

Perl

The *Friendly* Programming Language.

A short talk to introduce the Perl programming language and its **culture**.



History

- Perl is a programming language created by Larry Wall in 18 December 1987
- Perl 5 released on 17 October 1994
- Perl 5.8.8 released 2 February 2006
- Perl 6 still to be released...



Perl's Parents

- Perl is a language that unashamedly takes ideas from other languages
- awk
- sed
- shell
- C
- Lisp



Compiled or Interpreted?

- Perl source code is compiled very quickly into a syntax tree
- The syntax tree is automatically optimised
- Finally the optimised syntax tree is executed
- Perl is compiled and interpreted just like other modern languages such as Java.



“hard tasks easy and seriously non-trivial tasks possible”

- Perl is designed to make life easy for the programmer
- Autovivication of variables
- Dynamic typing
- Automatic memory management
- Hashes
- Procedural, object-oriented or both
- Many, many built-in convenience features



Hello World

```
print 'Hello World';
```



Hello World MK-2

```
#!/usr/bin/perl
```

```
use strict;
```

```
use warnings;
```

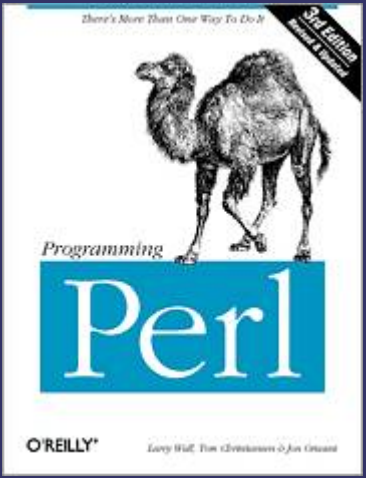
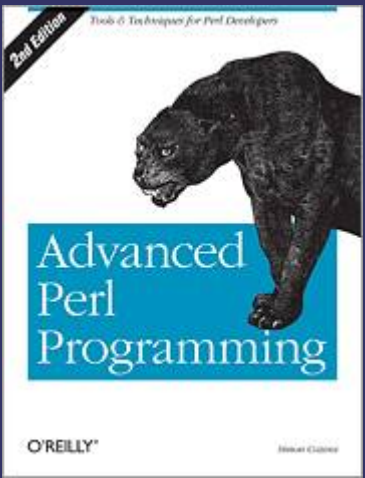
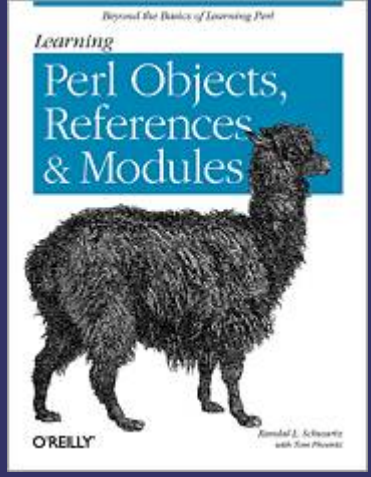
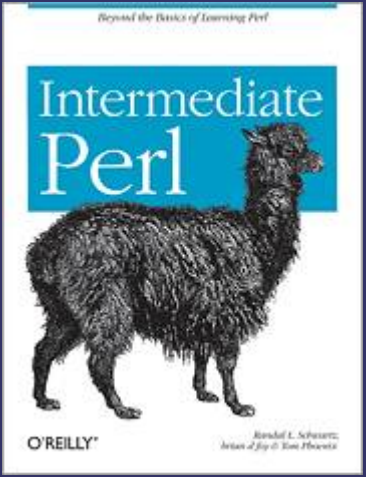
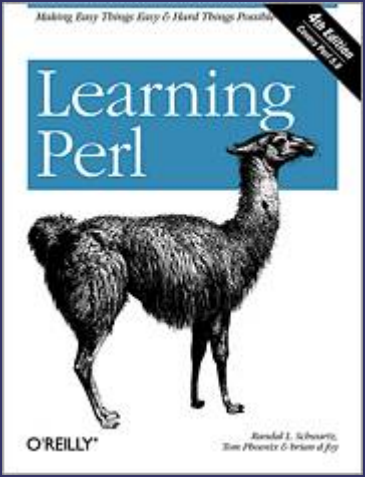
```
my $greeting = "Hello World\n";
```

```
print $greeting;
```

```
exit;
```



Books





Comprehensive Perl Archive Network

- Over 10,000 modules
- A module for every imaginable task:
 - Web/CGI
 - Databases
 - SAP
 - MP3/Oggs
 - Time/Dates
 - Templating
 - and much much more!





CPAN

- Global system with mirrors worldwide.
- PAUSE
 - Perl Authors Upload SErver
- RT
 - Request Tracker
- Search/Kobe Search
 - Two Web Based Search engines



Killer Modules!

