

Angewandte Programmierung in der Computerlinguistik - Packages, Module und CPAN

Andreas Neumann M.A.

Themen

- ✳ Packages
- ✳ Einfache Perl-Module
- ✳ CPAN

Ein perl-Source-File in ein anderes integrieren

- * require
- * use

require

- ✳ Lädt Module bei erreichen des Statements ins Programm
- ✳ Passiert erst zu Laufzeit

use

- * Bindet Modul ein
- * Wird bevor das Programm ausgeführt wird eingebunden

```
#!/usr/bin/perl -w
use foo;
# entspricht
require foo;
foo->import();
```

```
#!/usr/bin/perl -w

require "insult.pl"
require insult;
use insult;

use MyModules::insult;
# Modul insult im Ordner/Modulpaket
MyModules/insult
```

Namespaces

- * Alle Variablen, Funktionen und Konstanten in Perl befinden sich in einem Namespace
- * Wird nichts angegeben befinden sie sich im Namespace **main**

Warum Namespaces – Zwei getrennte Dateien

```
#!/usr/bin/perl -w

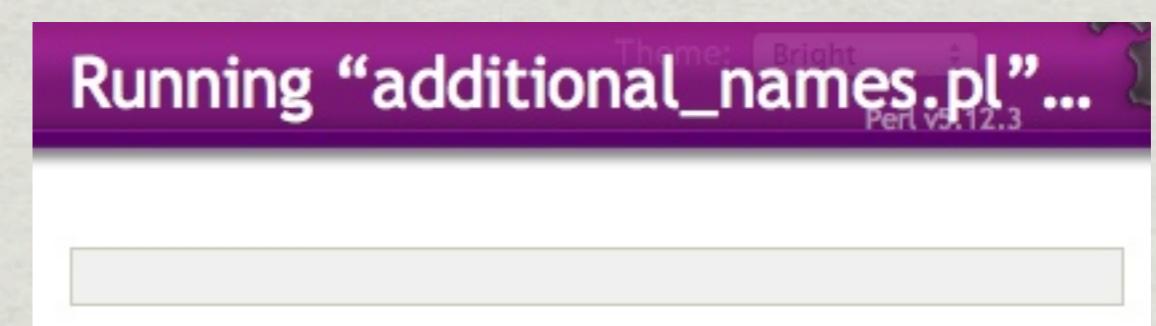
$name1 = "Andi";
$name2 = "Max";
$name3 = "Markus";

$names = [$name1, $name2, $name3];

print join("\n", @$names);
```

```
#!/usr/bin/perl -w

$name1 = "Chopy";
$name2 = "Max";
$name3 = "Momo";
```



Namespaces - Shadowing

- * Variablen und Funktionsnamen „überlagern“ einander
- * Letzte Definition sticht

```
#!/usr/bin/perl -w

$name1 = "Andi";
$name2 = "Max";
$name3 = "Markus";

require "additional_names.pl";

$names = [$name1, $name2, $name3];

print join("\n", @$names);
```

Running “nameclash.pl”... Theme: Light Perl v5.12.3

```
Chopy
Max
Momo
```

Namespaces definieren

- * **package** Paketname leitet Namespace ein
- * Bis zur nächsten package Deklaration sind alle folgenden Funktionen im package

```
#!/usr/bin/perl
package AdditionalNames;
$name1 = "Chopy";
$name2 = "Max";
$name3 = "Momo";

sub say_hello {
    print("Hello");
}
```

Auf Namespaces zugreifen

```
#!/usr/bin/perl

$name1 = "Andi";
$name2 = "Max";
$name3 = "Markus";

require "additional_names_done_right.pl";
$names = [$name1, $name2, $name3];
print join("\n", @$names);

# Auf Variable aus anderem Namespace zugreifen
print "\n".\$AdditionalNames::name1."\n";

# Auf eine Methode/Funktion im Namespace zugreifen
AdditionalNames::say_hello();
AdditionalNames -> say_hello();
```

