
Subject: Little Quiz

Posted by [mirek](#) on Thu, 09 Dec 2021 08:57:18 GMT

[View Forum Message](#) <> [Reply to Message](#)

What does this mean:

DropList dl;

...

if(*~~dl == '\$') ...

Subject: Re: Little Quiz

Posted by [Tom1](#) on Thu, 09 Dec 2021 09:19:54 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi,

Is it the short version for e.g.:

```
if(((const char*)dl.Get())[0] == '$')
```

?

Best regards,

Tom

Subject: Re: Little Quiz

Posted by [mirek](#) on Thu, 09 Dec 2021 09:51:36 GMT

[View Forum Message](#) <> [Reply to Message](#)

Tom1 wrote on Thu, 09 December 2021 10:19Hi,

Is it the short version for e.g.:

```
if(((const char*)dl.Get())[0] == '$')
```

?

Best regards,

Tom

Almost correct.

Subject: Re: Little Quiz

Posted by [Lance](#) on Fri, 10 Dec 2021 21:22:18 GMT

[View Forum Message](#) <> [Reply to Message](#)

the inner ~ is
DropList::operator ~
which inherits
Value Ctrl::operator ~()(maybe const modifier?)

the outer ~ is
String Value::operator~()(const modifier?)

the * cause a auto type cast be called

String::operatr const char *()(modifier?);

and then dereference of the returned constant char * returns the first char in the String.

something like

```
Value t1=~dl;  
String t2=~t1;  
const char* t3=(const char*)t2;  
char t4=*t3;  
if(t3=='$')
```

Subject: Re: Little Quiz

Posted by [mirek](#) on Mon, 13 Dec 2021 14:53:56 GMT

[View Forum Message](#) <> [Reply to Message](#)

* is String::operator*

Otherwise solved

Subject: Re: Little Quiz

Posted by [Lance](#) on Sat, 18 Dec 2021 04:47:35 GMT

[View Forum Message](#) <> [Reply to Message](#)

Didnot realise there is an operator * defined.

BTW, if there is no operator *, will the route suggested by my previous post be taken? IE, will * cause a implicit type case operator by called? I guess so.

Subject: Re: Little Quiz
Posted by [mirek](#) on Sat, 18 Dec 2021 08:07:10 GMT
[View Forum Message](#) <> [Reply to Message](#)

It indeed is. I guess that operator* is there from times when compilers were not up to the task.
