1. Write a shell script to check whether the entered number is prime or not.

```
echo -n "Enter number for checking: " read n
isPrime=true
for (( i=2; i<$n; i++)); do if [ `expr
$n % $i` -eq 0 ]; then
isPrime=false
break
fi
Done
if $isPrime; then
echo "Entered number is a Prime number"
else
echo "Entered number is not a Prime number"
fi
```

```
21012021003@telnetserver:~$ ./p4_1.sh
Enter number for checking : 9
Entered number is not a Prime number
```

2. Write a shell script to calculate HRA of employees depending upon their basic.

```
echo -n "Enter your salary :"
read salary
echo -n "Do you live in city(Y/N) :"
read city
city=${city:-"N"}
if [ $city == 'y' ] || [ $city == 'Y' ]; then
hra=$(bc <<< "scale=2; $salary * 0.5")
elif [ $city == 'n' ] || [ $city == 'N' ]; then
hra=$(bc<<< "scale=2; salary * 0.4")
fi
echo "HRA: $hra/- Rs"
```

```
21012021003@telnetserver:~$ ./p4_2.sh
Enter your salary :1000

Do you live in city(Y/N) :y

HRA: 500.0/- Rs
```

3. Write a shell script that greets the user by saying Good Morning, Good Afternoon, and Good Evening according to the system time.

```
>>>> time=`date
+%H`
if [ $time -ge 6 ] && [ $time -lt 12 ]; then
 echo "Good Morning"
elif [ $time -ge 12 ] && [ $time -lt 17 ]; then
echo "Good Afternoon" else

echo "Good Evening" fi
```

```
21012021003@teTnetserver:~$ ./p4_3.sh'
Good Morning .
```

4. Write a shell script, which takes a filename as command line argument, asks the user if he wants to revoke the read, write permissions for the group and others for that particular file. If the answer is "y" then it should do so or else it should abort the operation.

```
echo -n "Do you want to revoke read/write permission for group(y/n): " read ch
if [ $ch == "y" ] || [ $ch == "Y" ]; then
chmod g-rw $1 echo "operation
Successful"
ls -l else
echo "operation Aborted"
fi
```

```
-rw-rw-r--   1   21012021003   21012021003   286   Jan   NG
-PWXPWXP-X   1   21012021003   21012021003   112   Jan   a.sh
-PWXPWXP-X   1   21012021003   21012021003   317   Feb   b.sh
-PWXPWXP-X   1   21012021003   21012021003   351   Feb   case.sh
-PWXPWXP-X   1   21012021003   21012021003   292   Feb   c.sh
-rwXrwxr-x   1   21012021003   21012021003   223   Feb   d.sh
drwxrwxr-x   1   21012021003   21012021003   4096  Jan   meet
-PWXPWXP-X   1   21012021003   21012021003   243   Mar   p3_1.sh
```

5. Write a shell script that asks the capital of Gujarat and repeats the question until the user gives correct answer.

```
>>>> state="Gujrat"
capital="gandhinagar"
while(true); do
echo -n "Capital of $state: " read
ans
if [ $ans == $capital ]; then
echo "Congratulation, Correct Answer..." break
```

```
fi
echo "Wrong Answer"
done
```

```
21012021003@telnetserver:~$ ./p4_5.sh'
Capital of Gujrat: Gandhinagar
Congratulation, Correct Answer.
```

6. Write a shell script to display desired line from a file.

```
echo -n "Enter file name: " read
file
echo -n "Enter line num: "
read n head -$n $file | tail -1
```

```
21012021003@telnetserver:~$ ./p4_6.sh
Enter file name: max.txt

Enter line num: 3

Bye
```

7. Write a shell script to count number of newline characters in a file.

```
echo -n "Enter file name: " read
file
lineCount=$(wc -l $file | cut -d " " -f1) lineCount=`expr $lineCount + 1`
echo "$file contains $lineCount lines"
```

```
21012021003@telnetserver:~$ ./p4_7.sh
Enter file name: max.txt
max. txt contains 5 Tines
```

8. Write a shell script to count number of spaces in a file.

```
echo -n "Enter file name: " read
file
spaceCount=$(grep -o " " $file | wc -l)
echo "$file contains $spaceCount spaces"
```

```
21012021003@telnetserver:~$ ./p4_8.sh
Enter file name: max.txt
max.txt contains 0 spaces
```

9. Write a Shell script, which counts the number of words in a file, without taking into consideration the blank space, tab spaces and the newline characters using WC.

```
echo -n "Enter file name: " read
file
wordCount=$(wc -w $file | cut -d " " -f1)
echo "$file contains $wordCount words"
```

```
21012021003@telnetserver:~$ ./p4_9.sh
Enter file name: max.txt
max. txt contains 4 words
```

10. Write a Shell script, which counts the number of words in a file, without taking into consideration the blank space, tab spaces and the newline characters without using WC.

```
echo -n "Enter file name: "
read file
words=($(grep -E '\w+' $file))
wordCount=${#words[@]}
echo "$file contains $wordCount words"
```

```
21012021003@telnetserver:~$ ./p4_10.sh
Enter file name: max.txt
max.txt contains 4 words
```

11. Write a Shell script, which counts the number of characters in a file, without taking into consideration the blank space, tab spaces and the newline characters using WC.

```
echo -n "Enter file name: " read
file
charCount=$(wc -c $file | cut -d " " -f1)
echo "file $file contains $charCount characters"
```

```
21012021003@telnetserver:~$ ./p4_11.sh
Enter file name: max.txt
file max.txt contains 22 characters
```

12. Write a Shell script, which counts the number of characters in a file, without taking into consideration the blank space, tab spaces and the newline characters without using WC.

```
echo -n "Enter file name: " read
file
charCount=$(cat $file | tr -c '\n' '[\n*]' | grep -c '^')
echo "file $file contains $charCount characters"
```

```
21012021003@telnetserver:~$ ./p4_12.sh
Enter file name: max.txt
file max.txt contains 22 characters
```