

perl.beginners: RE: variable declaration, was RE: moving to the next line

RE: variable declaration, was RE: moving to the next line

Source: <http://coding.derkeiler.com/Archive/Perl/perl.beginners/2004-12/0669.html>

From: Charles K. Clarkson (*cclarkson_at_htcomp.net*)

Date: 12/17/04

To: "'Perl Beginners List'" <beginners@perl.org>

Date: Fri, 17 Dec 2004 12:51:49 -0600

Chris Devers <cdevers@pobox.com> wrote:

: On Thu, 16 Dec 2004, Charles K. Clarkson wrote:

:

:: Don't declare all your variables at the beginning
:: of the script. It works in other languages, but not
:: in perl. Declare them as you go.

:

: Out of curiosity, why this rule?

:

: When taking programming classes in college, it was drummed into
: us that having a data dictionary at the top of a scope was a
: good habit, and it's something that I've generally done with the
: Perl I've written.

I took one college programming course. We used PL/1 on a Multics mainframe. Personal Computers didn't yet exist. I don't recall how variables were declared. :)

: Several people on this list have discouraged the habit. How
: come?

I discourage it mainly because it creates an opportunity for global variables and for unused variables to remain declared.

: I can see the logic in discouraging global variables, but
: predeclaring variables at the top of a scope --- the beginning of
: a subroutines, and a small handful in the main block --- still
: seems acceptable to me. Indeed, waiting to declare until the
: variable is used is, to my thinking, kind of defeating the point
: of using 'strict': if the declarations are scattered all over
: the place, why bother being strict?

I read a college text on programming once. It referred to programming in flow charts and then translating the problem from

RE: variable declaration, was RE: moving to the next line

perl.beginners: RE: variable declaration, was RE: moving to the next line

the charts to the language of choice. Actually I think there was an intermediate written language in there also. I doubt anyone does this with perl.

We tend to write scripts with perl the same way we write essays for book reports. On this list, I assume we don't have many beginners which know the basic algorithm and data structure of their script before opening their text editor.

Beginner perl scripts tend to be written as we go. We start with basic I/O and logic and then fill in the necessary parts. This style of scripting means a lot more rewriting before the product is finished. In fact, we can assume that any script offered here is a work in progress, not a finished application.

The rules I tend to enforce during code review are a matter of personal style and of learned programming habits. I'm fortunate that few others replying here have the time to review code from this list, making me an authority. Unfortunately, that means I am not challenged on my rules.

I agree that a data dictionary is a great resource for a finished product, but it is also an advanced programming concept. Every time I review code I chance alienating the author. It has happened before. The tone which creeps in after a particularly long script is invariably negative and the archives prove how many readers dislike that.

When I enforce a rule for code review here on this list I tend to use a shotgun approach. I may never hear from this person again. I want to hit as many of those habits which lead to the worst offenses. Namely global variables, poor data structures, poor I/O handling, poor algorithm design, etc.

I need to stop those practices which often lead problems down the road. I have to be short-winded enough to not lose my audience and long-winded enough not to leave out important ideas. Hopefully, other readers of my reply will ask questions about specific topics, as you have done.

Intermediate and advanced programmers should probably use any technique to make their scripts easier to read and to maintain. Beginners need to work on their basic skills and pre-declaring all variables at the top of scope usually opposes that effort.

HTH,

Charles K. Clarkson

--

Mobile Homes Specialist

RE: variable declaration, was RE: moving to the next line

perl.beginners: RE: variable declaration, was RE: moving to the next line

254 968-8328