CU110 - Fundamentals of UNIX

Course Description:

This course provides a comprehensive introduction to the full range of UNIX user commands and utilities. Students will develop shell programming and vi editing skills.

Audience:

End-users and programmers who are new to the UNIX environment.

Prerequisites:

None

Course Contents

Introduction

- Course objectives
- Course overview
- Suggested references and readings

Getting Started

- What is UNIX?
- A brief history of UNIX
- Logging in
- Logging out
- Try a few more commands
- Changing your password
- Using on-line manuals

P.O. Box 307218
Columbus, Ohio 43230
+1 (866) 521-1776
http://www.corder.com

SAGE certification PREFERRED TRAINING PROVIDER

The File System - Files

- What is a file?
- The ls command
- The cat command
- The more and pg commands
- The head and tail commands
- The cp command
- The mv command
- The rm command
- File names

The File System - Directories

- Hierarchical file system
- Pathnames
- The pwd command - print working directory
- The cd command - change directory
- The mkdir command - make directories
- The rmdir command - remove directories
- The cp command - copy files
- Two useful directory names - . and ..

Editing With vi

- What is vi?
- The vi buffering process
- Command mode and insert mode
- Modes diagram
- Getting started
- Moving the cursor around

Inserting text

- Deleting a character or line
- Undo last command
- Opening a new line
- Save your work or abort the session
- Review of vi commands

More Editing With vi

- Scrolling the buffer
- Cursor motion commands - w, W, b, B, e, E
- Cursor motion commands - $, ^, 0, G
- Cursor motion commands - f, t, F, T
- Delete operator - d
- Change operator - c
- Yank operator - y
- Put commands - p, P
- Searching for a pattern - /, n, N, ?
- The join command
- The fi le command - :f
- Edit fi le command - :e
- Cut and paste between fi les
- Read fi le command - :r
- Set options command
- Set options command - .exrc fi le

Personal Utilities

- The date utility
- The bc utility
- The expr utility
- The cal utility
- The news utility
- The id utility
- The uname utility
- The fi nger utility
- The script utility
- The clear utility
- Appendix: The at and crontab utilities

Text Handling Utilities

- The grep utility
- The tr utility
- The cut utility
- The paste utility
- The sort utility
- The wc utility
- The diff utility
- The lp utility

File System Security

File permissions

- The chmod utility
- Directory permissions
- The umask command

File System Management Utilities

- The fi nd utility
- The df utility
- The du utility
- Compressing fi les
- The ln utility
- The ulimit utility
- The tar utility

Communication Utilities

- The write and talk utilities
- The mesg utility
- Mail overview
- The mail utility
- The mailx utility
- elm - electronic mail
- Sending mail with elm
- Reading mail with elm
- Customizing elm
- elmrc example

Using the Shell

- What is a shell?
- Which shell?
- The command line
- Standard input, standard output and standard error
- Using default standard in and standard output
- I/O redirection
- Appending output of a fi le
- Pipes
- The tee utility

Filename Generation

- Filename generation
- The ? special character
- The * special character
- The [ ] special characters
- The ! special characters

Introduction to Shell Programming

- Shell programming objectives
- Overview
- Suggested references and reading

UNIX Processes

- What is a process?
- Process structure
- The ps utility
- Options to the ps utility
- Background commands (&)
- Killing background processes
- Redirecting the standard error

Shell Programming Concepts

- What is a shell?
- What is a shell script?
- Why use shell scripts?

Flow Control.

- The exit status of commands
- Command line examples
- The test command
- The if-then-else construct
- The elif construct
- A loop example

Variables

- User created variables
- The shell environment
- The export command
- Sub-shells
- Command substitution
- Quoting mechanisms
- Assigning variables - summary

Special Variables

- Command line arguments
- $# - Number of arguments
- The shift command
- $* - All arguments
- $$ - PID of shell

More Flow Control

- The for loop
- The while loop
- The case construct

Appendix: Korn shell features

- Viewing your command history
- Editing and re-executing commands