BASH SCRIPT

1. HELLO SCRIPT
Create a new variable as print
To create a new script file

```
hepzi@ubuntu:~$ touch hello.sh
hepzi@ubuntu:~$ nano hello.sh
hepzi@ubuntu:~$ cat hello.sh
#!/bin/bash
print="Hello Script"
echo $print
hepzi@ubuntu:~$ bash hello.sh
Hello Script
```

2.Echo command to print text

```
hepzi@ubuntu:~$ nano echo.sh
hepzi@ubuntu:~$ cat echo.sh
#!/bin/bash
var="LINUX"
echo 'I am still learning about $var'
echo "I am still in training phase"
hepzi@ubuntu:~$ bash echo.sh
I am still learning about $var
I am still in training phase
```

3.Terminal to sleep for a specified amount of time(delaying execution)

```
hepzi@ubuntu:~$ nano sleep.sh
hepzi@ubuntu:~$ bash sleep.sh
Im going to turn off the laptop
Turn on the laptop
hepzi@ubuntu:~$ cat sleep.sh
sleep 10 &&
echo "Im going to turn off the laptop" &&
sleep 5 &&
echo "Turn on the laptop"
```

4.Wait for any running process to complete

```
hepzi@ubuntu:~$ nano wait.sh
hepzi@ubuntu:~$ bash wait.sh
User
Ubuntu
VM
hepzi@ubuntu:~$ cat wait.sh
#!/bin/bash
sleep 5 &
sleep 3 &
echo "User"
echo "Ubuntu"
wait
echo "VM"
```

5.Comments

```
hepzi@ubuntu:~$ nano comments.sh
hepzi@ubuntu:~$ cat comments.sh
#This is a single line comment
:'
Type something
'
#This is a multiline comment
```

6.Get user input

```
hepzi@ubuntu:~$ nano read.sh
hepzi@ubuntu:~$ bash read.sh
Enter your name
Hepzi
Hello, Hepzi
Enter your age
00
hepzi@ubuntu:~$ cat read.sh
echo "Enter your name"
read name
echo "Hello, $name"
echo "Enter your age"
read age
```

LOOPS

7.While loop

```
hepzi@ubuntu:~$ nano whileloop.sh
hepzi@ubuntu:~$ bash whileloop.sh
Reverse order: 10
Reverse order: 9
Reverse order: 8
Reverse order: 7
Reverse order: 6
Reverse order: 5
Reverse order: 4
Reverse order: 3
Reverse order: 2
Reverse order: 1
hepzi@ubuntu:~$ cat whileloop.sh
#!/bin/bash
n=10
while [ $n -gt 0 ] ;
do
echo Reverse order: $n
((n--))
done
```

8.For loop

```
hepzi@ubuntu:~$ nano forloop.sh
hepzi@ubuntu:~$ bash forloop.sh
Typing
Typing
Typing
```

Typing
Typing
Typing
hepzi@ubuntu:~$ cat forloop.sh
#!/bin/bash
for ((i=0; i<=5; i++));
do
  echo "Typing"
done

8.Create an array
hepzi@ubuntu:~$ nano array.sh
hepzi@ubuntu:~$ bash array.sh
3

egg
burger
milk
hepzi@ubuntu:~$ cat array.sh
#!/bin/bash
array=(1 2 3 4 5 6 7)
echo ${array[2]}

IndexedArray=(egg burger milk)
#Iterate over the array to get all the values
for i in "${IndexedArray[@]}";
do
  echo "$i";
done

9.Conditional statements
hepzi@ubuntu:~$ nano if.sh
hepzi@ubuntu:~$ bash if.sh
Enter your age
20
ELIGIBLE
hepzi@ubuntu:~$ cat if.sh
#!/bin/bash
echo "Enter your age"
read age
if [$age -ge 18];
then
    echo "ELIGIBLE"
else
    echo "NOT ELIGIBLE"
fi

10.Functions
hepzi@ubuntu:~$ nano funcn.sh
hepzi@ubuntu:~$ bash funcn.sh
Hello World!
Welcome To Bash script

hepzi@ubuntu:~$ cat funcn.sh
Without argument
```
#!/bin/bash
hello () {
  echo 'Hello World!'
}
hello
```

With argument
```
funcn_args ()
{
  echo $1 $2 $3
}
funcn_args "Welcome" "To" "Bash script"
```

11.Display string length
hepzi@ubuntu:~$ nano stringlength.sh
hepzi@ubuntu:~$ bash stringlength.sh
Length: 36
hepzi@ubuntu:~$ cat stringlength.sh
```
#!/bin/bash
mystring="lets count the length of this string"
i=${#mystring}
echo "Length: $i"
```