- ✳ Zugriff über :: - Operator
- ✳ Alternative Syntax über -> - Notation

Module

- ✳ Module verwenden
- ✳ Eigene Module

Module verwenden

- * @INC
- * Umgebungsvariablen

Pfade

- * Um Module zu verwenden müssen sie im Pfad liegen
- * d.h. der Perl-Interpreter muss wissen in welchen Ordnern er nach den Modulen suchen soll
- * **@INC** enthält die Pfade in denen laut Perl-Konfiguration die Module liegen

@INC

```
#!/usr/bin/perl -w
# Im Array INC werden alle
Pfade angegeben in denen nach
Modulen gesucht wird
foreach (@INC) {
    print "$_\n"
}

use utf8;
```

Running “paths.pl”...

Theme: Bright
Perl v5.12.3

```
/Users/andi/Library/Application Support/TextMate/Managed/Bundles/Perl.tmbundle/Support
/Library/Perl/5.12/darwin-thread-multi-2level
/Library/Perl/5.12
/Network/Library/Perl/5.12/darwin-thread-multi-2level
/Network/Library/Perl/5.12
/Library/Perl/Updates/5.12.3
/System/Library/Perl/5.12/darwin-thread-multi-2level
/System/Library/Perl/5.12
/System/Library/Perl/Extras/5.12/darwin-thread-multi-2level
/System/Library/Perl/Extras/5.12
.
```

Pfade zur Programmausführung hinzufügen

- * Switch **-I** Pfad1:Pfad2:Pfad3
- * **use lib “foo/bar”**
- * Umgebungsvariable **PERL5LIB**
- * push in **@INC**

@INC erweitern

```
#!/usr/bin/perl -w
# Pfade hinzufügen

require 'additional_names.pl'; #Führt zum Abbruch

push(@INC,"../namespaces");

foreach (@INC) {print("$_\n")}

require 'additional_names.pl'; #Funktioniert an dieser Stelle
weil nun im Pfad
```

```
/Network/Library/Perl/5.12
/Library/Perl/Updates/5.12.3
/System/Library/Perl/5.12/darwin-thread-multi-2level
/System/Library/Perl/5.12
/System/Library/Perl/Extras/5.12/darwin-thread-multi-2level
/System/Library/Perl/Extras/5.12
.
../namespaces
```



Selektiver Import von Modulteilen

- * **use Module qw(\$scalar sub @list)**
- * **:DEFAULT** importiert den vom Modul definierten Standardsatz
- * Versionsbeschränkung mit *use module 1.5 qw ()*

Ein einfaches Modul schreiben

- * package
- * Exporter
- * @ISA
- * @EXPORT

package

- * Ein module benutzt namespaces um den Standardnamensraum nicht zu verseuchen

Exporter

- * Vereinfacht Interface / Schnittstelle für Module bereitzustellen
- * Einbinden: **use Exporter;**
- * **@EXPORT** definiert die Funktionen die beim einbinden mit use zur Verfügung gestellt werden

```
#!/usr/bin/perl -w
package MModule;

use Exporter;

@EXPORT = qw (@foo &bar)
```

Insult.pm - einfaches Beispiel für ein Modul

```
#!/usr/bin/perl -w

package insult;
use Exporter;
@ISA = ("Exporter");
@EXPORT = ("cow", "cow_reposte");

sub cow {
    print "You fight like a dairy farmer.\n";
}

sub cow_reposte {
    print "How appropriate. You fight like a cow.\n";
}

sub hello {
    print("I'm selling these coll leather-jackets!")
}

1;
#Obligatorisch ein wahrer Wert zum Schluss des Moduls
```

Verwendung insult.pm

```
#!/usr/bin/perl -w
use insult; # Funktioniert weil "." teil von @INC

cow();
cow_reposte();

# nicht exportierte Methoden nicht zugreifbar
# hello();
```

```
dhcp27:myModule andi$ perl use_insult_module.pl
You fight like a dairy farmer.
How appropriate. You fight like a cow.
```

