Shell Scripting

1) Guess the number using while loop

```
  GNU nano 6.2
#!/bin/bash

while [ True ];
do
        read -p "Enter No:" num
        if [ $num -eq 40 ]
        then
                echo "correct"
                break;
        elif [ $num -lt 40 ]
        then
                echo "Not enough"
        else
                echo "too much"
        fi
done
```

```
tirth@Ubuntu: ~/demo
tirth@Ubuntu:~/demo$ nano ex1.sh
tirth@Ubuntu:~/demo$ bash ex1.sh
Enter No:33
Not enough
Enter No:44
too much
Enter No:40
correct
tirth@Ubuntu:~/demo$
```

2) if person age is greater than 18 and have voter id --> eligible to vote. if less than 18 --> need to wait

```
tirth@Ubuntu: ~/demo
  GNU nano 6.2                                          ex2.sh
#!/bin/bash

read -p "Enter Age:" age
if [ $age -ge 18 ];
then
        read -p "Do you have Voter ID:(1/0)" vid
        if [ $vid -eq 1 ];
        then echo You are eligible for voting.
        else echo you are not eligible for voting.

        fi
else echo you are under age.
fi
```

```
tirth@Ubuntu: ~/demo
tirth@Ubuntu:~/demo$ nano ex2.sh
tirth@Ubuntu:~/demo$ bash ex2.sh
Enter Age:19
Do you have Voter ID:(1/0)0
you are not eligible for voting.
tirth@Ubuntu:~/demo$ bash ex2.sh
Enter Age:23
Do you have Voter ID:(1/0)1
You are eligible for voting.
tirth@Ubuntu:~/demo$
```

3) Get 3 numbers from the user and find the greater number

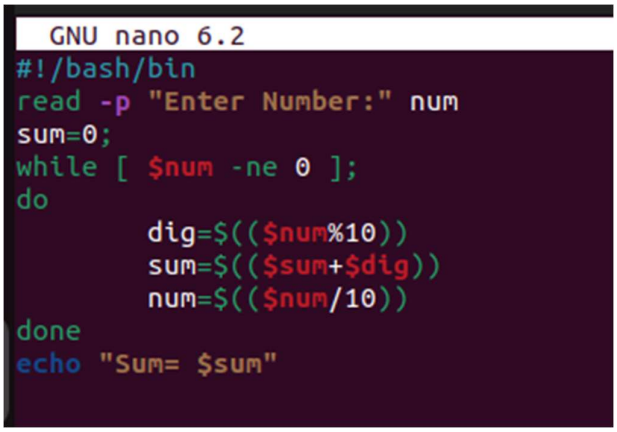
```
tirth@Ubuntu: ~/demo
 GNU nano 6.2                                  file.sh
#!/bash/bin
read -p "Enter No1:" a
read -p "Enter No2:" b
read -p "Enter No3:" c
if [ $a -gt $b ] && [ $a -gt $c ];
then
echo "$a is greater."
elif [ $b -gt $c ] && [ $b -gt $a ];
then
echo "$b is greater."
else
echo "$c is greater."
fi
```

```
tirth@Ubuntu: ~/demo
tirth@Ubuntu:~/demo$ nano file.sh
tirth@Ubuntu:~/demo$ bash file.sh
Enter No1:7
Enter No2:6
Enter No3:5
7 is greater.
tirth@Ubuntu:~/demo$
```

4) give 5 change to user to guess the number using for loop


```
tirth@Ubuntu:~/demo$ nano ex4.sh
tirth@Ubuntu:~/demo$ bash ex4.sh
Enter No:22
Not enough
4 attempt left
Enter No:33
Not enough
3 attempt left
Enter No:44
too much
2 attempt left
Enter No:55
too much
1 attempt left
Enter No:40
correct
tirth@Ubuntu:~/demo$
```

5) find the sum of digits of number entered.



```
tirth@Ubuntu:~/demo$ nano ex5.sh
tirth@Ubuntu:~/demo$ bash ex5.sh
Enter Number:123
Sum= 6
tirth@Ubuntu:~/demo$ bash ex5.sh
Enter Number:562
Sum= 13
```

6) find the reverse of the number entered.

```
  GNU nano 6.2
#!/bash/bin
read -p "Enter No:" num
sum=0;
while [ $num -ne 0 ];
do
        dig=$(($num%10))
        sum=$(($sum*10))
        sum=$(($sum+$dig))
        num=$(($num/10))
done
echo "Reversed No: $sum"
```

```
tirth@Ubuntu:~/demo$ touch ex6.sh
tirth@Ubuntu:~/demo$ nano ex6.sh
tirth@Ubuntu:~/demo$ bash ex6.sh
Enter No:123
Reversed No: 321
```

7) find whether the given number in prime or not

```
  GNU nano 6.2
#!/bash/bin
read -p "Enter No:" num
count=0;
for((i=2;i<$num;i++))
do
        n=$(($num%$i));
        if [ $n -eq 0 ]
        then
                count=$(($count+1))
        fi
done
if [ $count -eq 0 ]
then
        echo "Prime No."
else
        echo "Not prime No."
fi
```

```
tirth@Ubuntu:~/demo$ touch ex7.sh
tirth@Ubuntu:~/demo$ nano ex7.sh
tirth@Ubuntu:~/demo$ bash ex7.sh
Enter No:20
Not prime No.
tirth@Ubuntu:~/demo$ nano ex7.sh
tirth@Ubuntu:~/demo$ bash ex7.sh
Enter No:5
Prime No.
tirth@Ubuntu:~/demo$
```