 65
- 2.15.1 setAsLongChar(text as SQLStringMBS) 65

setAsNull

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets value to null.

setAsNumeric(value as SQLNumericMBS)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as SQLNumeric data.

setAsShort(value as Int16)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as short data.

setAsString(text as string)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as character string data.
Notes: Same as setAsString, but for your convenience with a REALbasic string instead of a SQLStringMBS object.
See also:

- 2.15.1 setAsString(value as SQLStringMBS) 67

setAsString(value as SQLStringMBS)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as character string data (SQLString)
See also:

- 2.15.1 setAsString(text as string) 67

setAsULong(value as UInt32)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as unsigned long data.

setAsUnknown

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's type as unknown.

setAsUShort(value as UInt16)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value as unsigned short data.
Notes: Sets value as unsigned short data.

setAsValueRead(value as SQLValueReadMBS)

method from class SQLValueMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets parameter's value from SQLParam or SQLField objects.
Notes: This method allows using SQLField or SQLParam object received from one SQL statement as a parameter for another SQL statement.

2.16 class SQLConnectionMBS

class SQLConnectionMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The class for a SQL Plugin Database connection.
Example:

```
dim con as new SQLConnectionMBS

try

// where is the library?
con.SetFileOption con.kOptionLibraryMySQL, SpecialFolder.UserHome.Child("libmysqlclient.dylib")
```

```
// connect to database
// in this example it is Oracle,
// but can also be Sybase, Informix, DB2
// SQLServer, InterBase, SQLBase and ODBC

dim server as string = "192.168.1.80:3306@test"

con.Connect(server,"root","",SQLConnectionMBS.kMySQLClient)

MsgBox "We are connected!"

// Disconnect is optional
// autodisconnect will ocur in destructor if needed
con.Disconnect

msgbox "We are disconnected!"

catch r as RuntimeException
MsgBox r.message

// SAConnection::Rollback()
// can also throw an exception
// (if a network error for example),
// we will be ready
try

// on error rollback changes
con.Rollback

catch rr as runtimeexception
MsgBox rr.message
end try
end try
```

Notes: Supported databases: Oracle, Microsoft SQL Server, DB2, Sybase, Informix, InterBase/Firebird, SQLBase, MySQL, PostgreSQL and ODBC and SQLite

2.16.1 Methods

AutoCommit as integer

method from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Whether autocommit is enabled or disabled for the current connection.
Notes:

If autocommit is on, the database is committed automatically after each SQL command. Otherwise, transaction is committed only after Commit calling.
(Read and Write computed property)

Client as integer

method from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The current DBMS client assigned for the connection.
Notes: (Read and Write computed property)

ClientVersion as integer

method from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Gets the DBMS client API version number.
Notes:

The higher word contains the major client version (the XX value in the XX.YY version number); the lower word contains the minor client version (the YY value in the XX.YY version number).

If an DBMS client was not set calling ClientVersion method will throw an exception.

Commit

method from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Saves any changes and ends the current transaction.
Notes:

Use Commit method to write transaction changes permanently to a database. It commits the work of all commands that associated with that connection.

All changes to the database since the last commit are made permanent and cannot be undone. Before a commit, all changes made since the start of the transaction can be rolled back using Rollback method.

Connect(DBString as string, UserID as string, Password as string, client as integer=0)

method from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Opens the connection to a data source.
Notes:

DBString:	Name of database this connection will connect to (see Server specific notes).
UserID:	A string containing a user name to use when establishing the connection (see Server specific notes).
Password:	A string containing a password to use when establishing the connection.
client:	Optional. One of the following values from k*Client constants.

Using the Connect method on a SAConnection object establishes the physical connection to a data source. After this method successfully completes, the connection is live and you can issue commands against it and process the results.

If you use the default value of Client parameter, you should set Client before using Connect.

To check whether a connection established use isConnected method. To check whether a connection is brocken or not use isAlive method.

see also for server specific notes:

http://www.sqlapi.com/OnLineDoc/Connection_ Connect.html

Disconnect

method from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Disconnects the connection from the database.

isAlive as boolean

method from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the database server connection status for a particular connection object.
Notes:

Returns true if the database server is active and accessible; otherwise false.
This method uses the safe query execution for most supported DBMS-es. The query uses the well known database table or procedure (mysql_ ping is used for MySQL). If the query fails the method returns false.

isConnected as boolean

method from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the connection state for a particular connection object.
Notes: Returns true if connected; otherwise false.

IsolationLevel as integer

method from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The transaction isolation level.
Notes:

Use the kReadCommitted, kReadUncommitted, kRepeatableRead, kSerializable and kLevelUnknown constants.
(Read and Write computed property)

NativeAPI as SQLAPIMBS

method from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.5, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns a set of functions of native DBMS client API.

Option(name as string) as string

method from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: A string value of a specific connection or command option.
Notes:

see also:
http://www.sqlapi.com/OnLineDoc/Connection_ Option.html
(Read and Write computed property)

Rollback

method from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Cancels any changes made during the current transaction and ends the transaction.
Notes:

Rollback method rolls back the database to the state it was in at the completion of the last commit operation. All uncommitted work is undone.

Rollback method rolls back the work of all commands that associated with that connection.

To commit all changes made since the start of the transaction use Commit method.

ServerVersion as integer

method from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Gets the currently connected DBMS server version number.
Notes: The higher word contains the major server version (the XX value in the XX.YY version number); the lower word contains the minor server version (the YY value in the XX.YY version number).

ServerVersionString as string

method from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Gets the currently connected DBMS server version string.
Notes: A server version string may contain some useful information about server brand, configuration and so on. It is a good idea to display this information in all your applications.