CPAN - Comprehensive Perl Archive Network

- ✳ Archiv von perl Modulen
- ✳ ~19.000 Anwendungen
- ✳ Mehrfach gespiegelt



Comprehensive Perl Archive Network

OVER 15 YEARS OF PERL LIBRARIES AT YOUR FINGERTIPS

[Home](#) [Modules](#) [Ports](#) [Perl Source](#) [FAQ](#) [Mirrors](#)

Search:

Welcome to CPAN

The Comprehensive Perl Archive Network (CPAN) currently has [106,815 Perl modules](#) in 24,877 distributions, written by 9,699 authors, [mirrored](#) on 276 servers.

The archive has been online since October 1995 and is constantly growing.

Search CPAN via

- [metacpan.org](#)
- [search.cpan.org](#)

Recent Uploads

- [Bot-Cobalt-0.003](#)
- [Cinnamon-0.02](#)
- [Wx-0.9908](#)
- [Bot-Cobalt-0.002](#)
- [Socket-2.001_002](#)
- [Perl-Dist-Strawberry-3.003](#)
- [Geo-Coder-Google-0.10_01](#)
- [Dezi-UI-0.001001](#)
- [DR-Tarantool-0.04](#)
- [Task-BeLike-AMD-0.003](#)
- [more...](#)

Getting Started

- [Installing Perl Modules](#)
- [Learn Perl](#)

