

# What is a shell?

Shell is an environment in which we can run commands, programs and Shell scripts. There are different versions of Shell, which differ only from the version of the operating system. Each version of Shell has its own set of recognized commands and functions.

Shell is an environment in which we can run commands, programs and Shell scripts. There are different versions of Shell, which differ only from the version of the operating system. Each version of Shell has its own set of recognized commands and functions.

Shell provides you with an interface to Unix systems. It collects input from you and runs programs based on that input. When a program completes, it displays the output (output) of that program.

## The command prompt line Shell in Unix / Linux

Command prompt, \$, is notified by Shell. While the prompt is displayed, you can enter a command.

Shell reads the input you enter after you press Enter. It determines the command you want to run by looking at the first word in your input. A word is an unbroken set of characters. Spaces and tabs separate words.

Here is a simple example of the date command that displays the current date and time:

```
$ date Thu Jun 25 08 : 30 : 19 MST 2009
```

You can customize your command prompt using the PS1 environment variable explained in the Work environment chapter.

## Shell types in Unix / Linux

There are two main types of Unix in Unix:

1. Bourne Shell style. If you are using a Bourne Shell, the command line is the default \$ character.
2. Type C Shell. If you are using C Shell type, the default prompt line is% character.

In Bourne Shell, there are other subtypes as follows:

Bourne Shell (sh)

Korn Shell (ksh)

Bourne Again Shell (bash)

POSIX Shell (sh)

In type C Shell there are other subtypes:

C Shell (csh)

TENEX / TOPS C Shell (tcsh)

The first Unix Shells were recorded in the mid-1970s by Stephen R. Bourne while he was working at the AT&T Bell Labs in New Jersey.

Bourne Shell is the first Shell to appear on Unix systems, so it is referred to as "the Shell".

Bourne Shell is usually installed as / bin / sh on most Unix versions. For this reason, it is Shell selected to write scripts to use on different versions of Unix.

In this chapter, we refer to most of the Shell concepts based on Bourne Shell.

## Shell Scripts in Unix / Linux

The basic concept of a Shell script is a list of commands, which are listed in order of execution. A good Shell script will have comments, preceded by the # sign, describing the steps.

There are conditional checks, such as value A greater than value B, loops that allow us to handle large amounts of data, files to read and store data and variables to read and store data, and the script may include functions.

Shell scripts and functions are interpreted. This means they are not compiled.

We are going to write many scripts in the next tutorial. It can be a simple text file in which we place all necessary commands and structures that tell the Shell environment what is done and when it is done.

## Examples of scripts in Unix / Linux

Suppose we create a test.sh. Remember that all scripts should have an .sh extension. Before you add anything to your script, you need to tell the system that a prepared Shell script is started. This is done using the shebang structure. For example:

```
#!/ bin / sh
```

This command tells the system that the following commands are run by Bourne Shell. It is called a shebang because the # symbol is called a hash and the symbol! called a state.

To create a script that contains these commands, first place the shebang line and then add the commands:

```
#!/ bin / bash pwd ls
```

## Comments Shell in Unix / Linux

You can put Shell Comments in the script as follows:

```
#!/bin/bash # Author: Zara Ali # Copyright (c) Tutorialspoint.com # Script
```

Now save the above content and make this script run as follows:

```
$ chmod +x test . sh
```

Now you have your Shell script ready to run as follows:

```
$ ./test . sh
```

This produces the following result:

```
/home/amrood/index . jsp unix - basic_utilities . jsp unix - directories
```

**Note** : To run any of your programs in the current directory, you should run using **./program\_name**

## Shell script extension in Unix / Linux

Shell scripts have some required structures that tell the Shell environment what to do and when to do it. Of course, most scripts here are more complicated than the ones introduced above.

Shell is a real, complete program language with variables, control structures and . No matter how complicated it is, however, it is still a list of commands executed in a way continuity.

The following scripts use the **read** command which receives input from the keyboard and assigns it to the PERSON variable and finally prints it on the STDOUT variable.

```
#!/bin/sh # Author: Zara Ali # Copyright (c) Tutorialspoint.com # Script f
```

Below is a sample run of the script above:

```
$ ./test . sh What is your name ? Zara Ali Hello , Zara Ali $
```

### According to Tutorialspoint

Previous article: [Micro editor in Unix / Linux](#)

Next lesson: [Use variables in Shell](#)

You finished reading the article "**What is a shell?**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.