SetFileOption(name as string, file as folderitem)

method from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.5, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets an option with passing a file path.
Example:

```
dim db as new SQLConnectionMBS

// where is the library?
db.SetFileOption SQLConnectionMBS.kOptionLibraryMySQL, SpecialFolder.UserHome.Child("libmysqlclient.dylib")
```

Notes: Makes sure the path is correct.

SQLExecute(command as string, CommandType as integer=0)

method from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.2, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Executes a SQL command and ignores result.
Notes:

This is a convenience function.
Internally it creates a SQLCommandMBS with the given command and calls Execute.

All text strings sent to the plugin must have a defined encoding. Else the internal text encoding conversions will fail.

SQLExecuteMT(command as string, CommandType as integer=0)

method from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.4, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Executes a SQL command and ignores result.
Notes:

This is a convenience function.
Internally it creates a SQLCommandMBS with the given command and calls Execute.

The work is performed on an extra thread, so this function can yield time to other REAL Studio threads. And it calles the Working event regulary. For best user experience run this command on a REAL Studio thread, so your GUI stays responsive.

All text strings sent to the plugin must have a defined encoding. Else the internal text encoding conversions will fail.

SQLSelect(command as string, CommandType as integer=0) as string

method from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.2, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Executes a SQL command and returns the first field's string value.
Notes:

This is a convenience function.
Internally it creates a SQLCommandMBS with the given command and calls Execute.
If the result is a record set, the first field from the first row is returned.
This is basicly useful for commands like "select sqlite_ version()".

All text strings sent to the plugin must have a defined encoding. Else the internal text encoding conversions will fail.

SQLSelectMT(command as string, CommandType as integer=0) as string

method from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.4, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Executes a SQL command and returns the first field's string value.
Notes:

This is a convenience function.
Internally it creates a SQLCommandMBS with the given command and calls Execute.
If the result is a record set, the first field from the first row is returned.
This is basicly useful for commands like "select sqlite_ version()".

The work is performed on an extra thread, so this function can yield time to other REAL Studio threads. And it calles the Working event regulary. For best user experience run this command on a REAL Studio thread, so your GUI stays responsive.

All text strings sent to the plugin must have a defined encoding. Else the internal text encoding conversions will fail.

2.16.2 Events

Working

event from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.4, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The event called while the SQLExecuteMT and SQLSelectMT methods are running.

2.16.3 Constants

kANSILevel0 = 0

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the ANSI level constants.

kANSILevel1 = 1

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the ANSI level constants.

kANSILevel2 = 2

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the ANSI level constants.

kANSILevel3 = 3

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the ANSI level constants.

kAutoCommitOff = 0

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the constants for the autocommit property.
Notes: Autocommit is off.

kAutoCommitOn = 1

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the constants for the autocommit property.
Notes: Autocommit is on.

kAutoCommitUnknown = -1

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the constants for the autocommit property.

kClientNotSpecified = 0

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the database client constants.
Notes: Client is not specified.

kDB2Client = 6

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the database client constants.
Notes: DB2 client.

kFirebirdClient = 4

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.4, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the database client constants.
Notes: InterBase/Firebird client.

kInformixClient = 7

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the database client constants.
Notes: Informix client.

kInterBaseClient = 4

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the database client constants.
Notes: InterBase/Firebird client.

kLevelUnknown = -1

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the isolation level constants.
Notes: Unknown

kMySQLClient = 9

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the database client constants.
Notes: MySQL client.

kODBCClient = 1

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the database client constants.
Notes: ODBC client.

kOptionAPPNAME = "APPNAME"

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: A constant for the options.

kOptionLibraryDB2 = "DB2CLI.LIBS"

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.5, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the option constant to specify the DB2 library.
Notes: Use with SetFileOption.

kOptionLibraryFirebird = "IBASE.LIBS"

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.5, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the option constant to specify the firebird library.
Notes: Use with SetFileOption.

kOptionLibraryInformix = "INFCLI.LIBS"

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.5, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the option constant to specify the informix library.
Notes: Use with SetFileOption.

kOptionLibraryInterbase = "IBASE.LIBS"

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.5, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the option constant to specify the interbase library.
Notes: Use with SetFileOption.

kOptionLibraryMySQL = "MYSQL.LIBS"

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.5, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the option constant to specify the MySQL library.
Notes: Use with SetFileOption.

kOptionLibraryODBC = "ODBC.LIBS"

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.5, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the option constant to specify the ODBC library.
Notes: Use with SetFileOption.

kOptionLibraryOracle = "OCI8.LIBS"

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.5, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the option constant to specify the Oracle library.
Notes: Use with SetFileOption.

kOptionLibraryPostgreSQL = "LIBPQ.LIBS"

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.5, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the option constant to specify the Postgre SQL library.
Notes: Use with SetFileOption.

kOptionLibrarySeparator = ":"

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.5, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the option constant to specify the platform specific path separator.
Notes:

Use with SetFileOption to specify multiple file paths for a library.
Has a different value on the different platforms.

kOptionLibrarySQLBase = "SQLBASE.LIBS"

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.5, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the option constant to specify the SQLBase library.
Notes: Use with SetFileOption.

kOptionLibrarySQLite = "SQLITE.LIBS"

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.5, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the option constant to specify the sqlite library.
Notes: Use with SetFileOption.

kOptionLibrarySybaseComn = "SYBCOMN.LIBS"

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.5, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the option constant to specify the Sybase library.
Notes: Use with SetFileOption.

kOptionLibrarySybaseCS = "SYBCS.LIBS"

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.5, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the option constant to specify the Sybase library.
Notes: Use with SetFileOption.

kOptionLibrarySybaseCT = "SYBCT.LIBS"

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.5, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the option constant to specify the Sybase library.
Notes: Use with SetFileOption.