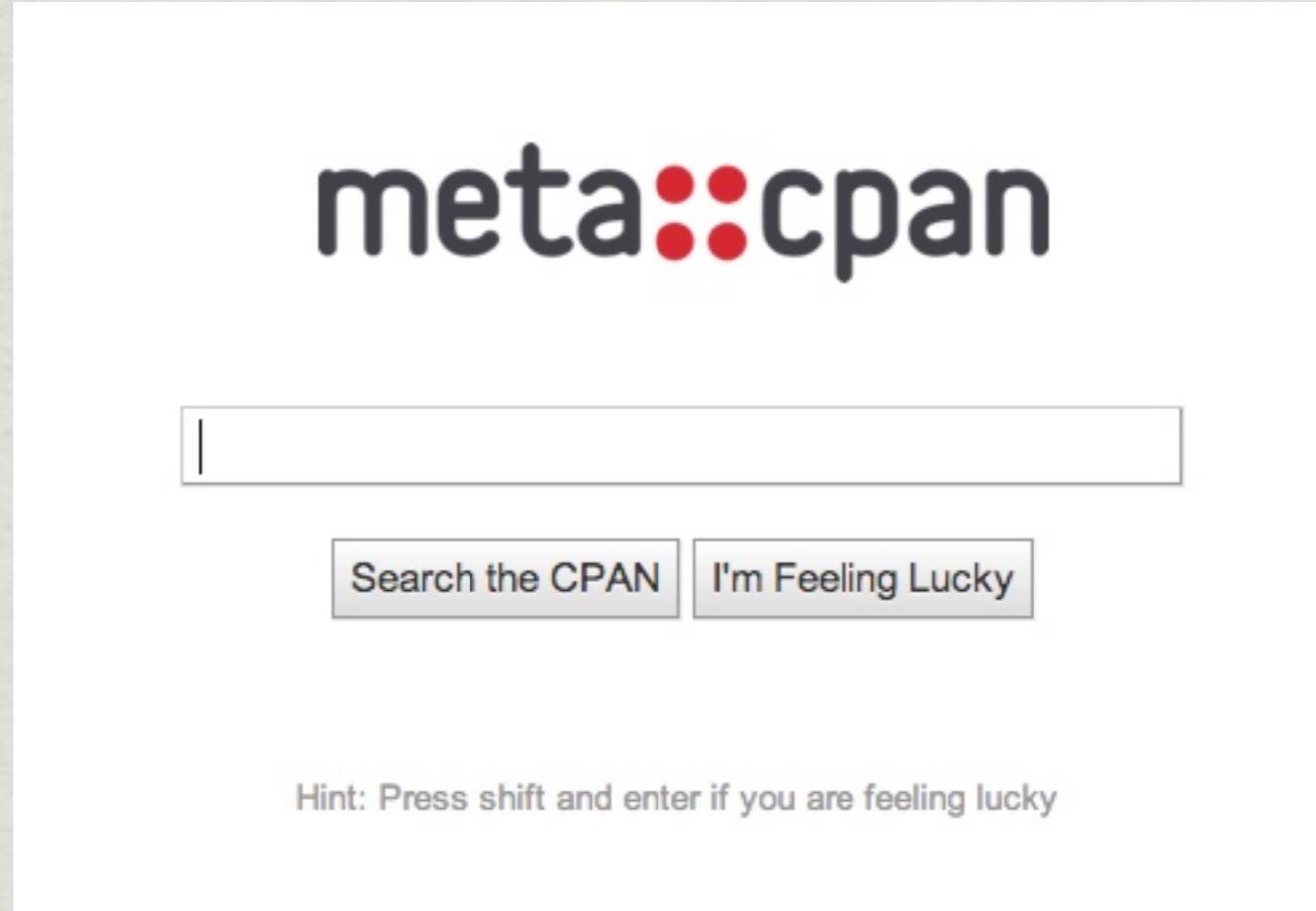
Perl Resources

- [The Perl Programming language](#)
- [Perl Documentation](#)
- [Mailing Lists](#)
- [Perl FAQ](#)
- [Scripts Repository](#)

Yours Eclectically, The Self-Appointed Master Librarians (OOK!) of the CPAN.
© 1995-2010 Jarkko Hietaniemi. © 2011 [Perl.org](#). All rights reserved. [Disclaimer](#).

Master mirror hosted by YellowBot®

* <http://www.cpan.org/>



* <https://metacpan.org/>

CPAN - shell

```
Terminal — perl5.12 — bash — 99x31 — %1

cpan[1]> h

Display Information (ver 1.9456)
command argument      description
a,b,d,m WORD or /REGEXP/ about authors, bundles, distributions, modules
i          WORD or /REGEXP/ about any of the above
ls         AUTHOR or GLOB  about files in the author's directory
           (with WORD being a module, bundle or author name or a distribution
            name of the form AUTHOR/DISTRIBUTION)

Download, Test, Make, Install...
get      download        clean   make clean
make     make (implies get)  look    open subshell in dist directory
test     make test (implies make) readme  display these README files
install  make install (implies test) perldoc display POD documentation

Upgrade
r        WORDs or /REGEXP/ or NONE report updates for some/matching/all modules
upgrade WORDs or /REGEXP/ or NONE upgrade some/matching/all modules

Pragmas
force   CMD   try hard to do command fforce CMD   try harder
notest  CMD   skip testing

Other
h,?      display this menu ! perl-code eval a perl command
o conf [opt] set and query options q       quit the cpan shell
reload cpan load CPAN.pm again reload index load newer indices
autobundle Snapshot recent   latest CPAN uploads

cpan[2]>
```

Exkurs: CPAN-Shell - Mein erstes mal

```
neumannb@edelreizker.cip.ifi.lmu.de:~ $ cpan  
Sorry, we have to rerun the configuration dialog for CPAN.pm due to  
some missing parameters. Configuration will be written to  
<</home/n/neumannb/.cpan/CPAN/MyConfig.pm>>
```

CPAN.pm requires configuration, but most of it can be done automatically.
If you answer 'no' below, you will enter an interactive dialog for each
configuration option instead.

Would you like to configure as much as possible automatically? [yes] █

Warning: You do not have write permission for Perl library directories.

