

Helpful Resources

And how to use them effectively

What's Available

- If you need help with an assignment, or don't understand a Perl feature, there are several resources available, in a variety of formats:
- Books
 - Course Texts (Camel, Llama, Alpaca)
 - Other suggested (next slide)
- Built-In Documentation
- Instructor and TAs
 - Office hours, Email, AIM
- Web
- Usenet

Books

- Suggested books, in addition to course texts:
 - CGI Programming with Perl
 - CGI Reference
 - Perl Cookbook
 - Companion to the Camel - Common programming tasks
 - Learning Perl for Win32 Systems
 - Outdated, but helpful if you hate Unix
 - Mastering Regular Expressions
 - RegExp bible – but not only Perl
 - Mastering Perl/Tk
 - for GUI programming
 - Perl Best Practices
 - Tips and suggestions for clean maintainable code

Built In Documentation

- **perldoc <subject>**
 - Common subjects: `perlop`, `perlvar`, `perlfunc`
 - `perldoc perltoc` for a full list
- **perldoc -f <function_name>**
- **perldoc -q <FAQ keyword>**
 - `perldoc perlfaq` for the full FAQ
- If you install ActivePerl, all perldocs will be available to you via:
 Start → Programs → ActiveState ActivePerl
 5.10 → Documentation

Instructor and TAs

- I welcome all requests for help, with one large caveat: You must have made a genuine effort to help yourself first!
- Specifically:
 - Check the documentation for the functions you’re using
 - Check the PerlFAQ
- If I get the feeling you haven’t done either of those, I may respond with a pointer to the documentation.
 - more polite than “RTFM!”
 - Don’t get offended by this. (“Teach a man to fish...”)
- The quality of my answer is directly proportional to the quality of your question.

Effective Emailing

Bad Idea	Good Idea
“It doesn’t work”	“I received this error: ‘...’ “It output 5 instead of 5.8”
Speaking English (“I have an array with foo”)	Speaking Perl (“I have: <code>@arr = ('foo');</code>)
“The push function doesn’t work”	“The docs say push should do this, but I’m getting that”
“It gave some error message”	“It gave ‘Use of uninitialized value in prog1.pl at line 23’”
Not enabling strict and warnings	Enabling strict and warnings
“r u sposed 2 do this?”	“Are you supposed to do this?”

Beware the XY Problem!

- “I want to do X”
- “I think Y is a good way to do X”
- “I don’t know how to do Y”
- “I’ll email the instructor to ask how to do Y”
- Ex:
 - You want to get the last modified date of a file. You know you can get this by parsing the output of the `ls` command, but you don’t know how to get that output.
 - You should ask how to get the last modified date, not how to capture the output of `ls`
 - (The answer is in `perldoc -f -x`)

Questions I Don’t Want to Ask

And how to avoid me asking them

- “Have you read the docs for that function?”
 - “I don’t understand this part of the docs for this function”
- “What happened when you tried it?”
 - Don’t ask: “Will this work?”; try it first and see!
- “What’s the error message?”
 - Give me any error messages the code produces

I am a Human Being

- So asking me to do the work of a computer is a very poor use of resources
- While not strictly required for homework, `use strict;` and `use warnings;` are required for asking for help.
 - This way, I don’t have to give answers like “You have a variable `$bigone`, but you spelled it `$big_one` on line 42”
- If you don’t understand a warning message, check the docs *before* emailing me to ask about it!
 - `perldoc perldiag`
 - `use diagnostics;`

World Wide Web

- <http://www.perl.com>
- <http://search.cpan.org>
- <http://learn.perl.org>
- <http://www.google.com>

Usenet

- comp.lang.perl.misc
- Read their Posting Guidelines before posting!
 - posted twice a week – search groups.google.com if you can't find them.
- Paul frequents this group
 - don't try to ask them to do your homework for you
