Expression Binding

An expression binding is one of the most powerful kinds of property bindings.

It uses a simple expression language to calculate a value. This expression can involve lots of dynamic data, such as other properties, tag values, results of Python scripts, queries, and so on.

You can use expressions for many different purposes. Anytime information needs to be massaged, manipulated, extracted, combined, split, and so on - think expressions!

Example 1

You have 3 bits in a PLC, only one of which will be on at a time. You want to turn these three bits into a single integer (0, 1, 2) to drive a component's Styles. Bind a custom integer property to:

```
binEnum{{MyTags/Bit1}, {MyTags/Bit2}, {MyTags/Bit3}}
```

Example 2

You have a Date, and need to extract the year, and concatenate the word "Vintage" to the end for a label display. Bind a label's text property to:

```
dateExtract{{Root Container.VintageDate}, 'year')} + ' Vintage'
```

Example 3

You have a button that starts a batch, but you only want to let it be pressed after the operator has entered a scale weight. Bind the button's enabled property to:

```
{Root Container.EntryArea.WeightBox.doubleValue} > 0.0
```

Example 4

You want to display a process's current state, translating a code from the PLC to a human-readable string, use of these two expressions (they're equivalent)
if ({CurrentProcessState} = 0, "Not Running",
if ({CurrentProcessState} = 1, "Warmup phase - please wait",
if ({CurrentProcessState} = 2, "Running",
"UNKNOWN STATE")))

- or -

switch ({CurrentProcessState},
  0,1,2,
  "Not Running",
  "Warmup phase - please wait",
  "Running",
  "UNKNOWN STATE")

See also: Expression Overview and Syntax

Next ...

- Expression Binding - Concat Strings