kOptionLibrarySybaseIntl = "SYBINTL.LIBS"

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.5, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the option constant to specify the Sybase library.
Notes: Use with SetFileOption.

kOptionLibrarySybaseTCL = "SYBTCL.LIBS"

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.5, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the option constant to specify the Sybase library.
Notes: Use with SetFileOption.

kOptionWSID = "WSID"

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: A constant for the options.

kOracleClient = 2

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the database client constants.
Notes: Oracle client.

kPostgreSQLClient = 10

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the database client constants.
Notes: PostgreSQL client.

kReadCommitted = 1

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the isolation level constants.
Notes: Read committed.

kReadUncommitted = 0

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the isolation level constants.
Notes: Read uncommitted.

kRepeatableRead = 2

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the isolation level constants.
Notes: Repeatable read.

kSerializable = 3

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the isolation level constants.
Notes: Serializable.

kSQLBaseClient = 5

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the database client constants.
Notes: SQLbase client.

kSQLiteClient = 11

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the database client constants.
Notes: SQLite client.

kSQLServerClient = 3

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the database client constants.
Notes:

Mircosoft SQL Server client.

You may need to download the client packages for accessing SQL Server. Files like the SQLNCLI dll may be missing. You can download for example the Feature Pack for Microsoft SQL Server 2005 from the microsoft download page.

kSybaseClient = 8

const from class SQLConnectionMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the database client constants.
Notes: Sybase client.

2.17 class SQLBLobMBS

class SQLBLobMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: A class for a blob.
Notes:

Basicly this is a SQLStringMBS which is always marked to contain binary data. You only need this class to use the constructor with dataprovider to stream data to the database.
Subclass of the SQLLongOrLobMBS class.

2.17.1 Methods

Constructor(data as SQLStringMBS)

method from class SQLBLobMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new blob object from a string object.
See also:

- 2.17.1 Constructor(dataProvider as SQLDataProviderMBS, BlockSize as UInt32) 88

Constructor(dataProvider as SQLDataProviderMBS, BlockSize as UInt32)

method from class SQLBLobMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new blob object from a data provider.
Notes:

The blocksize specifies in which sizes data is requested from the provider.
You must make sure that the data provider and this new blob object life long enough. Because the actual data is requested later when you do the update on the database.

If BlockSize is 0, the default block size is used.
See also:

- 2.17.1 Constructor(data as SQLStringMBS) 88

2.18 class SQLAPIMBS

class SQLAPIMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.5, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: This is a class for the native APIs.
Notes: The plugin does not implem

2.18.1 Properties

Connection as SQLConnectionMBS

property from class SQLAPIMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.5, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The database connection this API is used with.
Notes: (Read only property)

2.19 class SQLCLobMBS

class SQLCLobMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: A class for a clob (character large object).
Notes:

Basicly this is a SQLStringMBS which is always marked to contain text. You only need this class to use the constructor with dataprovider to stream data to the database.
Subclass of the SQLLongOrLobMBS class.

2.19.1 Methods

Constructor(data as SQLStringMBS)

method from class SQLCLobMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new clob object from a string object.
See also:

- 2.19.1 Constructor(dataProvider as SQLDataProviderMBS, BlockSize as UInt32) 90

Constructor(dataProvider as SQLDataProviderMBS, BlockSize as UInt32)

method from class SQLCLobMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new clob object from a data provider.
Notes:

The blocksize specifies in which sizes data is requested from the provider.
You must make sure that the data provider and this new clob object life long enough. Because the actual data is requested later when you do the update on the database.

If BlockSize is 0, the default block size is used.
See also:

- 2.19.1 Constructor(data as SQLStringMBS) 90

2.20 class SQLBytesMBS

class SQLBytesMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The class for a string of bytes.
Notes: Subclass of the SQLStringMBS class.

2.20.1 Methods

Constructor(data as SQLStringMBS)

method from class SQLBytesMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new bytes object based on the given string object.

2.21 class PostgreSQLAPIMBS

class PostgreSQLAPIMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.8, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The class for Postgre SQL specific functions.
Notes: Subclass of the SQLAPIMBS class.

2.21.1 Methods

DB(conn as SQLConnectionMBS) as string

method from class PostgreSQLAPIMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.8, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The database name used to create the connection.

ErrorMessage(conn as SQLConnectionMBS) as string

method from class PostgreSQLAPIMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.8, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The last error message.

Field(cmd as SQLCommandMBS, RecordIndex as integer, FieldIndex as integer) as string

method from class PostgreSQLAPIMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.8, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Queries a field by index for the row with the RecordIndex.
See also:

- 2.21.1 Field(cmd as SQLCommandMBS, RecordIndex as integer, FieldName as string) as string 92

Field(cmd as SQLCommandMBS, RecordIndex as integer, FieldName as string) as string

method from class PostgreSQLAPIMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.8, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Queries a field by name for the row with the RecordIndex.
See also:

- 2.21.1 Field(cmd as SQLCommandMBS, RecordIndex as integer, FieldIndex as integer) as string 92

FieldCount(cmd as SQLCommandMBS) as integer

method from class PostgreSQLAPIMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.8, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The number of fields in the result.

Host(conn as SQLConnectionMBS) as string

method from class PostgreSQLAPIMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.8, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The host used to create the connection.

Options(conn as SQLConnectionMBS) as string

method from class PostgreSQLAPIMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.8, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The options used to create the connection.

Password(conn as SQLConnectionMBS) as string

method from class PostgreSQLAPIMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.8, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The password used to create the connection.

Port(conn as SQLConnectionMBS) as string

method from class PostgreSQLAPIMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.8, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The port used to create the connection.

RecordCount(cmd as SQLCommandMBS) as integer

method from class PostgreSQLAPIMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.8, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The number of records in the result.

