

Re: finding common words

Source: <http://coding.derkeiler.com/Archive/Perl/comp.lang.perl.misc/2004-02/3026.html>

From: Hunter Johnson (hunter_at_hunterandlori.com)

Date: 02/25/04

Date: 25 Feb 2004 05:07:09 -0800

Uri Guttman <uri@stemsystems.com> wrote in message news:<x7d6842rkf.fsf@mail.sysarch.com>...

> >>>>> "HJ" == Hunter Johnson <hunter@hunterandlori.com> writes:

>

> HJ> Uri Guttman <uri@stemsystems.com> wrote in message news:<x74qth75vw.fsf@mail.sysarch.com>...

> >> now the docs will say there is no

> >> generated list in void context so that may be harder to argue. but i

> >> still say that map tells the reader a list is being made so look for

> >> it.

>

> HJ> If the docs say there is no list generated, how is map going to tell

> HJ> the reader that the docs are wrong?

>

> HJ> If a reader reads something extra into what's written (like "here

> HJ> comes a list" when none is coming), how is that more the writer's

> HJ> problem than the reader's? Either the writer can write it

> HJ> differently (using 'for' in this case) or the reader can read it

> HJ> differently. I don't see how the former is the only right answer.

>

> it is a semantic communication to the reader of the code. it is your

> (the coder's) responsibility to convey as much accurate information to

> the reader as possible. map in a void context is not as accurate as a

> for modifier even with the optimization.

Yes, it is because, as you say:

> sure the docs will say it won't generate a list

Of course, you then qualify that with:

> but its history has always been that way.

Historically, map wasn't even in perl, but people use it with its current implementation anyway. That's all I'm suggesting will happen now -- no list in void context means that programmers can (and should, as they like) use it in void context, and readers will have to learn that the language has changed, which it has.

comp.lang.perl.misc: Re: finding common words

Hunter

--

<http://www.hunterandlori.com>