**BASH**

BASH is an acronym for Bourne Again Shell, a punning name, which is a tribute to Broune Shell.

Bash is a shell program written by Brian Fox as an upgrade version of Bourne Shell program '**sh**'. It is an open-source GNU project. It was released in 1989 as one of the most popular shell distributions of GNU/Linux operating systems. It provides functional improvements over Bourne Shell for both programming and interactive uses. It includes command line editing, key bindings, command history with unlimited size, etc.

In basic terms, Bash is a command line interpreter that typically runs in a text window where user can interpret commands to carry out various actions. The combination of these commands as a series within a file is known as a Shell Script. Bash can read and execute the commands from a Shell Script.

Bash is the default login shell for most Linux distributions

**Shell**: A UNIX Shell is a program or a command line interpreter that interprets the user commands which are either entered by the user directly or which can be read from a file and then pass them to the operating system for processing. It is important to note that Shall scripts are interpreted and not compiled, as the computer system interprets them and there is not any need to compile Shell Scripts in order of execution.

There are different types of shells available in Linux Operating Systems. Some of which are as follows:

1. Bourne Shell
2. C shell
3. Korn Shell
4. GNU Bourne Shell

To know, which shell types of your operating system supports, type the command into the terminal as given below:

* **cat /etc/shells**

And to know where bash is in your OS, type the below command and you will get a specific location:

* **which bash**

**Features of Bash**

1. Bash is **sh-compatible** as it derived from the original UNIX Bourne Shell. It is incorporated with the best and useful features of the Korn and C shell like directory manipulation, job control, aliases, etc.
2. Bash can be **invoked by** single-character command line options (**-a, -b, -c, -i, -l, -r, etc.** ) as well as by multi-character command line options also like --debugger, --help, --login, etc.
3. Bash **Start-up files** are the scripts that Bash reads and executes when it starts. Each file has its specific use, and the collection of these files is used to help create an environment.
4. Bash consists of **Key bindings** by which one can set up customized editing key sequences.
5. Bash contains **one-dimensional arrays** using which you can easily reference and manipulate the lists of data.
6. Bash comprised of **Control Structures** like the **select construct** that specially used for menu generation.
7. Directory Stack in Bash specifies the history of recently-visited directories within a list. Example: **pushd** builtin is used to add the directory to the stack, **popd** is to remove directory from the stack and **dirs** builtin is to display content of the directory stack.
8. Bash also comprised of restricted mode for the environment security. A shell gets restricted if bash starts with name **rbash**, or the bash --restricted, or bash -r option passed at invocation.

ADVANTAGES OF BASH SCRIPTING

* Automation: Shell script allow you to automate repetitive tasks and processes and saving time.
* Portability: Shell script can be run on various platforms and operating systems, including Unix, Linux, macOS, and even Windows through the use of emulators or virtual machines.
* Flexibility: Shell scripts are highly customizable and can be easily modified to suite specific requirements.
* Accessibility: Shell scripts are easy to write and don’t require any special tools or software.
* Integration: Shell scripts can be integrated with other tools and applications such as databases, web servers, and cloud services, allowing for more complex automation and system management tasks.
* Debugging: Shell scripts are easy to debug , and most shells have built-in debugging and error-reporting tools that can help identify and fix issues quickly.

Bash Script

Bash Script is a computer program written in the bash programming language.

**How to create and run a bash script?**

* To create an empty bash script, first change the directory in which you want to save your script using cd command.
* Try to use text editor like gedit in which you want to type the shell commands.
* Use **touch** command to cresate the zero bytes sized script. Ex: touch filename.
* To open the script in the text editor (eg. gedit) type gedit filename.
* Here, .sh is suffixed as an extension that you have to provide for execution.
* Type the shell commands for your bash script in the newly opened text window or the text editor. Before typing bash shell commands, first, look at the base of any bash script.
* Each Bash based Linux script starts by the line-

1. #! /bin/bash

Where #! is referred to as the ***shebang*** and rest of the line is the path to the interpreter specifying the location of bash shell in our operating system.

Bash use # to comment any line.

Bash use **echo** command to print the output.

At the end, execute the bash script prefixing with ./ .