TTY(conn as SQLConnectionMBS) as string

method from class PostgreSQLAPIMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.8, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The tty used to create the connection.

User(conn as SQLConnectionMBS) as string

method from class PostgreSQLAPIMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.8, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The user name used to create the connection.

2.22 class SQLCommandMBS

class SQLCommandMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: This is the central class for the using the SQL database access.
Example:

```
dim con as SQLConnectionMBS
dim cmd as SQLCommandMBS

try

con = new SQLConnectionMBS // connection object
cmd = new SQLCommandMBS // create command object

// where is the library?
con.SetFileOption con.kOptionLibraryMySQL, SpecialFolder.UserHome.Child("libmysqlclient.dylib")

// connect to database (mySQL in our example)
// server: 192.168.1.80
// port: 3306
// database: test
// name: root
// no password
con.Connect("192.168.1.80:3306@test","root","",SQLConnectionMBS.kMySQLClient)
// associate a command with connection
// connection can also be specified in SACommand constructor
cmd.Connection=con

// create table
cmd.setCommandText("Create table test_ tbl(fid integer, fvarchar20 varchar(20), fblob blob)")
cmd.Execute

// insert value
cmd.setCommandText("Insert into test_ tbl(fid, fvarchar20) values (1, 'Some string (1)')")
cmd.Execute

// commit changes on success
con.Commit

MsgBox("Table created, row inserted!")

catch r as SQLErrorExceptionMBS
// SAConnection::Rollback()
// can also throw an exception
// (if a network error for example),
// we will be ready
try

// on error rollback changes
if con<>nil then
con.rollback
end if
catch x as SQLErrorExceptionMBS
// ignore
```

```
end try

// show error message
MsgBox r.message
end try
```

2.22.1 Methods

Cancel

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Attempts to cancel the pending result set, or current statement execution.
Notes:

Cancel can cancel the following types of processing on a statement:

A function running asynchronously on the statement.
A function running on the statement on another thread.
After an application calls a function asynchronously, it checks repeatedly to determine whether it has finished processing. While the function is processing, an application can call Cancel to cancel the function.

In a multithread application, the application can cancel a function that is running synchronously on a statement.

see also
http://www.sqlapi.com/OnLineDoc/Command_ Cancel.html

Close

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Closes the specified command object.
Notes:

Use the Close method to close the command explicitly.

A command will be implicitly closed in destructor, so you don't have to call Close method explicitly.

CommandText as string

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Gets the command text associated with the SACommand object.
Example:

```
dim s as new SQLCommandMBS(nil, "select * from test")

MsgBox s.CommandText
```

Notes:

Use the CommandText method to return the command text declared in SACommand constructor or set-CommandText method.

All text strings sent to the plugin must have a defined encoding. Else the internal text encoding conversions will fail.

CommandType as integer

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Gets the command type currently associated with the SACommand object.
Notes:

One of the following values from SACommandType_ t enum:

* kCommandTypeUnknown Command type is not defined. Library will detect command type automatically

when needed.
kCommandTypeSQLStmt Command is an SQL statement.
kCommandTypeSQLStmtRaw Command is an SQL statement that mustn't be interpreted by SQLAPI++.
kCommandTypeStoredProc Command is a stored procedure or a function.
Remarks

The command type can be explicitly set in SACommand constructor and setCommandText method, but it's not necessary to do it.

The CommandTypemethod returns the command type value that was specified in SACommand constructor or setCommandText method. If you declared the command type value as kCommandTypeUnknown (the default value) then command type is detected by the Library and the CommandType method returns this detected value.

Connection as SQLConnectionMBS

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The connection for the command.
Notes:

When you set the connection on a command object that already has associated connection, the previous association will be correctly discarded (with closing opened command if needed) and new connection will be set.

If you attempt to call any method on a SACommand object that requires database access with no valid connection, an error occurs.
(Read and Write computed property)

Constructor

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new command object with no connection and no command text.
See also:

Constructor(connection as SQLConnectionMBS, SQLCommand as String, CommandType as integer=0)

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: This constructor initializes a new SQLCommandMBS object.
Notes:

Connection: the connection to associated with the command.
SQLCommand: A string which represents command text string (an SQL statement or a stored procedure name). If it is an empty string, no command text is associated with the command, and you have to call setCommandText method later.
CommandType: The type of command like kCommandTypeUnknown, kCommandTypeSQLStatement, kCommandTypeSQLStatementRaw or kCommandTypeStoredProcedure.

All text strings sent to the plugin must have a defined encoding. Else the internal text encoding conversions will fail.
See also:

CreateParam(name as string, ParamType as integer, DirType as integer=0) as SQLParamMBS

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates parameter associated with the specified command.
Notes:

Parameters

Returns a new SQLParamMBS object on success or nil on any error.

Normally you should not create parameters by yourself. The Library automatically detects whether the command has parameters in terms of the command text and implicitly creates a set of SAParam objects.

name:	A string representing the name of parameter.
ParamType:	Type of the parameter's value. Use the kDataType constants.
ParamSize:	An integer value represents parameter's value size.
ParamPrecision:	An integer value represents parameter's value precision.
ParamScale:	An integer value represents parameter's value scale.
DirType:	Type of the parameter. Use the kParamDirType* constants.

Nevertheless, if you call CreateParam explicitly you have to delete all SAParam objects created automatically by the Library before. Use DestroyParams method before the first call of CreateParam method.
See also:

- 2.22.1 CreateParam(name as string, ParamType as integer, NativeType as integer, ParamSize as integer, ParamPrecision as integer, ParamScale as integer, DirType as integer=0) as SQLParamMBS 100

CreateParam(name as string, ParamType as integer, NativeType as integer, ParamSize as integer, ParamPrecision as integer, ParamScale as integer, DirType as integer=0) as SQLParamMBS

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates parameter associated with the specified command.
Notes:

Parameters

name:	A string representing the name of parameter.
ParamType:	Type of the parameter's value. Use the kDataType constants.
ParamSize:	An integer value represents parameter's value size.
ParamPrecision:	An integer value represents parameter's value precision.
ParamScale:	An integer value represents parameter's value scale.
DirType:	Type of the parameter. Use the kParamDirType* constants.