To install modules, you need to configure a local Perl library directory or
escalate your privileges. CPAN can help you by bootstrapping the local::lib
module or by configuring itself to use 'sudo' (if available). You may also
resolve this problem manually if you need to customize your setup.

What approach do you want? (Choose 'local::lib', 'sudo' or 'manual')
[local::lib] █

Ein Modul suchen in der CPAN-Shell

```
cpan[8]> i /levenshtein/
Distribution    JGOLDBERG/Text-Levenshtein-0.05.tar.gz
Distribution    JGOLDBERG/Text-LevenshteinXS-0.03.tar.gz
Distribution    MATEU/Lingua-EN-SimilarNames-Levenshtein-0.10.tar.gz
Distribution    UGEXE/Text-Levenshtein-Damerau-0.02.zip
Module < Lingua::EN::SimilarNames::Levenshtein (MATEU/Lingua-EN-SimilarNames-Levenshtein-0.10.tar.gz)
Module < Text::Levenshtein      (JGOLDBERG/Text-Levenshtein-0.05.tar.gz)
Module < Text::Levenshtein::Damerau (UGEXE/Text-Levenshtein-Damerau-0.02.zip)
Module < Text::LevenshteinXS     (JGOLDBERG/Text-LevenshteinXS-0.03.tar.gz)
8 items found
```

Ein Modul suchen

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

mechanize
in

Results 1 - 10 of 103 Found
[1](#) · [2](#) · [3](#) · [4](#) · [5](#) · [6](#) · [Next >>](#)

[WWW::Mechanize::Boilerplate](#)
Compose Mechanize macros from specifications
[WWW-Mechanize-Boilerplate-0.03](#) - 24 Apr 2012 - Peter Sergeant

[WWW::Mechanize::Sleepy](#)
A Sleepy Mechanize Agent
[WWW-Mechanize-Sleepy-0.7](#) - 29 Sep 2010 - Kostas Ntonas

[WWW::Mechanize::Timed](#)
Time Mechanize requests
[WWW-Mechanize-Timed-0.44](#) ★★★☆☆ (1 Reviews) - 29 Apr 2008 - Léon Brocard

[Test::WWW::Mechanize::Driver](#)
Drive Test::WWW::Mechanize Object Using YAML Configuration Files
[Test-WWW-Mechanize-Driver-0.6](#) - 01 Jul 2009 - Dean Serenevy

[Test::WWW::Mechanize::Driver::Manual](#)
Writing Tests for Test::WWW::Mechanize::Driver
[Test-WWW-Mechanize-Driver-0.6](#) - 01 Jul 2009 - Dean Serenevy

[WWW::Mechanize](#)
Handy web browsing in a Perl object
[WWW-Mechanize-1.72](#) ★★★★★ (18 Reviews) - 02 Feb 2012 - Jesse Vincent

[WWW::Mechanize::FAQ](#)
Frequently Asked Questions about WWW::Mechanize
[WWW-Mechanize-1.72](#) ★★★★★ (18 Reviews) - 02 Feb 2012 - Jesse Vincent

Ein Modul suchen

meta::cpan levenshtein 24 results (0.238 seconds)

[Text::Levenshtein - An implementation of the Levenshtein edit distance](#) ++

This module implements the Levenshtein edit distance. The Levenshtein edit distance is a measure of the degree of proximity between two strings. This distance is the number of substitutions, deletions or insertions ("edits") needed to transform one s...

JGOLDBERG/Text-Levenshtein-0.05 ★★★★☆ (3 reviews) - Jul 01, 2004 - [Search in distribution](#)

[Lingua::EN::SimilarNames::Levenshtein - Compare people first and last names.](#) ++

MATEU/Lingua-EN-SimilarNames-Levenshtein-0.10 - Aug 24, 2011 - [Search in distribution](#)

[Text::Levenshtein::Damerau - Damerau Levenshtein edit distance](#) ++

Returns the true Damerau Levenshtein edit distance of strings. AUTHOR Copyright 2011 Nick Logan <nickl@skunkds.com> This package is free software and is provided "as is" without express implied warranty. You can redistribute it and/or modify it un...

