

## Fundamentals of Perl

**Level:** foundation

**Length:** 35 – 40 ore

**Course Objective:** introduction to Perl programming language, where and how it is used

### What you will learn

- How the programs are organized, software development cycle
- Code elements: data structures, control structures, fundamental types
- How Perl is used especially for information extraction and data reporting
- Exercise soft skills like communication, team work, presentation of ideas and solutions

**Who Can Attend:** anybody who wants to learn Perl programming language

### Prerequisites

- Basic notions of programming
- Comfortable with using the host operating system
- The knowledge of another programming language, for example C, Python, C++, Java, etc.) would be helpful to easier learn Pearl

**Required infrastructure:** VGA projector, whiteboard, workstation

### Bibliography:

- Programming Perl, Fourth Edition, Tom Christiansen & others, O'Reilly 2012, ISBN: 978-0-596-00492-7
- Learning Perl, Sixth Edition, Randal L. Schwartz & all, O'Reilly 2011, ISBN: 978-1-449-30358-7

**Related Courses:** Advanced Perl Subjects

### Description

The course is addressed to programmers who want a quick and solid introduction to Perl. The focus is the language, its constructive elements – data structures, control structures, how the programs are organized, particularities of using it in procedural programming, and a little bit of object oriented programming.

Perl is exercised by examples and practical assignments, it is shown how the problems are approached and solved, particularities and stereotypes of using this language.

## **Contents**

1. Introduction to Perl programming language
2. Installation
3. Running standalone Perl programmers
4. Basic language elements
5. Perl documentation
6. UNIX and Windows differences regarding Perl
7. Using of an Integrated Development Environment (IDE)
8. CGI – Common Gateway Interface
9. Scalar variables and operators
10. Arrays
11. Control structures
12. Code organization – functions (subroutines)
13. Working with the file system
14. Regular expression
15. References
16. Subroutines – more details
17. Complex data structures
18. Typeglob and NULL (undef)
19. Constants and special variables
20. Perl functions
21. Launching processes
22. Packages and modules
23. Object Oriented Programming