Returns a new SQLParamMBS object on success or nil on any error.

Normally you should not create parameters by yourself. The Library automatically detects whether the command has parameters in terms of the command text and implicitly creates a set of SAParam objects.

Nevertheless, if you call CreateParam explicitly you have to delete all SAParam objects created automatically by the Library before. Use DestroyParams method before the first call of CreateParam method.
See also:

- 2.22.1 CreateParam(name as string, ParamType as integer, DirType as integer=0) as SQLParamMBS 99

DestroyParams

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Destroys all parameters associated with the specified command.
Notes:

DestroyParams method destroys all parameters either created automatically by the Library or by user.

Normally you should not create and delete parameters by yourself. The Library automatically detects whether the command has parameters, implicitly creates a set of SAParam objects and then deletes them in SACommanddestructor. But if you have some reason to create parameters explicitly use CreateParam method and then call DestroyParams method to delete all parameters after your work with parameters is over.

Execute

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Executes the current command.
Notes:

Use the Execute method to execute the query or stored procedure specified in the command text. Execute method calls Prepare method implicitly if needed. If the command has input parameters, they should be bound before calling Execute method. Input parameters represented by SAParam object. To bind input variables assign a value to SAParam object returning by Param or ParamByIndex methods.

A command (an SQL statement or procedure) can have a result set after executing. To check whether a result set exists use isResultSet method. If result set exists, a set of SAField objects is created after command execution. Rows from the result set can be fetched one by one using FetchNext method. To get field

description or value use Field method.

Output parameters represented by SAParam objects. They are available after command execution. To get parameter description or value use Param or ParamByIndex methods.

ExecuteCommand(SQLCommand as string, CommandType as integer=0)

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.2, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Executes the given command.
Notes:

This is a convenience function.
Internally it calls setCommandText with the given command and calls Execute.

All text strings sent to the plugin must have a defined encoding. Else the internal text encoding conversions will fail.

ExecuteCommandMT(SQLCommand as string, CommandType as integer=0)

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.4, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Executes the given command.
Notes:

This is a convenience function.
Internally it calls setCommandText with the given command and calls Execute.

The work is performed on an extra thread, so this function can yield time to other REAL Studio threads. And it calles the Working event regulary. For best user experience run this command on a REAL Studio thread, so your GUI stays responsive.

All text strings sent to the plugin must have a defined encoding. Else the internal text encoding conversions will fail.

ExecuteMT

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.4, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Executes the current command.
Notes:

Use the Execute method to execute the query or stored procedure specified in the command text. Execute method calls Prepare method implicitly if needed. If the command has input parameters, they should be bound before calling Execute method. Input parameters represented by SAParam object. To bind input variables assign a value to SAParam object returning by Param or ParamByIndex methods.

A command (an SQL statement or procedure) can have a result set after executing. To check whether a result set exists use isResultSet method. If result set exists, a set of SAField objects is created after command execution. Rows from the result set can be fetched one by one using FetchNext method. To get field description or value use Field method.

Output parameters represented by SAParam objects. They are available after command execution. To get parameter description or value use Param or ParamByIndex methods.

The work is performed on an extra thread, so this function can yield time to other REAL Studio threads. And it calles the Working event regulary. For best user experience run this command on a REAL Studio thread, so your GUI stays responsive.

FetchNext as boolean

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Fetches rows from a result set.
Notes:

Returns true if the next row was fetched; otherwise false .

Use FetchNext method to fetch row by row from the result set.

Each column of fetched row is represented by SAField object. If a result set exists after the last command execution, a set of SAField objects is created implicitly. To check whether a result set exists use isResultSet method. FetchNext method updates value parts of SAField objects.

To get field description or value use Field method.

Field(index as integer) as SQLFieldMBS

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the column specified by its position in the result set.
Notes:

index: A one-based field number in a result set.

Use Field method to access a field by its name or position in the result set.

Using an index smaller than 1 and greater then the value returned by FieldCount method will result in a failed assertion.

A set of SAField objects creates implicitly after the command execution if the result set exists.SAField object contains full information about a column: name, type, size, value.
See also:

- 2.22.1 Field(name as string) as SQLFieldMBS 104

Field(name as string) as SQLFieldMBS

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the column specified by its name in the result set.
Notes:

name: A string that represents a name of the requested field.

Returns a reference to a SAField object.

Use Field method to access a field by its name or position in the result set.

Using a non-existent field name will throw an exception.

A set of SAField objects creates implicitly after the command execution if the result set exists.SAField object contains full information about a column: name, type, size, value.
See also:

- 2.22.1 Field(index as integer) as SQLFieldMBS 104

FieldCount as integer

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the number of fields (columns) in a result set.
Notes:

FieldCount method returns the number of fields created implicitly after the command execution if a result set exists.

A field is represented by SAField object. You can get field value and description using Field method.

isExecuted as boolean

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Whether this command was already executed.

isOpened as boolean

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns true if the SACommand object is opened; otherwise false.

isResultSet as boolean

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Tests whether a result set exists after the command execution.
Notes: Returns true if the result set exists; otherwise false.

Open

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Opens the specified command object.
Notes:

Use the Open method to open the command explicitly.

A command will be implicitly opened by any method that needs an open command, therefore you don't have to call it explicitly.