12.Extract string
cut -d (denotes cut the string separated by comma)
-f (field, To Extract the value)
hepzi@ubuntu:~$ nano extract.sh
hepzi@ubuntu:~$ bash extract.sh
5005
Hostinger
hepzi@ubuntu:~$ cat extract.sh
```
#!/bin/bash
cut -d , -f 5 <<< "Website,Domain,DNS,SMTP,5005"
```
To extract only text using expr substr (the start position and the string length)
```
expr substr "458449Hostinger4132" 7 9
```

13.Find and replace string
hepzi@ubuntu:~$ nano find.sh
hepzi@ubuntu:~$ bash find.sh
I drive a Audi and Volvo
hepzi@ubuntu:~$ cat find.sh
```
#!/bin/bash
first="I drive a BMW and Volvo"
second="Audi"
echo "${first/BMW/"$second"}"
```

14.Concatenate strings
hepzi@ubuntu:~$ nano concatenate.sh
hepzi@ubuntu:~$ bash concatenate.sh
The secret is...Bash

```
hepzi@ubuntu:~$ cat concatenate.sh
firststring="The secret is..."
firststring+="Bash"
echo "$firststring"
```

15.Check if a number is even or odd

```
hepzi@ubuntu:~$ nano evenoddnumber.sh
hepzi@ubuntu:~$ bash evenoddnumber.sh
Enter a number :7
Number is even
hepzi@ubuntu:~$ cat evenoddnumber.sh
#!/bin/bash
read -p "Enter a number :"
if [ $((mynumber%2)) -eq 0 ];
then
echo "Number is even"
else
echo "Number is odd"
fi
```

16.Generate factorial of number

```
hepzi@ubuntu:~$ nano factorial.sh
hepzi@ubuntu:~$ bash factorial.sh
Enter the number you want to get factorial
5
120
hepzi@ubuntu:~$ cat factorial.sh
#!/bin/bash
echo "Enter the number you want to get factorial"
read mynumber
factorial=1
for ((i=1;i<=mynumber;i++))
do
factorial=$(($factorial*$i))
done
echo $factorial
```

17.Create directories

To create a set of directories with same sub directories using bash sript

```
hepzi@ubuntu:~$ nano directories.sh
hepzi@ubuntu:~$ sudo bash directories.sh

hepzi@ubuntu:~$ ls
array.sh                        event                  p95v308b17.linux64.tar.gz
comments.sh                     event_datascience      Pictures
concatenate.sh                  event_fun              pswd.sh
_datascience                    event_notes            Public
date.txt                        extract.sh             python
DAY6_nginx_Hepzibah.txt         factorial.sh           python_datascience
DAY7_LAMP_Hepzibah.txt          find.sh                python_fun
DAY9_linux_Hepzibah             forloop.sh             python_notes
Desktop                 _fun    read.sh
```

```
directories.sh              funcn.sh          sleep.sh
Documents                   hello.sh          snap
downloads                   if.sh             stringlength.sh
Downloads                   index.html        Templates
downloads_datascience       jail.local        test-cpu
downloads_fun               latest.tar.gz     Videos
downloads_notes             mm                wait.sh
echo.sh                     Music             whileloop.sh
evenoddnumber.sh            _notes            wordpress
```

```
hepzi@ubuntu:~$ cat directories.sh
#!/bin/bash
mkdir -p {downloads,event,python,}_{notes,fun,datascience}
```

18.Read files

To read a file in bash, you will need to create a sample file first

```
hepzi@ubuntu:~$ nano samplereading.txt
```

Then create the actual script file

```
hepzi@ubuntu:~$ nano readfiles.sh
hepzi@ubuntu:~$ bash readfiles.sh
```

Out of all scripting languages, bash is the most popular one.

It allows programmers to run scripts effortlessly in a variety
of Linux distros.

```
hepzi@ubuntu:~$ cat readfiles.sh
#!/bin/bash
myvalue=`cat samplereading.txt`
echo "$myvalue"
```

19.Print files with line count

```
hepzi@ubuntu:~$ nano cars.txt
hepzi@ubuntu:~$ cat cars.txt
Audi
Mahindra xuv
MG
BMW
volvo
```

```
hepzi@ubuntu:~$ nano printlines.sh
hepzi@ubuntu:~$ cat printlines.sh
#!/bin/bash
myfile='cars.txt'
i=1
while read lines;
do
echo "$i : $lines"
i=$((i+1))
done < $myfile
```

```
hepzi@ubuntu:~$ bash printlines.sh
1 : Audi
2 : Mahindra xuv
3 : MG
4 : BMW
5 : volvo
```

Another method - To find the number of lines using wc (-l total number of lines)

```
hepzi@ubuntu:~$ wc -l < cars.txt
5
```

