**Events**

Events

Event Change()  
- Occurs when the contents of a control have changed.

Event Click()  
- Occurs when the user presses and then releases a mouse button over an object.

Event DblClick()  
- Occurs when the user presses and releases a mouse button and then presses and releases it again over an object.

Event DragDrop(Source As Control, X As Single, Y As Single)  
- Occurs when a drag-and-drop operation is completed.

Event DragOver(Source As Control, X As Single, Y As Single, State As Integer)  
- Occurs when a drag-and-drop operation is in progress.

Event GotFocus()  
- Occurs when an object receives the focus.

Event KeyPress(KeyAscii As Integer)  
- Occurs when the user presses and releases an ANSI key.

Event LostFocus()  
- Occurs when an object loses the focus.

.....etc

If...Then...Else Statements

You can use If...Then...Else statements to execute a specific statement or block of statements depending on the Boolean value of a condition. The condition usually results from a comparison of two values, but it can be any expression that evaluates to a Boolean value (True or False). This includes values of other data types, such as numeric types, that have been converted to Boolean.

Executing Statements if a Condition Is True

To execute only one statement when a condition is True, you can use the single-line syntax of If...Then...Else, omitting the Else and End If statements

Syntax:

If condition Then

....statement

end if

Example

IF text1.text = 1 Then

print "one"

else

Print "not one"

End If

Lesson 8

DATA TYPES

• Boolean

The Boolean data type has only two states, True and False. These types of variables are stored as 16-bit (2 Byte) numbers, and are usually used for flags.

• Byte

The Byte data type is an 8-bit variable which can store value from 0 to 255. This data type is very useful for storing binary data. It can also be very useful when sending/receiving byte values to/from a Basic Stamp or PIC.

• Double

The Double data type is a 64-bit floating point number used when high accuracy is needed. These variables can range from -1.79769313486232e308 to -4.94065645841247e-324 for negative values and from 4.94065645841247e-324 to 1.79769313486232e308 for positive values.

• Integer

The Integer data type is a 16-bit number which can range from -32768 to 32767. Integers should be used when you are working with values that can not contain fractional numbers.

• Long

The Long data type is a 32-bit number which can range from -2,147,483,648 to 2,147,483,647. Long variables can only contain non-fractional integer values. I myself use Long variables over Integers for increased performance. Most Win32 functions use this data type for this reason.

• Single

The Single data type is a 32-bit number ranging from -3.402823e38 to -1.401298e-45 for negative values and from 1.401298e-45 to 3.402823e38 for positive values. When you need fractional numbers within this range, this is the data type to use.

• String

The String data type is usually used as a variable-length type of variable. A variable-length string can contain up to approximately 2 billion characters. Each character has a value ranging from 0 to 255 based on the ASCII character set. Strings are used when Text is involved.

--------------------------------------------------------------------------

Boolean

Boolean

Depends on implementing platform

True or False

Byte

Byte

1 byte

0 through 255 (unsigned)

Char (single character)

Char

2 bytes

0 through 65535 (unsigned)

Date

DateTime

8 bytes

0:00:00 (midnight) on January 1, 0001 through 11:59:59 PM on December 31, 9999

Decimal

Decimal

16 bytes

0 through +/-79,228,162,514,264,337,593,543,950,335 (+/-7.9...E+28) † with no decimal point; 0 through +/-7.9228162514264337593543950335 with 28 places to the right of the decimal;

smallest nonzero number is +/-0.0000000000000000000000000001 (+/-1E-28) †

Double (double-precision floating-point)

Double

8 bytes

-1.79769313486231570E+308 through -4.94065645841246544E-324 † for negative values;

4.94065645841246544E-324 through 1.79769313486231570E+308 † for positive values

Integer

Int32

4 bytes

-2,147,483,648 through 2,147,483,647 (signed)

Long (long integer)

Int64

8 bytes

-9,223,372,036,854,775,808 through 9,223,372,036,854,775,807 (9.2...E+18 †) (signed)

Object

Object (class)

4 bytes on 32-bit platform

8 bytes on 64-bit platform

Any type can be stored in a variable of type Object

SByte

SByte

1 byte

-128 through 127 (signed)

Short (short integer)

Int16

2 bytes

-32,768 through 32,767 (signed)

Single (single-precision floating-point)

Single

4 bytes

-3.4028235E+38 through -1.401298E-45 † for negative values;

1.401298E-45 through 3.4028235E+38 † for positive values

String (variable-length)

String (class)

Depends on implementing platform

0 to approximately 2 billion Unicode characters

UInteger

UInt32

4 bytes

0 through 4,294,967,295 (unsigned)

ULong

UInt64

8 bytes

0 through 18,446,744,073,709,551,615 (1.8...E+19 †) (unsigned)

User-Defined (structure)

(inherits from ValueType)

Depends on implementing platform

Each member of the structure has a range determined by its data type and independent of the ranges of the other members

UShort

UInt16

2 bytes

0 through 65,535 (unsigned)