UGEXE/Text-Levenshtein-Damerau-0.02 - Jan 02, 2012 - [Search in distribution](#)

[Text::LevenshteinXS - An XS implementation of the Levenshtein edit distance](#) ++

This module implements the Levenshtein edit distance in a XS way. The Levenshtein edit distance is a measure of the degree of proximity between two strings. This distance is the number of substitutions, deletions or insertions ("edits") needed to tra...

JGOLDBERG/Text-LevenshteinXS-0.03 ★★★★☆ (2 reviews) - Jul 01, 2004 - [Search in distribution](#)

[Text::Brew - An implementation of the Brew edit distance](#) ++

This module implements the Brew edit distance that is very close to the dynamic programming technique used for the Wagner-Fischer (and so for the Levenshtein) edit distance. Please look at the module references below. For more information about the B...

KCIVEY/Text-Brew-0.02 - Apr 02, 2005 - [Search in distribution](#)

[Text::EditTranscript - Perl extension for determining the edit transcript between two strings](#) ++

The edit transcript is a sequence of operations to transform one string into another string. The operations include 'Insertion', 'Deletion', and Substitution. This module creates a string denoting the l operations to transfer the second string ...

LMETCALF/Text-EditTranscript-0.07 - Oct 16, 2009 - [Search in distribution](#)

[Text::JaroWinkler - An implementation of the Jaro-Winkler distance](#) ++

Ein Modul installieren

```
cpan[2]> install Cards::Poker
```

```
Warning: Cannot install Cards::Poker, don't know what it is.
```

```
Try the command
```

```
i /Cards::Poker/
```

```
to find objects with matching identifiers.
```

```
cpan[3]> i /Cards::Poker/
```

```
Module id = Games::Cards::Poker
```

```
DESCRIPTION Pure Perl Poker Functions
```

```
CPAN_USERID PIP (Pip Stuart <Pip@CPAN.0rg>)
```

```
CPAN_VERSION 1.002565
```

```
CPAN_FILE P/PI/PIP/Games-Cards-Poker-1.2.565CHh5.tgz
```

```
DSLIP_STATUS Rdpfg (released,developer,perl,functions,GPL)
```

```
INST_FILE (not installed)
```

```
cpan[4]> install Games::Cards::Poker
```

Vorteile von CPAN

- * Dependencies auflösen

```
----- Unsatisfied dependencies detected during -----
----- PIP/Games-Cards-Poker-1.2.565CHh5.tgz -----
    Math::BaseCnv [requires]
    Algorithm::ChooseSubsets [requires]
Shall I follow them and prepend them to the queue
of modules we are processing right now? [yes] ■
```

```
Building Module-Build
LEONT/Module-Build-0.40.tar.gz
./Build -- OK
Running Build test
t/00-compile.t ..... ok
t/PL_files.t ..... ok
t/actions/installdeps.t ..... ok
t/actions/manifest_skip.t ..... ok
t/add_property.t ..... ok
t/basic.t ..... ok
t/bundle_inc.t ..... skipped: $E
t/compat.t ..... 26/? ■
```

Vorteile CPAN

- ⌘ Online-Dokumentation zu den Modulen

[Pip Stuart > Games-Cards-Poker-1.2.565CHh5 > Games::Cards::Poker](#)

Module Version: 1.2.565CHh5 [Source](#)

NAME

VERSION

SYNOPSIS

DESCRIPTION

2DO

USAGE

Deck()

Shuffle(@cards)

SortCards(@cards)

ShortHand(@hand)

ScoreHand(@hand)

HandScore(\$score)

ADVANCED USAGE

SlowScoreHand(@hand)

UseSlow([\$slow])