To test whether a command is opened use isOpened method.

Option(name as string) as string

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: A string value of a specific command option.
Notes:

see also:
http://www.sqlapi.com/OnLineDoc/Command_ Option.html
(Read and Write computed property)

Param(ID as integer) as SQLParamMBS

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the command parameter specified by its position.
Notes:

ID: A position of parameter specified in the command text. Normally position is a number stated in the command text after a colon (for example, 1 for :1, 5 for :5).

Returns a reference to a SAParam object which is only valid as long as the param object is not deleted by the library.

Use Param method to access a parameter by its name or position (in SQL statement). If, for example, you want to walk through all the parameters use ParamByIndex method.

If parameters were not created before calling Param method the Library creates them implicitly (can query native API if needed and therefore can throw exception on error) and then returns the specified parameter.

Passing a value of name or position which does not specified in the command text will throw an exception.

SAParam object contains full information about a parameter: name, type, size, etc. Values for the input parameters can be assigned to SAParam object.
See also:

Param(name as string) as SQLParamMBS

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the command parameter specified by its name.
Notes:

Name: A string that represents a name of the requested parameter. Normally name is a string stated in the command text after a colon (for example, 'city' for :city, 'my city' for :"my city") or a parameter name in a stored procedure or function.

Returns a reference to a SAParam object which is only valid as long as the param object is not deleted by the library.

Use Param method to access a parameter by its name or position (in SQL statement). If, for example, you want to walk through all the parameters use ParamByIndex method.

If parameters were not created before calling Param method the Library creates them implicitly (can query native API if needed and therefore can throw exception on error) and then returns the specified parameter.

Passing a value of name or position which does not specified in the command text will throw an exception.

SAParam object contains full information about a parameter: name, type, size, etc. Values for the input parameters can be assigned to SAParam object.
See also:

- 2.22.1 Param(ID as integer) as SQLParamMBS 107

ParamByIndex(index as integer) as SQLParamMBS

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the command parameter specified by index.
Notes:

Index: A zero-based index of the requested parameter in the array of SAParam objects. It must be greater

than or equal to 0 and 1 less than the value returned by ParamCount method.

Returns a reference to a SAParam object.

Normally you should use Param method to access a parameter by its name or position (in SQL statement). ParamByIndex method can be used if, for example, you want to walk through all the parameters.

If parameters were not created before calling ParamByIndex method the Library creates them implicitly (can query native API if needed and therefore can throw exception on error) and then returns the specified parameter.

Passing a negative value of index or a value greater or equal than the value returned by ParamCount method will result in a failed assertion.

SAParam object contains full information about a parameter: name, type, size, etc. Values for the input parameters can be assigned to SAParam object.

ParamCount as integer

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the number of parameters associated with the SACommand object.
Notes:

ParamCount method returns the number of parameters created explicitly by using CreateParam method or (if parameters were not created before) creates them implicitly (can query native API if needed and therefore can throw exception on error) and returns the number of created parameters.

Command parameter is represented by SAParam object. You can look SAParam objects through and assign their values with Param and ParamByIndex methods.

Prepare

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Prepares command before execution.
Notes:

Prepare method compiles the command, but does not execute it. The method detects syntax errors in command text and verifies the existence of database objects.

Execute method calls Prepare method implicitly if needed, therefore you don't have to call it explicitly.

RowsAffected as integer

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the number of rows affected by the last insert/update/delete command execution.

setCommandText(SQLCommand as string, CommandType as integer=0)

method from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets the command text.
Example:

```
dim s as new SQLCommandMBS

s.setCommandText "select * from test"

MsgBox s.CommandText
```

Notes:

SQLCommand: A string which represents command text string (an SQL statement or a stored procedure name).
CommandType: The type of command like kCommandTypeUnknown, kCommandTypeSQLStatement, kCommandTypeSQLStatementRaw or kCommandTypeStoredProcedure.

It's not necessary to set a command type explicitly, because it is defined automatically in terms of command text string. But if you still have any reason to do it, use one of the kCommandType* constants. To get command type use CommandType method.

2.22.2 Events

Working

event from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.4, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The event called while the ExecuteMT and ExecuteCommandMT methods are running.

2.22.3 Constants

kCommandTypeSQLStatement=1

const from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the constants for the command type.
Notes: Command is an SQL statement.

kCommandTypeSQLStatementRaw=2

const from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the constants for the command type.
Notes: Command is an SQL statement that mustn't be interpreted by SQLAPI.

kCommandTypeStoredProcedure=3

const from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the constants for the command type.
Notes: Command is a stored procedure or a function.

kCommandTypeUnknown=0

const from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the constants for the command type.
Notes: Used by default. Library detects command type automatically.

kOptionPreFetchRows="PreFetchRows"

const from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the option constants.
Example:

```
dim cmd as new SQLCommandMBS
// do something

dim nBulkSize as integer = 1000
cmd.Option("PreFetchRows") = str(nBulkSize)
```

kParamDirTypeInput=0

const from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the parameter direction type constants.
Notes: Input parameter.

kParamDirTypeInputOutput=1

const from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the parameter direction type constants.
Notes: Input/output parameter.

kParamDirTypeOutput=2

const from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the parameter direction type constants.
Notes: Output parameter.

kParamDirTypeReturn=3

const from class SQLCommandMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the parameter direction type constants.
Notes: Returning parameter.

2.23 class SQLDataProviderMBS

class SQLDataProviderMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The class for a data provider.
Notes: Use this to set a blob/clob object with streaming data. For example if you want to add a 1 GB big file to the database without loading it into RAM in one piece, you can use this class to read it in small chunks.