- CGI
- DBI
- Template
- CPAN
- Apache mod_perl
- File::Find::Rule
- LWP/WWW::Mechanize



Useful Resources

- Perl Training Australia:
 - <http://perltraining.com.au/notes.html>
- Ovid's CGI Course:
 - http://users.easystreet.com/ovid/cgi_course/
- Learn.perl.org:
 - <http://learn.perl.org/>
- Online Documentation:
 - <http://perldoc.perl.org/>
- Randal's Perl Columns:
 - <http://www.stonehenge.com/merlyn/columns.html>



Community Sites

- PerlMonks:
 - <http://www.perlmonks.org/>
- Use Perl;
 - <http://use.perl.org/>
- Perl Foundation:
 - <http://www.perl.org/>
- Perl Mongers:
 - <http://www.pm.org/>



More Links...

- <http://www.cpan.org/>
- <http://search.cpan.org/>
- <http://kobesearch.cpan.org/>
- <http://rt.cpan.org/>
- <http://pause.perl.org/>
- <http://cpants.perl.org/>
- <http://perldoc.perl.org/>
- <http://www.perl.com/>





"be consistent."

Search **Perl Monks**

The Monastery Gates

- [Log in](#) | [Create a new user](#) | [The Monastery Gates](#) | [Super Search](#) |
- [Seekers of Perl Wisdom](#) | [Meditations](#) | [PerlMonks Discussion](#) | [Snippets](#) |
- [Obfuscation](#) | [Reviews](#) | [Cool Uses For Perl](#) | [Perl News](#) | [Q&A](#) | [Tutorials](#) |
- [Code](#) | [Poetry](#) | [Recent Threads](#) | [Newest Nodes](#) | [Donate](#) | [What's New](#) |

(#131=superdoc: [print w/replies](#), [xml](#))

Donations gladly accepted
If you're new here please read [Perl Monks Site FAQ](#)
and [Create a new user](#).

[Need Help??](#)

Login:
Password:
 remember me
[password reminder](#)
[Create A New User](#)

Want Mega XP? Prepare to have your hopes dashed, join in on the: [poll ideas quest](#) (11534 days remain)

New Questions

[Strange interaction between command line and encoding](#) [3 replies](#) by [VSarkiss](#)
on Jun 02, 2006 at 12:57

This might be blatantly obvious, but nobody in the CB seemed to spot the problem, so I thought I'd ask here.

I have some UTF-16 files I need to convert to UTF-8, so I wrote this tiny program:

```
#!/perl
use encoding "utf16", STDOUT => "utf8";
while (<=>) { print }
```

It works fine. Then I realized I could just do it all from the command line:

```
perl -Mencoding=utf16,STDOUT,utf8 -p -e 1 < in > out
```

Much to my surprise the output isn't UTF-8, it's some strange UTF-16-ish thing (the file is hosed, basically).

Just for grins, I even tried

```
perl -Mencoding=utf16,STDOUT,utf8 -n -e print < in > out
```

But it had the same results

[\[Offer your reply\]](#)

Chatterbox

[\[enhering\]](#): Hello, everybody. I'd like to ask two questions: 1 - If the right way to read and set cookies on different browsers is by using HTTP::Cookies::etc or if any other modules (CGI::Cookies) know how to deal with this problem.

[\[mutatedgene\]](#): ok so another question... now that my program works .. when i set the \$ENV variable in my perl script, how does the module LWP Simple recognize that variable from my script?? im not passing it to any of the functions in the variable

[\[enhering\]](#): 2 - How do I set a cookie using HTTP::Cookies?

[\[jimbus\]](#): here's the http-ssl.conf that included in httpd.conf: [my scratchpad](#)

[\[bart\]](#): The environment variables are like super-globals. Usually they're set outside a script for all programs to use. See for



Where 2.0 Conference June 13-14, 2006, San Jose, CA

Search

Advanced Search >

Perl.com
 Downloads
 Documentation
 CPAN Mirror
 FAQs
 Training
 Resources
 Article Archive
 O'Reilly Books
 Search

Perl Columns
 PSP Digest
 P6P Digest
 Perl Success Stories

Sites
 codezoo.com
 Databases
 Digital Media
 Emerging Telephony
 LinuxDevCenter.com
 MacDevCenter.com
 Mozilla DevCenter
 ONJava.com
 ONLamp.com
 Apache
 BSD
 MySQL
 PHP
 Python
 Security
 OpenP2P.com
 Perl.com
 Policy DevCenter
 Ruby



[A DSL for web client programming](#)



[Writing server-side functions in Perl](#)



[Programming Perl among the penguins](#)

[Unraveling Code with the Debugger](#)

Reading other people's code can be difficult, especially if you have no idea what happens when and where. Understanding code flow is vital to maintenance and bug fixes, but littering code with print and debugging statements is tedious and prone to error. There's another way: use the debugger! Daniel Allen demonstrates how to pinpoint a problem with Perl's debugger. [\[Perl.com\]](#)

[Using Ajax from Perl](#)

The recently rediscovered Ajax technique makes the client side of web programming much more useful and pleasant. However, it also means revising your existing web applications to take advantage of this new power. Dominic Mitchell shows how to use CGI::Ajax to give your Perl applications access to this new power. [\[Perl.com\]](#)