20.Delete file

```
hepzi@ubuntu:~$ nano deletefile.sh
hepzi@ubuntu:~$ cat deletefile.sh
#!/bin/bash
myfile='cars.txt'
touch $myfile
if [ -f $myfile ]; then
   rm cars.txt
   echo "$myfile deleted"
fi

hepzi@ubuntu:~$ bash deletefile.sh
cars.txt deleted
```

21.Test if file exists

```
hepzi@ubuntu:~$ nano exists.sh
hepzi@ubuntu:~$ bash exists.sh
cars.txt does not exist.

hepzi@ubuntu:~$ cat exists.sh
#!/bin/bash
MyFile=hello.sh
if [ -f "$MyFile" ]; then
echo "$MyFile exists."
else
echo "$MyFile does not exist."
fi

hepzi@ubuntu:~$ bash exists.sh
hello.sh exists.
```

22.Check Inodes and Disk usage

Inodes represent data units on a physical or virtual server. Each text file, video, folder, HTML file, or script is 1 inode.

```
hepzi@ubuntu:~$ nano inode.sh
hepzi@ubuntu:~$ cat inode.sh
#!/bin/bash
find . -printf "%h\n" | cut -d/ -f-2 | sort | uniq -c | sort -rn
du -shc * | sort -rh
```

```
hepzi@ubuntu:~$ bash inode.sh
  19022 ./snap
    575 ./.cache
    168 ./.config
     77 .
     60 ./.local
     23 ./Pictures
     13 ./test-cpu
     11 ./Documents
      9 ./Downloads
      3 ./.rpmdb
      3 ./python
      3 ./.links2
      3 ./event
      3 ./downloads
      2 ./.gnupg
917M  total
828M  snap
44M   test-cpu
22M   latest.tar.gz
8.1M  Downloads
6.9M  p95v308b17.linux64.tar.gz
5.0M  Documents
4.0M  Pictures
20K   DAY6_nginx_Hepzibah.txt
16K   python
16K   event
16K   downloads
12K   DAY7_LAMP_Hepzibah.txt
4.0K  wordpress
4.0K  whileloop.sh
4.0K  wait.sh
4.0K  Videos
4.0K  update.sh
4.0K  Templates
4.0K  stringlength.sh
4.0K  sleep.sh
4.0K  samplereading.txt
4.0K  read.sh
4.0K  readfiles.sh
4.0K  python_notes
4.0K  python_fun
4.0K  python_datascience
4.0K  Public
4.0K  pswd.sh
4.0K  printlines.sh
4.0K  _notes
4.0K  Music
4.0K  mm
4.0K  jail.local
4.0K  inode.sh
4.0K  info.sh
```

4.0K index.html
4.0K if.sh
4.0K hello.sh
4.0K funcn.sh
4.0K _fun
4.0K forloop.sh
4.0K find.sh
4.0K factorial.sh
4.0K extract.sh
4.0K exists.sh
4.0K event_notes
4.0K event_fun
4.0K event_datascience
4.0K evenoddnumber.sh
4.0K echo.sh
4.0K downloads_notes
4.0K downloads_fun
4.0K downloads_datascience
4.0K directories.sh
4.0K Desktop
4.0K deletefile.sh
4.0K DAY9_linux_Hepzibah
4.0K date.txt
4.0K _datascience
4.0K concatenate.sh
4.0K comments.sh
4.0K array.sh

24.Update packages

hepzi@ubuntu:/$ sudo nano update.sh
hepzi@ubuntu:/$ sudo cat update.sh
[sudo] password for hepzi:
#!/bin/bash
sudo apt-get update
sudo apt-get upgrade

hepzi@ubuntu:/$ sudo bash update.sh
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:2 http://security.ubuntu.com/ubuntu jammy-security/main amd64 DEP-11 Metadata [41.5 kB]

25.Show server information

hepzi@ubuntu:~$ nano info.sh
hepzi@ubuntu:~$ cat info.sh
#!/bin/bash
echo "Date"
date
echo "Uptime"
uptime
echo "Memory Usage"
free -m
echo "Network Usage"

ip a
hepzi@ubuntu:~$ bash info.sh

Date
Friday 27 January 2023 05:40:50 PM IST

Uptime
17:40:50 up 22 min, 1 user, load average: 0.58, 0.56, 0.70

Memory Usage

	total	used	free	shared	buff/cache	available
Mem:	1975	1238	66	31	670	547
Swap:	2679	510	2169			

Network Usage
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:69:d2:c8 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
       valid_lft 85107sec preferred_lft 85107sec
    inet6 fe80::324e:b007:a32f:42a4/64 scope link noprefixroute
       valid_lft forever preferred_lft forever