

Bash Scripting Cheat Sheet - Cody Short

<p>Hello World: #!/usr/bin/env bash</p> <p>echo Hello World!</p> <p>Command Substitution:</p> <p>date = \$(date)</p> <p>echo \$date</p>	<p>Variables in Bash:</p> <p>SecretMessage = "Don't forget the shebang"</p> <p>echo \$SecretMessage</p> <p>Command line variables:</p> <p>\$1, \$2, etc.</p>
<p>If-then in Bash:</p> <pre>if [<condition>]; then <statements> elif [<condition>]; then <statements> else <statements></pre>	<p>While loop:</p> <pre>while [<condition>] do <statements> done</pre>
<p>For loops (foreach):</p> <pre>for item in <list> do <commands> done</pre>	<p>Range loop (more like for):</p> <pre>for val in {1...100} do <commands> done</pre>

Bash scripts make it easier to automate running system tasks.

Special if-statement conditions:

-e "Path"	Does a file exist?
-r "Path"	Is a file readable?
-d "Path"	Is the file a directory?

More of these conditions can be found here:

http://tldp.org/LDP/Bash-Beginners-Guide/html/sect_07_01.html

Ways in which bash scripts are useful:

- Automating system task for system admins
- Startup scripts
- Fast prototyping of programs with system access
- Eliminate a repetitive process
- Automating server rollouts
- Automatically backup files