TCL Scripting

Strings

Basem Atia

June-2025

TCL Strings

String first

- Command Description
  - string first string1 string2
  - Returns the index of the character in string1 that starts the first match to string2, or -1 if there is no match

- Example:

```
puts [string first "l" "hello"]
puts [string first "z" "hello"]
```

- Output:

```
2
-1
```

TCL Strings

String last

- Command Description
  - string last string1 string2
  - Returns the index of the character in string1 that starts the last match to string2, or -1 if there is no match

- Example:

```
puts [string last "l" "hello"]
puts [string last "z" "hello"]
```

- Output:

```
3
-1
```

TCL Strings

String equal

- Command Description
  - string equal string1 string2
  - Compares two strings for exact equality, case-sensitive. Returns 1 if equal, 0 if not.

- Example:

```
puts [string equal "hello" "hello"]
puts [string equal "hello" "Hello"]
```

- Output:

```
1
0
```

TCL Strings

String match

- Command Description
  - string match string1 string2
  - Returns the index of the character in string1 that starts the first match to string2, or -1 if there is no match

- Example:

```
puts [string match "he*" "hello"]
puts [string match "he?" "hello"]
```

- Output:

```
1
0
```

TCL Strings

String range

- Command Description
  - string range string firstIndex lastIndex
  - Extracts a substring from a string, between given start and end indices

- Example:

```
puts [string range "abcdef" 2 4]
```

- Output:

```
cde
```

TCL Strings

String index

- Command Description
  - string insert string index
  - Returns the **character** at a specific index in a string and is
  - zero based

- Example:

```
puts [string index "hello" 1]
```

- Output:

```
e
```

TCL Strings

String length

- Command Description
  - string length string
  - Returns the **number of characters** in a string
  - NOT zero based

- Example:

```
puts [string length "hello"]
```

- Output:

```
5
```

TCL Strings

join

- Command Description
  - Join string delimiter
  - Joins a list of strings into a **single string**, separated by a given delimiter.
  - Default delimiter is white space

- Example:

```
puts [join {apple banana cherry} ", "]
```

- Output:

```
apple, banana, cherry
```

TCL Strings

split

- Command Description
  - Split string delimiter
  - Splits a **string into a list** based on each **character** of a delimiter string
  - Default delimiter is whitespace

- Example:

```
set str "a,b,c"
set result [split $str ","]
puts $result
```

- Output:

```
a b c
```

TCL Strings

String append

- Command Description
  - append varName value
  - Appends one or more strings to the **end of a variable**'s current value.

- Example:

```
set a "Hello"
append a " World"
puts $a
```

- Output:

```
Hello World
```

TCL Strings

regexp

- Command Description
  - regexp ?switches? exp string
  - Tests if a string matches a regular expression pattern.
  - Returns 1 if there's a match, 0 otherwise.

- Example:

```
puts [regexp {h.llo} "hello"]
puts [regexp {\d+} "abc123"]
```

- Output:

```
1
1
```

TCL Strings

regsub

- Command Description
  - regsub ?switches? exp string subSpec ?varName?
  - **Replaces** parts of a string that match a regular expression with a replacement string

- Example:

```
set str "abc123"
regsub {\d+} $str "NUM" newstr
puts $newstr
```

- Output:

```
abcNUM
```

TCL Strings

Format

- Command Description
  - format formatString ?arg arg ...?
  - Formats strings in a C-style way (like printf),

- Example:

```
puts [format "Pi is approximately %.2f" 3.14159]
```

- Output:

```
Pi is approximately 3.14
```