

TCL - FILE I/O

http://www.tutorialspoint.com/tcl-tk/tcl_file_io.htm

Copyright © tutorialspoint.com

Tcl supports file handling with the help of the built in commands open, read, puts, gets and close.

A file represents a sequence of bytes, does not matter if it is a text file or binary file.

Opening Files

Tcl uses the open command to open files in Tcl. The syntax for opening a file is as follows.

```
open fileName accessMode
```

Here, **filename** is string literal, which you will use to name your file and **accessMode** can have one of the following values:

| Mode | Description |
|------|---|
| r | Opens an existing text file for reading purpose and the file must exist. This is the default mode used when no accessMode is specified. |
| w | Opens a text file for writing, if it does not exist then a new file is created else existing file is truncated. |
| a | Opens a text file for writing in appending mode and file must exist. Here your program will start appending content in the existing file content. |
| r+ | Opens a text file for reading and writing both. File must exist already. |
| w+ | Opens a text file for reading and writing both. It first truncate the file to zero length if it exists otherwise create the file if it does not exist. |
| a+ | Opens a text file for reading and writing both. It creates the file if it does not exist. The reading will start from the beginning but writing can only be appended. |

Closing a File

To close a file, use the close command. The syntax for close is as follows.

```
close fileName
```

Any file that has been opened by a program must be closed when the program finishes using that file. In most cases the files need not be closed explicitly; they are closed automatically when File objects are terminated automatically.

Writing a File

puts command is used to write to an open file.

```
puts $filename "text to write"
```

A simple example for writing to a file is shown below.

```
#!/usr/bin/tclsh
set fp [open "input.txt" w+]
puts $fp "test"
close $fp
```

When the above code is compiled and executed, it creates a new file **input.txt** in the directory that it has been started under *intheprogram's working directory*.

Reading a File

Following is the simple command to read from a file:

```
set file_data [read $fp]
```

A complete example of read and write is shown below.

```
#!/usr/bin/tclsh

set fp [open "input.txt" w+]
puts $fp "test"
close $fp
set fp [open "input.txt" r]
set file_data [read $fp]
puts $file_data
close $fp
```

When the above code is compiled and executed, it reads the file created in previous section and produces the following result:

```
test
```

Here is another example for reading file till end of file line by line.

```
#!/usr/bin/tclsh

set fp [open "input.txt" w+]
puts $fp "test\ntest"
close $fp
set fp [open "input.txt" r]

while { [gets $fp data] >= 0 } {
    puts $data
}
close $fp
```

When the above code is compiled and executed, it reads the file created in previous section and produces the following result:

```
test
test
```

```
Loading [MathJax]/jax/output/HTML-CSS/jax.js
```