

# Kurs: Perl-Aufbau

© T. Birnthaler, OSTC GmbH ([www.ostc.de](http://www.ostc.de))

(Version 1.1 vom 3.3.2005)

**Dauer:** 3-4 Tage

**Motivation:** Perl ist eine extrem portable und leistungsfähige Skript-Sprache, die weltweit von sehr vielen Entwicklern für allgemeine Programmentwicklung, für Web-Applikationen und in der Systemadministration eingesetzt wird. Man könnte Perl auch als "Linux/UNIX in a Box" bezeichnen, d.h. man erhält im Rahmen einer Programmiersprache die Vielzahl an Möglichkeiten und Hilfsmitteln des Linux/UNIX-Betriebssystems zur Verfügung gestellt. D.h. bei Linux/UNIX-Vorkenntnissen werden Ihnen viele Sprachelemente und -eigenschaften von Perl bekannt vorkommen, umgekehrt vereinfacht die Kenntnis der Programmiersprache Perl den Umgang mit dem Linux/UNIX-Betriebssystem.

**Ziel:** Mittels vieler praktischer Übungen lernen Sie die Elemente von Perl gründlich kennen und erstellen einfache sowie auch komplexere Programme (Skripte), die gründlich ausgetestet werden. Der Kurs legt viel Wert auf sorgfältig und klar geschriebene Programme und gibt auch Tipps, um trotz der (zunächst) etwas kryptisch erscheinenden Perl-Syntax übersichtlichen Programmcode zu formulieren.

**Hinweis:** Der Kurs ist für Linux/UNIX- und für MS-Windows-Systeme geeignet, als Programmierumgebung wird die Kommandozeile und ein beliebiger Editor verwendet.

## **Inhalt:**

- Weitere Kontrollstrukturen (unless, until, do, elsif, for, last, redo, next)
- Dateien und Verzeichnisse verwalten (Globbing, Links, File-Modul)
- Formate (Report-Generator)
- Prozessverwaltung (system, Exit-Status, exec, Umgebungs-Variablen, Kommando-Substitution, fork, waitpid, Signale)
- Stringfunktionen (index, rindex, substr, sprintf)

- Fortgeschrittenes Sortieren
- Einfache Datenbanken (DBM-Hash, pack, unpack, Inplace-Edit)
- Fortgeschrittene Techniken (eval, grep, map, Slices)
- Erweiterte Reguläre Ausdrücke (non-greedy, multiline, weitere Anker, once, Kommentare, ...)
- Module und Pakete (Namensräume)
  - Konzept der Modularisierung in Perl
  - Exportbeschränkung
  - Sichtbarkeit von Funktionen und Variablen
  - Schreiben eigener Module
  - Beispiele für Module
    - \* Standardmodule (File, ...)
    - \* Weitere Module (z.B. CGI, DBI)
    - \* Win32-Modul

#### **Schulungsunterlagen:**

- Skript
- Lösungsblätter zu allen Übungen

#### **Voraussetzungen:**

- Muss: Allgemeine Computer-Kenntnisse (Tastatur, Maus, Grafische Oberfläche, Editor)
- Muss: Kenntnis einer beliebigen Programmiersprache
- Muss: Kenntnis des Kurses "Perl-Einführung"
- Optional: Kenntnis einer Integrierten Entwicklungsumgebung (IDE)
- Optional: Linux/UNIX-Kenntnisse
- Optional: Shell-Programmierung-Kenntnisse
- Optional: HTML-Kenntnisse

#### **Abgrenzung:**

- HTML und CGI-Programmierung wird nur am Rande behandelt (extra Kurs)