**How To Write First UNIX Korn Shell Script Program**

by Vivek Gite · [3 comments](http://www.cyberciti.biz/faq/how-to-write-first-unix-korn-shell-script-program/#comments)

This entry is part 1 of 3 in the series [UNIX Korn Shell Scripting](http://www.cyberciti.biz/faq/series/unix-korn-shell-scripting/)

[UNIX Korn Shell Scripting](http://www.cyberciti.biz/faq/series/unix-korn-shell-scripting/)

* How To Write First UNIX Korn Shell Script Program
* [Korn Shell Variables](http://www.cyberciti.biz/faq/unix-linux-korn-shell-variables/)
* [KSH IF Command Conditional Scripting Examples](http://www.cyberciti.biz/faq/ksh-if-command-examples/)

[](http://www.cyberciti.biz/faq/category/unix/)

Can you tell me how to write a UNIX korn shell script program? I'm new to UNIX scripting and HP-UX UNIX operating system.  
  
Korn shell scripting under UNIX / Linux can be used to automate lots of stuff. It easy to write a shell script. You must know how to use a text editor such as vi to write a script.

**Writing your first Korn shell script**

Let us write a shell program to print knowledge is power on screen. Type the following command to open file:  
vi hello.ksh  
The first line should be as follows:

*#!/bin/ksh*

It is called a shebang. It consists of a number sign and an exclamation point character (#!), followed by the full path to the interpreter such as /bin/ksh. All scripts under UNIX execute using the interpreter specified on a first line.  
Next append code as follows:

*# A shell script to print message*

*# Written by Tom - Jan/13/2008*

print "Knowledge is power"

Save and close the file. At the end your script should look like as follow

*#!/bin/ksh*

*# A shell script to print message*

*# Written by Tom - Jan/13/2008*

*# ------------------------------*

print "Knowledge is power"

**Set executable permission**

Type the following command to set executable permission:  
chmod +x hello.ksh

**Run your korn shell script**

Type the following command:  
./hello.ksh  
Sample output:

Knowledge is power

**What is Shell Scripting**

**Category:** [**UNIX**](http://www.exforsys.com/tech-articles/unix.html)[**Comments (0)**](file:///C:\Documents%20and%20Settings\hii\Desktop\unix\what-is-shell-scripting.html#art_cmts)

**Basic Definition of Shell Scripting is Storing Frequently Used Commands in Files.**

**Store the following in a file named** [**simple.sh**](http://users.sdsc.edu/%7Esteube/Bshell/Scripts/simple.sh) **and execute it.**

**#!/bin/sh**

**# Generate some useful info for**

**# use at the start of the day**

**date**

**cal**

**last $USER | head -6**

**Shows current date, calendar, and a six of your previous logins for security check. You might run this at the beginning of each day.Notice that the commands themselves are not displayed, only the results.To display the commands verbatim as they run, execute with sh -v simple.sh**

**To echo the commands after variable translation, execute with**

**sh -x simple.sh**

**With -v or -x (or both) you can easily relate any error message that may appear to the command that generated it**

**When an error occurs in a script, the script continues executing at the next command**

**Verify this by changing 'cal' to 'caal' to force an error, and then run the script again**

**Run the 'caal' script with 'sh -v simple.sh' and 'sh -x simple.sh' and verify the error message comes from cal**

**Now you can re-use commands easily and save some typing and mistakes**

**Other standard variable names include: HOME, PATH, TERM, PAGER, PRINTER**