

## Read, Read, Code, Read, Code, Code

by Dr. David J. Ritchie,  
Computer Club Advisor

Just as with reading and writing other languages, the best way to improve in perl is to read and write it.

### Try it:

Write a program a week over the summer.

### Ideas:

1. Make a perl program that does "knock knock" jokes.
2. Extend the "Not So Great Escape" game to be a Mammoth Cave Game
3. Simulate your vacation. Make random events occur to extend (or shorten) your vacation.

For (hint, hint) each day of your vacation, add the gas and food used to a running total....

## CPAN

Comprehensive  
Perl  
Archive  
Network

a good place to look for Perl things

<http://www.perl.com/CPAN>

TO: «First Name» «Last Name»  
c/o Home Room: «Home Room»

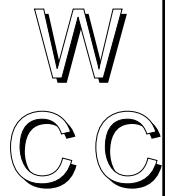
Issue 6

May 24, 1999

## Madison Computer Club News

Mrs. Gilmore

Dr. David J. Ritchie



## Students To Escape!

We've all found our way to the end of the Computer Club sessions.

Perl is a language with many resources. Try exploring the perl web site at <http://www.perl.com/perl/>.

Look at some of the perl books at Borders, Books and Bytes, and Barnes and Noble.

See what sorts of programs you can find on CPAN--the Comprehensive Perl Archive Network at <http://www.perl.com/CPAN/> Look in the modules sub-directory in particular.

Many of the things that you will find on CPAN were given to the world in the same spirit that Perl was given to the world.

There is a sense in which all of us, having been the recipients of such gifts, should give something back.

So, try to write a fun game, a neat program, and give it back to the community which brought you perl in the first place.

You will find the instructions for doing that on the CPAN web site.

## Things We Haven't Taught You...

One of the great things about perl is that it has many levels of use from very simple to complex.

- push - pushes onto an array a list of items adding them to the end of the array.

### Quoting lines and lines of text

```
# use the "here" document to quote lots  
# of text
```

```
$Variable = 1.23
```

```
print <<EOF
```

```
Everything on this line up until the  
text after the << that appears a line by  
itself will be considered quoted. In  
this case, it will be printed and  
variables like $Variable will be  
interpolated. It's easier than quotes!  
EOF
```

- reverse - reverses the order of the elements in a list or an array.
- split - splits a string into a separate individual strings using patterns of regular expressions to

You should now be able to pick up almost any book on perl and piece your way through it picking up interesting and useful information.

define the split.

A good place to start are the libraries. Here are some examples of interesting library routines and what they do:

In addition, there are ways you can use perl to do "object-oriented" programming for those who want to explore that kind of thing.

- localtime - returns an array containing various parts of the current time, such as the second, minute, hour, etc.
- pop - pops the last value of an array out of the array and into your variable

Lastly, there is a version of perl which is combined with Tk which lets you do more graphics sorts of things than straight perl.

Now, it's over to you to educate yourself.

**Good Luck!**