[Perl 5 Summary for 15-21 May 2006](#)

Perl 5 Summary for 15-21 May 2006 [\[Perl 5 List Summaries\]](#)

[Perl 5 Summary for 8-14 May 2006](#)

Perl 5 Summary for 8-14 May 2006 [\[Perl 5 List Summaries\]](#)

[Perl 5 Summary for 1-7 May 2006](#)

Perl 5 Summary for 1-7 May 2006 [\[Perl 5 List Summaries\]](#)

[Advanced Subroutine Techniques](#)

Subroutines seem like a basic building block of code. They're simple and easy to understand and use, right? That's true--but there are a few advanced techniques to make your code more maintainable and robust. Rob Kinyon goes beyond making sense of subroutines to making subroutines work for you. [\[Perl.com\]](#)

[A Canary Trap for URI Escaping](#)

As web applications talk to each other more and more, the issue of URI escaping becomes more important--and more difficult. Escaping reserved characters correctly is vital to preserving user security and proper behavior, but it's neither idempotent nor free. Robert Spier shows how to build a canary trap into your URIs to help you escape and unescape data appropriately, effectively, and sufficiently. [\[ONLamp\]](#)



You want to create searches at runtime. For example, you want users of your program to be able to specify combinations of columns and allowable ranges of values.

[Do it now.](#)



- [Stable is 5.8.8.](#)
- [Latest is 5.8.8.](#)
- [Devel is 5.9.3.](#)



[Refactoring Everything, Day 27](#) This 30-day project explores the refactoring of a legacy system. The Everything Engine is an... *chromatic*

[Installing Ubuntu Dapper: The Good, The Bad, The ... well, you know](#) I've been all but gushing about Ubuntu since my first

Sponsored By:

London perl m[ou]ngers

[Home](#) [About Us](#) [Meetings](#) [People](#) [Reviews](#) [Admin](#) [Subscriptions](#) [View Source](#)

London Perl Mongers

So, what is this all about ?

We are a group of people dedicated to the encouragement of all things Perl-like in London. This involves helping each other, [discussing](#) topics, sharing information and the occasional drink and mention of Buffy the Vampire Slayer.

We are part of the world wide [Perl Mongers](#) group and play host to (as well as visit) others from round the world..

Next Official Meeting

Find out more on the [meetings](#) page. In the meantime, why not [join](#) the mailing list, or hang out on IRC?

The site

© 2003-2006 on everything, in a GNU type of way. Created with Template Toolkit, XML::XPath and Perl!

There are several sections which are not finished on this site. These will be finished ASAP.

Any errors, additions or (and there will be lots) spelling mistakes contact: leo at london.pm.org.

[Home](#) [About Us](#) [Meetings](#) [People](#) [Reviews](#) [Admin](#) [Subscriptions](#) [View Source](#)

Page Last Modified: 17:13:18 21-Mar-2006

London perl m[ou]ngers



[Home](#) · [Authors](#) · [Recent](#) · [News](#) · [Mirrors](#) · [FAQ](#) · [Feedback](#)

atrickett
in

- | | | | | | | | |
|--|-----------------------------|----------------------------|---|-------------------------|---|------------------------|----------------------------|
| Archiving | Compression | Conversion | File Name Systems | Locking | Option Parameter | Config | Processing |
| Bundles (and SDKs) | | | Graphics | | Perl6 | | |
| Commercial Software Interfaces | | | Internationalization | Locale | Pragmas | | |
| Control Flow Utilities | | | Language Extensions | | Security | | |
| Data and Data Types | | | Language Interfaces | | Server Daemon Utilities | | |
| Database Interfaces | | | Mail and Usenet News | | String Language Text Processing | | |
| Development Support | | | Miscellaneous | | User Interfaces | | |
| Documentation | | | Networking Devices | IPC | World Wide Web | | |
| File Handle Input/Output | | | Operating System Interfaces | | | | |

hosted by [perl.org](#), hardware provided by





in

[Adam John Trickett](#) > XML-RSS-Tools-0.16

XML-RSS-Tools-0.16

This Release	XML-RSS-Tools-0.16	[Download] [Browse]	15 Jan 2006
Other Releases	XML-RSS-Tools-0.15 -- 06 Feb 2005 <input type="button" value="Goto"/>		
Links	[CPAN Testers] [Discussion Forum] [View/Report Bugs (0)] [Tools]		
CPAN Testers	PASS (4) [View]		
Rating	☆☆☆☆☆ (0) [Rate this distribution]		
License	GPL		
Special Files	Build.PL MANIFEST Makefile.PL CHANGES META.yml README		

Modules

XML::RSS::Tools	A tool-kit providing a wrapper around a HTTP client, a RSS parser, and a XSLT engine.	0.16
---------------------------------	---	------

Documentation

Article	Processing RSS Files with XSLT
Example-1	Very simple example
Example-2	A Longer example
Example-3	Another simple example
Example-4	Using a single XSLT stylesheet
Example-5	A complete web server based RSS-client
Introduction	A Simple Introduction To RSS

hosted by [perl.org](#), hardware provided by