BestIndices(@cards)

BestHand(@best, @cards)

HandName(\$score)

VerboseHandName(\$score)

NameCard(%names)

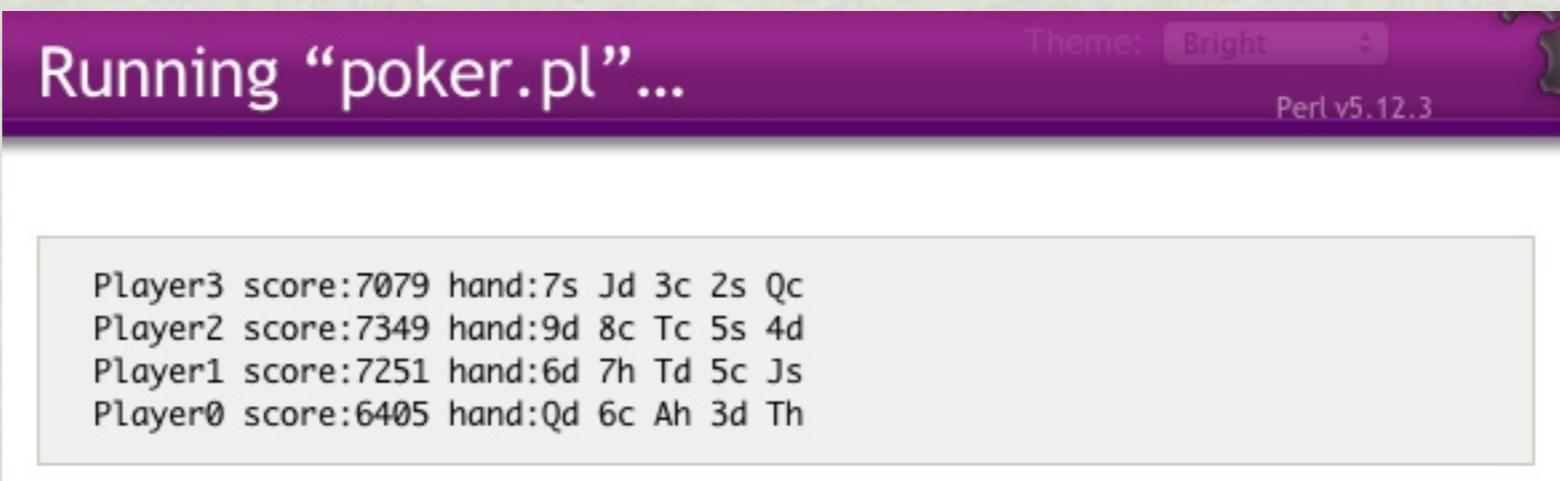
Das über CPAN installierte Modul verwenden

```
#!/usr/bin/perl
# Source: http://search.cpan.org/~pip/Games-Cards-Poker-1.2.565CHh5/Poker.pm

use Games::Cards::Poker;

# Deal Four (4) players hands and score them...
my $players = 4; # number of players to get hands dealt
my $hand_size = 5; # number of cards to deal to each player
my @hands = (); # player hand data
my @deck = Shuffle(Deck());

while($players--) {
    push(@{$hands[$players]}, pop(@deck)) foreach(1..$hand_size);
    printf("Player$players score:%4d hand:@{$hands[$players]}\n",
          ScoreHand(@{$hands[$players]}));
}
}
```



The screenshot shows a terminal window with a purple header bar. The title bar says "Running ‘poker.pl’...". In the top right corner, there are "Theme: Bright" and "Perl v5.12.3" buttons. The main window displays the output of the Perl script, which lists four poker hands with their scores:

```
Player3 score:7079 hand:7s Jd 3c 2s Qc
Player2 score:7349 hand:9d 8c Tc 5s 4d
Player1 score:7251 hand:6d 7h Td 5c Js
Player0 score:6405 hand:Qd 6c Ah 3d Th
```