Unix/Linux Shell Scripting Training

Duration: **2 Day**

BILGINC IT Academy
info@bilginc.com
bilginc.com

Overview

This {*training} introduces the Unix operating system and enables participants to write efficient shell scripts. Throughout the course, delegates will build scripts of increasing complexity and gain hands-on experience with console I/O, text processing, arithmetic operations, process control, and signal handling.

Prerequisites

No prior experience is required.

Who Should Attend

- UNIX users
- Programmers
- System administrators

What You Will Learn

The course is primarily delivered using the Korn shell (ksh), while also covering Bash, Bourne, and C shells. Training is typically conducted on Red Hat or Ubuntu Linux systems.

Outline

Review of Core Concepts

- The history and evolution of UNIX
- The filesystem, mounting devices and inodes
- Processes, threads, scheduling and signals
- Using the essential UNIX commands
- Comparing the different UNIX shells
- Picking the right shell for the job

Starting Shell Programming

- Why shell scripts are an essential tool
- Using the shebang line to pick an interpreter
- Declaring and working with shell variables
- Built in operators for testing and arithmetic
- Evaluating expressions using the expr command
- Testing variables with if and case

BILGINC IT Academy

info@bilginc.com

bilginc.com

- Creating and controlling for, while and until loops
- Debugging your shell scripts

Regular Expressions

- How a Regular Expression Engine operates
- The difference between basic and extended regular expressions
- Options for using regular expressions in shell scripts
- Creating character classes and specifying multiplicities
- Meta-characters for specifying positions in the input
- Using parenthesis for grouping and submatches
- The non greedy versions of *, + and ?
- Using parenthesis that do not capture
- Applying modifiers to only part of the expression
- Using look-around assertions to match without capturing

Creating Useful Scripts

- Creating formatted output with printf
- Working with environment variables
- Testing the exit status of a command
- Looping till a command is successful
- Discarding command output using /dev/null
- Using /dev/zero to create test files
- Testing the attributes of files
- Returning an exit status from your script
- Processing command line arguments
- Reading and parsing input from the keyboard
- Processing files one line at a time
- Creating and working with arrays

Using the Sed Tool in Shell Scripts

- Understanding how Sed transforms text input
- Working with commands and addresses
- Manipulating the contents of the pattern space
- Using the hold buffer to copy and paste text
- Sed commands for selection and iteration

Using the Awk Tool in Shell Scripts

- Understanding how Awk processes text
- Breaking up input into records and fields
- Writing actions to processes records
- Using regular expressions in actions
- Awk commands for selection and iteration
- Adding functions to Awk scripts

Advanced Shell Programming

- Trapping and handling signals
- Running other programs via exec

Unix/Linux Shell Scripting Training

Duration: **2 Day**

BILGINC IT Academy
info@bilginc.com
bilginc.com

- Starting, monitoring and deleting processes
- Breaking up scripts into functions
- Security issues with shell scripts

Alternatives to Shell Scripting

- How Perl improves and extends shell scripting
- Object oriented scripting using Ruby and Python