2.23.1 Events

Read(byref PieceType as integer, Length as UInt32) as string

event from class SQLDataProviderMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The event called whenever new data is needed.
Notes:

PieceType is kOnePiece, kFirstPiece, kLastPiece or kNextPiece.
If your stream is on the end, you set this to kLastPiece.
Return the raw data in a string.
Length is the number of bytes.

2.23.2 Constants

kFirstPiece=1

const from class SQLDataProviderMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the piece type constants.
Notes: The first piece is processed. You may setup everything you need to handle the data.

kLastPiece=3

const from class SQLDataProviderMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the piece type constants.
Notes: The last piece is processed. You can close files/network connections.

kNextPiece=2

const from class SQLDataProviderMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the piece type constants.
Notes: The next piece is processed. Not the first one or the last one, but one between.

kOnePiece=4

const from class SQLDataProviderMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the piece type constants.
Notes: The whole data stream is delivered in one call of the event.

2.24 class SQLDateTimeMBS

class SQLDateTimeMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The SQL date/time value class.
Example:

```
dim d as new SQLDateTimeMBS(2008, 3, 4, 23, 10, 20)
```

```
MsgBox d.StringValue // shows "2008-03-04T23:10:20"
```

2.24.1 Methods

Constructor

method from class SQLDateTimeMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The constructor to create a zero time/date.
See also:

- 2.24.1 Constructor(other as SQLDateTimeMBS) 116
- 2.24.1 Constructor(value as double) 117
- 2.24.1 Constructor(Year as integer, Month as integer, Day as integer, Hour as integer, Minute as integer, Second as integer) 117

Constructor(other as SQLDateTimeMBS)

method from class SQLDateTimeMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a copy of the SQL Date Time.
Example:

```
dim d as new SQLDateTimeMBS(2008, 3, 4, 23, 10, 20)
dim e as new SQLDateTimeMBS(d)

MsgBox e.StringValue // shows "2008-03-04T23:10:20"
```

See also:

- 2.24.1 Constructor 116
- 2.24.1 Constructor(value as double) 117

Constructor(value as double)

method from class SQLDateTimeMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new SQL date time value based on the double value.
Example:

```
dim d as new SQLDateTimeMBS(2008, 3, 4, 23, 10, 20)
dim e as new SQLDateTimeMBS(d.DoubleValue+1) // clone with one day more

MsgBox e.StringValue // shows "2008-03-05T23:10:20"
```

See also:

Constructor(Year as integer, Month as integer, Day as integer, Hour as integer, Minute as integer, Second as integer)

method from class SQLDateTimeMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Creates a new SQL Datetime with the given values.
Example:

```
dim d as new SQLDateTimeMBS(2008, 3, 4, 23, 10, 20)

MsgBox d.StringValue // shows "2008-03-04T23:10:20"
```

See also:

DoubleValue as double

method from class SQLDateTimeMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The double value of this date/time.
Notes:

Use these operators to get current date/time value using standard double representation. Days are represented by whole number increments starting with 30 December 1899, midnight as time zero. Hour values are expressed as the absolute value of the fractional part of the number.

Date and time	Representation
30 December 1899, midnight	0.00
1 January 1900, midnight	2.00
4 January 1900, midnight	5.00
4 January 1900, 6 A.M.	5.25
4 January 1900, noon	5.50
4 January 1900, 9 P.M.	5.875

Fraction as UInt32

method from class SQLDateTimeMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the value of the fraction of second (0 to 999,999,999) this SQLDateTime object represents.
Notes:

The value of the fraction field is the number of billionths of a second and ranges from 0 through 999,999,999 (1 less than 1 billion). For example, the value of the fraction field for a half-second is 500,000,000, for a thousandth of a second (one millisecond) is 1,000,000, for a millionth of a second (one microsecond) is 1,000, and for a billionth of a second (one nanosecond) is 1.

GetDay as integer

method from class SQLDateTimeMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the day this SADateTime object represents (1 31).

GetDayOfWeek as integer

method from class SQLDateTimeMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the day of the week this SADateTime object represents (Sunday = 1).

GetDayOfYear as integer

method from class SQLDateTimeMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the day of the year this SADateTime object represents (Jan 1 = 1).

GetHour as integer

method from class SQLDateTimeMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the hour this SADateTime object represents (0 23).

GetMinute as integer

method from class SQLDateTimeMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the minute this SADateTime object represents (0 59).

GetMonth as integer

method from class SQLDateTimeMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the month this SADateTime object represents (1 12).

GetSecond as integer

method from class SQLDateTimeMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the second this SADateTime object represents (0 59).

GetYear as integer

method from class SQLDateTimeMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns the year this SADateTime object represents.

StringValue as string

method from class SQLDateTimeMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The string value for this date/time.

2.25 class SQLDatabaseMBS

class SQLDatabaseMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The database class for the SQL plugin
Notes:

You can use the SQL plugin without using REALbasic built in database classes if you use the SQLConnectionMBS and SQLCommandMBS classes.

Or you use the SQLDatabaseMBS class which is a subclass of the database class and can be used with REALbasics RecordSet class. The current implementation is far from complete. You can connect with passing the database URL in the DatabaseName property of the SQLDatabaseMBS class. You prefix this URL with the database type you are using.

You can use Execute and Select to run SQL statements. Errors can be queries with the lasterror properties. For the RecordSet, you can get the column count, the column names and values and move to the next row. All the other methods like deleting a record or updating a value are not yet implemented and you may use SQL commands to do this.

Supported databases: Oracle, Microsoft SQL Server, DB2, Sybase, Informix, InterBase/Firebird, SQLBase, MySQL, PostgreSQL and ODBC and SQLite
Subclass of the Database class.

