Text box HTML syntax and formatting strings

Text box input HTML syntax and formatting strings provide you with a great deal of flexibility in controlling the appearance of your data on the web page. The formatting string is dynamically processed when your web page is displayed, and the formatting string is replaced with actual real time data.

A text input should be constructed as follows:

<input type="text" name="reg22" value="%d" readonly size="8">

The type for a text box is always "text", and name identifies the register data is formatted from or parsed to. The value field specifies how the data should be formatted in converting it to a printable string. Size specifies the size of the box on the page. The "readonly" tag is optional, and may be used to disallow input when data should displayed only and never changed by the user.

The value field in text boxes is used to tell the CGI processor how to format real time data when presenting it as content in a User HTML page. The format specifier is the printf format specifier as found in C or C++. The syntax is

%[flags] [width] [.precision] type

where width and precision are numeric values, and flags and type are as follows:

type	.
d	integer
u	unsigned integer
f	floating point
x	hexadecimal unsigned integer
s	character string
flags	.
0	insert leading zeroes
#	insert 0x etc for hex

The width and precision are optional. Width is the number of digits that will be formatted. Precision is the number of digits after the decimal point, only applicable for floating point. You rarely have any reason to use anything other than just "%d" for integer values. But the full printf capability of C is available, so you can create format strings like "%d %%RH" which for a value of 35 would be printed as "35 %RH"

The width and precision are more useful for floating point. Since floating point is not exact, you will want to specify precision to keep the number "user friendly". Without the

rounding forced by precision, you can see something like 24.999998 when you expected 25. The way to fix this is force precision of just one or two digits after the decimal point so that you get 25.00 instead.

Examples of some floating point specifiers and the result starting with a value of 12.35912:

%.2f	12.36
%f	12.35912
%03.2f	012.36
%.f	12

Character strings are formatted with a simple "%s" but you can limit the number of characters displayed. For example, "%10s" will display only the first 10 characters of the string regardless of its actual length. Character string formatting is only useful for register names in User HTML.

Article ID: 23
Created On: Thu, Sep 10, 2015 at 12:05 PM
Last Updated On: Thu, Sep 10, 2015 at 12:05 PM