1. What is the purpose of ‘exec’ command in TCL?

The exec command in TCL is used to execute external programs or system commands from within the TCL script. It allows you to run a command in the shell, capture its output, and even pass arguments to it. The command returns the output of the external program.

Example:

set output [exec ls -l]

puts "Command output: $output"

1. What is the ‘clock’ command used for in TCL?

The clock command in TCL is used to work with date and time. It can return the current time, format it, or perform time-based calculations.

Example:

# Get the current time in seconds since the epoch

set current\_time [clock seconds]

puts "Current time in seconds: $current\_time"

# Format the current time

set formatted\_time [clock format $current\_time -format "%Y-%m-%d %H:%M:%S"]

puts "Formatted current time: $formatted\_time"

1. How do you handle signals in TCL?

In TCL, signals (like interrupts or termination signals) are handled using the trap command. This command allows you to catch signals such as SIGINT or SIGTERM and specify what action to take.

Example:

trap {puts "Received SIGINT signal! Exiting."} SIGINT

# Simulate a long-running process

vwait forever

1. How can you execute TCL scripts in interactive mode?

To execute a TCL script in interactive mode, you can run the TCL interpreter and provide the script file as an argument or execute commands interactively by entering them at the TCL prompt.

Example:

tclsh myscript.tcl

Or

tclsh

1. What is the difference between ‘list’ and ‘array’ in TCL?

* List: A list in TCL is an ordered collection of elements. Lists are indexed by their position, starting from 0. Lists are simple and are used when the order of elements matters.

Example: set mylist {apple banana cherry}

* Array: An array in TCL is an associative collection of key-value pairs. The keys can be any valid string, and arrays are used when you need to access elements by a unique identifier (key) rather than by position.

Example: array set myarray {key1 value1 key2 value2}

1. What is the purpose of the ‘after’ command in TCL?

The after command in TCL is used to schedule the execution of a command after a specified amount of time (in milliseconds). It can be used for implementing delays or time-based operations.

Example:

# Run a command after 2000 milliseconds (2 seconds)

after 2000 {puts "This message appears after 2 seconds."}

1. Swap 2 numbers without using temporary variable in TCL.

set a 5

set b 10

# Swap the values

set a [expr {$a + $b}]

set b [expr {$a - $b}]

set a [expr {$a - $b}]

puts "After swapping: a = $a, b = $b"

1. Write a TCL code to multiply 2 numbers without using multiplication operator.

proc multiply {a b} {

set result 0

for {set i 0} {$i < $b} {incr i} {

set result [expr {$result + $a}]

}

return $result

}

set num1 5

set num2 3

set product [multiply $num1 $num2]

puts "Product of $num1 and $num2 is: $product"