2.25.1 Methods

Connect as boolean

method from class SQLDatabaseMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Connects to the database.
Example:

```
dim db as new SQLDatabaseMBS
```

```
// where is the library?
db.SetFileOption SQLConnectionMBS.kOptionLibraryMySQL, SpecialFolder.UserHome.Child("libmysqlclient.dylib")

// connect to database
// in this example it is MySQL,
// but can also be Sybase, Informix, Oracle, DB2
// SQLServer, InterBase, SQLBase and ODBC

db.DatabaseName="mysql:192.168.1.80:3306@test"
db.UserName="root"
db.Password=""

if db.Connect then

MsgBox "We are connected!"

MsgBox "Server Version: "+db.GetConnection.ServerVersionString

// Disconnect is optional
// autodisconnect will ocur in destructor if needed

else
MsgBox db.ErrorMessage
end if
```

Notes:

Returns true on success and false on failure.

Please set the DatabaseName, UserName and Password properties.

The database name must contain the complete information and a prefix for the kind of database.

Use this prefixes: "ODBC:" ,"Oracle:" ,"SQLServer:" ,"InterBase:" "SQLBase:", "DB2:" "Informix:", "Sybase:", "MySQL:", "PostgreSQL:" or "SQLite:".

GetConnection as SQLConnectionMBS

method from class SQLDatabaseMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The connection for this database used in the background.
Notes: Note that methods on this connection object can raise exceptions while methods on the SQLDatabaseMBS class sets the error properties.

Option(name as string) as string

method from class SQLDatabaseMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.5, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets an option for the connection.
Notes: (Read and Write computed property)

SetFileOption(name as string, file as folderitem)

method from class SQLDatabaseMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 10.5, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Sets an option with passing a file path.
Example:

```
dim db as new SQLDatabaseMBS

// where is the library?
db.SetFileOption SQLConnectionMBS.kOptionLibraryMySQL, SpecialFolder.UserHome.Child("libmysqlclient.dylib")
```

Notes: Makes sure the path is correct.

2.26 class SQLDataConsumerMBS

class SQLDataConsumerMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The class for a data consumer.
Notes: If you query a clob/blob value, that value may not fit into memory, so you may prefer to get a callback for data and write it to a file in small chunks.

2.26.1 Events

Write(PieceType as integer, data as string, Length as UInt32, BlobSize as UInt32)

event from class SQLDataConsumerMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The event called to process data.
Notes:

PieceType is kOnePiece, kFirstPiece, kLastPiece or kNextPiece.
Data is the raw data in a binary string.
Length is the number of bytes and BlobSize the size of data blocks used.

2.26.2 Constants

kFirstPiece=1

const from class SQLDataConsumerMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the piece type constants.
Notes: The first piece is processed. You may setup everything you need to handle the data.

kLastPiece=3

const from class SQLDataConsumerMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the piece type constants.
Notes: The last piece is processed. You can close files/network connections.

kNextPiece=2

const from class SQLDataConsumerMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the piece type constants.
Notes: The next piece is processed. Not the first one or the last one, but one between.

kOnePiece=4

const from class SQLDataConsumerMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, not console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: One of the piece type constants.
Notes: The whole data stream is delivered in one call of the event.

2.27 class SQLFieldMBS

class SQLFieldMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: This is the class for a SQL field in a record.
Notes:

Be aware that field objects exists only as long as their SQLCommand exists.

Subclass of the SQLValueReadMBS class.

2.27.1 Methods

FieldNativeType as integer

method from class SQLFieldMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns native type code of the field.

FieldPrecision as integer

method from class SQLFieldMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns precision of the field value (the total number of allowable digits).

FieldScale as integer

method from class SQLFieldMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns scale of the field value (the number of digits to the right of the decimal point).

FieldSize as integer

method from class SQLFieldMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns field data size.
Notes: (Read and Write computed property)

FieldType as integer

method from class SQLFieldMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns field data type.
Notes:

Value is one of the kDataType* constants.
(Read and Write computed property)

isFieldRequired as boolean

method from class SQLFieldMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Shows if it is possible for the field value to be null.
Notes: Returns true if the field value can be null; false otherwise.

Name as string

method from class SQLFieldMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns name of the field.

Option(name as string) as string

method from class SQLFieldMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The string value of a specific field option.
Notes:

See for more details:
http://www.sqlapi.com/OnLineDoc/Field_ setOption.html

(Read and Write computed property)

Pos as integer

method from class SQLFieldMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Returns a one-based position of the field in a result set.

ReadLongOrLob(consumer as SQLDataConsumerMBS, BlockSize as integer)

method from class SQLFieldMBS, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: Starts reading of Long or BLob(CLob) value using the given data consumer.
Notes:

BlockSize: Size of piece of data you want to get to the consumer event.

After a command execution all output parameters are updated by their values, including Long and BLob(CLob) parameters. If you want to control piecewise reading of Long or BLob(CLob) data you should do the following:

Before a command execution set kLongOrLobReaderManual reading mode (see LongOrLobReaderMode) for Long or BLob(CLob) parameters you want to process by a data consumer. After that SQLAPI++ will skip reading output Long and BLob(CLob) parameters that you set to be read manually.
After command execution use ReadLongOrLob method for each output parameter defined to be read manually.
Note, that if the command has result set(s) (it is possible in some servers, see Server specific notes) then output parameters are available only after all result sets are completely processed using FetchNext method.

2.28 class SQLErrorExceptionMBS

class SQLErrorExceptionMBS

class, SQL, MBS REALbasic SQL Plugin (SQL), Plugin version: 9.3, console safe, Mac OS X: Works, Windows: Works, Linux x86: Works.

Function: The error exception class to report SQL errors.
Notes:

The SQLDatabaseMBS class sets its error properties on an error. All other SQL classes raise exceptions where you can check the message property.
Subclass of the RuntimeException class.

Chapter 3

List of all classes