

PERL

Frank Cui, Brisha Howe, Nidaa Mungloo

History

- Practical Extraction and Report Language
- Larry Wall 1986
- Perl has struggled since 2000
- Open Source 1989, Perl3
- Linux and Unix integration
- Linguist- Designed for writability & quick development
- “There’s more than one way to do it.”
 - qq(This is a test) is the same as “This is a test”



Uses

- Biostats
- Writing Emails
- Unified performance over several SQL platforms
- DBM Databases for mapping
- TAP (Test Anything Protocol) for universal, language-independent testing
- Remote Login Commands
- Widespread implementation



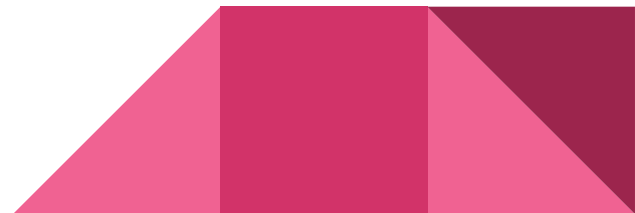
Design Goals

- High-level, general-purpose, interpreted, dynamic programming language
- Designed so programmers could code more quickly and easily, trade greater CPU and memory requirements for writability
- Perl syntax reflects the idea “Things that are different should look different”.

Scalars (variable), use \$ ex. \$numClass

Arrays(collection of elements), use @ ex. @classList

- Does not enforce any programming paradigm



Language Features

- Overall structure of perl derives broadly from C, also take features from shell programming ex. All variables marked with leading signals (\$,% etc)
- Unlike shell, perl also use signals to access variables.
- Perl takes lists from Lisp, hashes from AWK, and regular expressions from sed(stream editor), for simple parsing and text-handling tasks
- Automatic data-typing and memory management, legal type conversion are done at run time. Ex conversion from number to string




Language Features

- Dynamic binding
- Dynamic Scoping
- Reference point garbage collection



Writing Perl Code

- May be written in text editor or executed through shell script
 - Whitespace is ignored, so formatting is for readability
 - # for comments
 - Statements end with ;
 - Arrow notation (->) is used to call methods
 - {} used to group statements
 - Scalar variables start with \$
 - Arrays start with @
 - Hashes start with %
- 

Sample Code - Sending an email

```
#!/usr/bin/perl
use MIME::Lite;

$to = 'abcd@gmail.com';
$cc = 'efgh@mail.com';
$from = 'webmaster@yourdomain.com';
$subject = 'Test Email';
$message = '<h1>This is test email sent by Perl Script</h1>';

$msg = MIME::Lite->new(
    From      => $from,
    To        => $to,
    Cc        => $cc,
    Subject   => $subject,
    Data      => $message
);

$msg->attr("content-type" => "text/html");
$msg->send;
print "Email Sent Successfully\n";
```


Sample Code - Pattern Matching

```
#!/usr/bin/perl
use strict;
use warnings;

# SOME VARIABLES
$mystring = "Hello world!";

#Does the string contains the word "World"?
#This prints "No" because it's case sensitive
if($mystring =~ m/World/) {
    print "Yes"; } else {
    print "No";
}

z#This prints "Yes" because it's case insensitive
if($mystring =~ m/World/i) {
    print "Yes"; } else {
    print "No";
}
```

Sample Code - Comparing values

```
#!/usr/bin/perl

# HTTP HEADER
print "content-type: text/html \n\n";

# SOME VARIABLES
$name = "Sarah";
$x = 5;

# IF/ELSE STATEMENTS
if ($x > 10) {
    print "$x is greater than 10!";
} else {
    print "$x is not greater than 10!";
}
print "<br />";

# STRINGS ARE A LITTLE DIFFERENT
if ($name eq "Sarah") {
    print "Hello, $name!";
} else {
    print "You are not $name!";
}
]
```



Application

- The BioPerl Project - open source tools
- Mason: Amazon.com - generates HTML and dynamic content
- Booking.com - serialization libraries
- Google & Yahoo! - queries



Sources

- <https://stuff.mit.edu/iap/perl/slides/history.html>
- <https://learn.perl.org/>
- <http://perldoc.perl.org/index.html>
- <http://perl-begin.org/uses/>
- <https://softwareengineering.stackexchange.com/questions/115851/is-perl-still-a-useful-viable-language>
- <http://perl-begin.org/uses/qa/>
- http://www.softpanorama.org/Scripting/Perlbook/Ch01/perl_history.shtml
- <http://blog.builtonperl.com/post/5-major-websites-that-use-perl>
- <http://www.masonhq.com/?MasonPoweredSites>

The background is a solid dark blue color. In the top right corner, there is a decorative graphic consisting of several overlapping triangles in different shades of blue, including a medium blue and a very dark blue. The text "THANK YOU" is centered in the middle of the page in a white, bold, sans-serif font.

THANK YOU