

JobInfo Substitution does not support special characters

- 2021-05-07 - Comments (0) - Lاسernet Developer FAQs

Lasernet

Due to rules set in C++, JobInfo names cannot have special characters in them for use with JobInfo Substitution(JobInfo name inside hashtags to return value). Read more about JobInfos substitution in Manual for Lاسernet Developer: 1.33.4 JobInfo Substitution or at the end of this article in Additional Information.

Examples

Allowed:

Username

_Username

_1Username

User_name

U1sername

Not allowed:

1username

User name

User-name

User.name

user/name

User:name

User,name

User%name

User&name

User@name

...and many more.

Here are some of the basic rules:

- Variable names in C++ can range from 1 to 255 characters.
- All variable names must begin with a letter of the alphabet or an underscore(_).
- After the first initial letter, variable names can also contain letters and numbers.
- No spaces or special characters are allowed.

Further information can be found here:

<https://www.tutorialspoint.com/What-are-the-rules-to-declare-variables-in-Cplusplus>

Additional Information

Since the JobInfos are an essential part of a Job it is possible to assign values to them in many different ways and at multiple locations. When assigning a value to a JobInfo you must specify the name of the JobInfo as well. The value does not have to be text. Special syntax exists which allows for greater functionality. This is generally known as substitution. For example, this allows you to write:

```
The job came from #InputPort#
```

as a value.

This means that Lasernet will look up the value in the InputPort JobInfo and replace #InputPort# with that value. This can be done with more than one JobInfo:

```
The job came from #InputPort# and was created at #JobTime#
```

The #JobInfoName# syntax has a few more possibilities. Since a JobInfo can also be a list, you can choose to access a specific value in the list, by adding an index to the substitution:

```
The job came from #Source[0]#
```

You can also choose to expand the list with a custom delimiter:

```
The job has passed through these modules: #Source{,}#
```

To facilitate the use of # as a value, an escape character is used - The Generic Currency Symbol - ¤ - Ascii A4. Putting a ¤ before the # allows you to use # as a value. To get a ¤ as a value, simply use two: ¤¤ If you only need one # simply put that in. As long as there is no matching # LAsernet assumes you just want the one # In the following samples |NAME| means the contents of the JobInfo with the name NAME.

```
abc¤###def => abc##def abc¤¤#def => abc¤#def abc¤¤#def# => abc¤|def|
```

JobInfo substitution is not recursive, so if a jobinfo name is put in a jobinfo value, only the outer jobinfo is substituted e.g., if there are two JobInfos: Inner=Value, Outer=#Inner#,

typing "#Inner#" in a field with substitution will yield Value, while putting "#Outer#" will yield "#Inner", which will not be substituted further.

Due to historical implementation issues there may be some places in Lasernet where this approach does not work.