**BASH SCRIPTING**

**Bash:** It is a command line language. The name stands for **Bourne Again Shell**. Ist released in 1989.

In the context of Bash Scripts we are telling the Bash Shell what it should do.

Note:-Bash is a case sensitive environment.

**Bash Script:** A Bash Script is a plain text which contains a series of commands.

**cat/etc/shells**

It shows all the shells available on your system.

**which bash**

To know the path of bash.

**Variables**

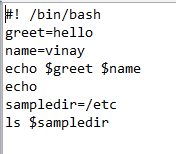
A variable is a temporary store for a piece of information. There are two actions we may perform for variables:

* Setting a value for a variable.
* Reading the value for a variable.

**Special Variable**

* **$0** - The name of the Bash script.
* **$1 - $9** - The first 9 arguments to the Bash script.
* **$#** - How many arguments were passed to the Bash script.
* **$@** - All the arguments supplied to the Bash script.
* **$?** - The exit status of the most recently run process.
* **$$** - The process ID of the current script.
* **$USER** - The username of the user running the script.
* **$HOSTNAME** - The hostname of the machine the script is running on.
* **$SECONDS** - The number of seconds since the script was started.
* **$RANDOM** - Returns a different random number each time is it referred to.
* **$LINENO** - Returns the current line number in the Bash script

**An Example of variables**.

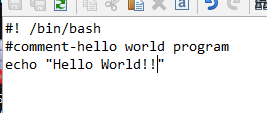


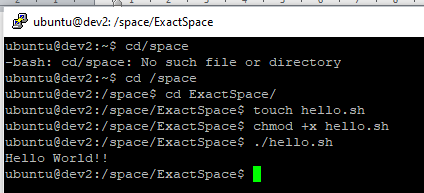


**echo**

For printing a statement.

**A Hello World Program……**





**How we can share output to a file?**

**>filename.txt**

**If we want to write on terminal and everything that we write save in a text file.**

#! /bin/bash

cat >filename.txt

**If we want text is not replaced but appended**

**#! /bin/bash**

**cat >>filename.txt**

**For Single line comment-#**

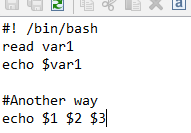
**For multiline Comment-**

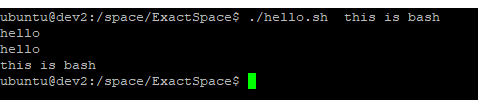
**:’ this is a comment**

**This is a comment**

**This is a comment ‘**

**Ask the user for input using read**

* **read var1**
* 



**To take multiple input through array**

**Args= ( “$@” )**

**echo $ {args[0]} $args{args[1]} ${args[2]}**

**To Print all input**

**echo $@**

**To know the length of the array**

**echo $#**

**Another way**

**#! /bin/bash**

**while read line**

**do**

**Echo “$line”**

**done “${1:-/dev/stdin}”**

**Creating a doc inside sh**

**#! /bin/bash**

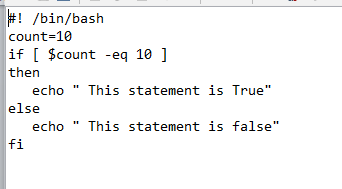
**cat << hello(Any name that you want)**

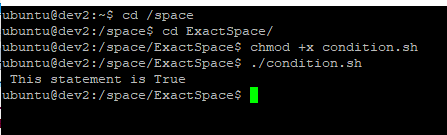
**this is hello creative text**

**add another line**

**hello**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Conditional Staement\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***





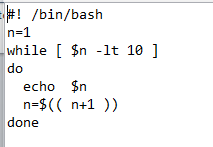
**-ne:not equal**

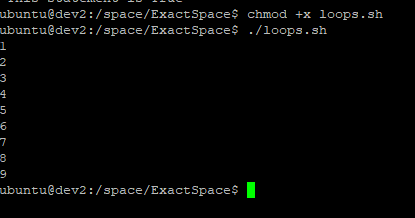
**-gt:greater than**

**(we can also use > for greater than and > for less than)**

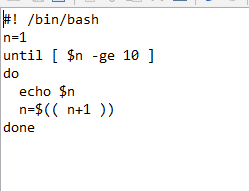
**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Loops\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

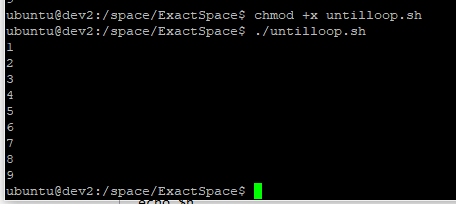
**1.while loop(for true statement)**



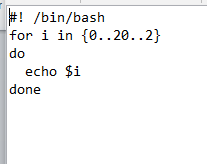


**2.until loop(run until condition becomes true i.e. for false statement)**

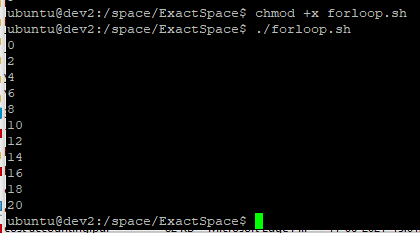




**3.for loop**



**{start..ending..increment/decrement}**



**Another way**

**#! /bin/bash**

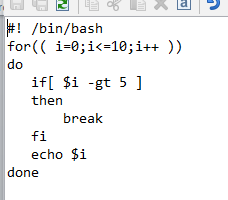
**For(( i=0;i<5;i++ ))**

**do**

**echo $i**

**done**

**break and continue statement**



**Similarly for continue**