

UNIX SHELL PROGRAMMING

TABLE OF CONTENTS

- **1 - INTRODUCTION:**
 - What you should already know... 1-2
 - What is the shell? 1-5
 - The role of the shell 1-6
 - Determining your shell type 1-7
 - What is a shell program? 1-8
 - Creating a shell script 1-10
 - Comments 1-13
 - The subshell type descriptor record 1-15
 - Exercises 1-17
 - Exercise solutions 1-18

- **2 - I/O AND VARIABLES:**
 - Using echo 2-2
 - Using print and printf 2-4
 - Variables: overview 2-6
 - Variable assignment 2-8
 - Variable substitution 2-9
 - Variable "dump": set 2-11
 - Variable assignment: read 2-12
 - Removing a variable: unset 2-14
 - Exporting variables: export 2-15
 - "Sourcing" a script with the dot (.) 2-18
 - The backslash as a quote: \ 2-19
 - Single quotes: ' ' 2-22
 - Double quotes: " " 2-24
 - Exercises 2-26
 - Exercise solutions 2-28

UNIX SHELL PROGRAMMING

TABLE OF CONTENTS (continued)

- **3 - CONDITIONAL BRANCHING:**
 - Overview 3-2
 - Return values 3-3
 - Basic decision making: if blocks 3-4
 - The opposite of if: else blocks 3-5
 - Contracted nesting: elif 3-8
 - The test command 3-11
 - The test operators 3-14
 - Short circuit operators 3-17
 - The exit command 3-19
 - Command substitution: `command` 3-20
 - Simple math: expr and \$(()) 3-21
 - Exercises 3-23
 - Exercise solutions 3-26

- **4 - WHILE LOOPS:**
 - Overview 4-2
 - The while loop 4-3
 - The until loop 4-7
 - Loop control: continue and break 4-9
 - Nested loops as branch points 4-12
 - Exercises 4-14
 - Exercise solutions 4-16

UNIX SHELL PROGRAMMING

TABLE OF CONTENTS (continued)

- **5 - FOR LOOPS AND POSITIONAL PARAMETERS:**

➤ Overview	5-2
➤ The for loop	5-3
➤ Positional parameters	5-5
➤ Accessing positional parameters: shift	5-6
➤ \$* versus \$@	5-8
➤ Loading up the positional parameters	5-15
➤ The for loop without a list	5-17
➤ Examples of for loops	5-18
➤ Exercises	5-20
➤ Exercise solutions	5-22

- **6 - DEBUGGING:**

➤ Overview	6-2
➤ Execution trace: set -x	6-3
➤ Verbose trace: set -v	6-8
➤ Combining -x and -v	6-10
➤ Error exit: set -e	6-12
➤ No execution: -n	6-14
➤ No unset: set -u	6-16
➤ Temporarily "commenting out" code	6-17
➤ Removing the # characters	6-19
➤ Temporary exits and breakpoints	6-20
➤ Bypasses	6-21
➤ Exercises	6-22
➤ Exercise solutions	6-24

UNIX SHELL PROGRAMMING

TABLE OF CONTENTS (continued)

- **7 - THE case AND select STATEMENTS:**
 - The case statement 7-2
 - The patterns 7-4
 - A case example 7-5
 - The select statement 7-7
 - A select example 7-9
 - LINES and COLUMNS 7-10
 - Exercises 7-12
 - Exercise solutions 7-13

- **8 - THE 'HERE' DOCUMENT:**
 - Overview 8-2
 - The 'here' document: << 8-3
 - Allowing indentation with tabs 8-5
 - Preventing substitutions 8-6
 - Self-extracting scripts 8-7
 - Exercises 8-8
 - Exercise solutions 8-10

- **9 - SIGNALS AND TRAPS:**
 - Keyboard "interrupts" 9-2
 - Signals, PID's and kill 9-4
 - Terminating a process: kill 9-6
 - Protecting your scripts: trap 9-11
 - Setting an ignore trap 9-12
 - Setting a catch trap 9-16
 - Setting a catch trap for the exit signal 9-19
 - A trap within a trap 9-20
 - Exercises 9-21
 - Exercise solutions 9-22

UNIX SHELL PROGRAMMING

TABLE OF CONTENTS (continued)

- **10 - FUNCTIONS:**
 - Overview 10-2
 - Creating a function 10-3
 - Function definition files 10-6
 - Removing functions 10-8
 - Autoloading functions 10-9
 - Variable scope in functions 10-12
 - Positional parameters and functions 10-14
 - Exercises 10-16
 - Exercise solutions 10-20

- **11 - INTRODUCTION TO sed:**
 - Automating changes: sed 11-2
 - How sed works 11-3
 - sed commands 11-4
 - sed command examples 11-5
 - sed example 11-6
 - sed example walk through 11-7
 - The -n option 11-10
 - Exercises 11-11
 - Exercise hints 11-12
 - Exercise solutions 11-14

- **APPENDIX: WHAT YOU SHOULD ALREADY KNOW:**
 - What you should know A-1
 - vi editor summary A-7
 - Regular expression metacharacter summary A-12
 - Finding patterns with grep A-13
 - Extracting fields with cut A-17

THIS PAGE IS INTENTIONALLY BLANK