# ASSIGNMENT: SHELL PRACTISE QUESTIONS

## 1. To find profit or loss given the cost price and selling price.

```
#!/bin/bash

echo "Enter Cost price: "
read costp
echo "Enter Selling price: "
read sellp

if [ $costp -eq $sellp ]
then
    echo "There is no profit or loss"
    exit 1
fi
if [ $costp -gt $sellp ]
then
    echo "There is loss of Rs.`expr $costp - $sellp` "
else
    echo "There is profit of Rs.`expr $sellp - $costp` "
fi
```

## 2. To check the given integer is prime or not.

```
#!/bin/bash

echo "Enter a number : "
read num
i=2

while [ `expr $num / 2` -gt $i ]
do
   if [ `expr $num % $i` -eq 0 ]
       then
  echo "$num is not a Prime number"
   exit 1
        Fi
 i=`expr $i + 1`
done
echo "$num is a Prime number"
```

3. **To check the given number is odd or even.**

```
#!/bin/sh
echo "Enter the number you want to check for : "
read num
rem=$(( $num % 2 ))
if [ $rem -eq 0 ]
then
    echo "$num is a even number"
else
    echo "$num is a odd number"
fi
```

4. **To find the area and circumference of a circle.**

```
#!/bin/sh

pi=3.14
echo "Please enter the circle's radius : "
read radius

area=$( echo "$pi * $radius * $radius" | bc )
echo "Area = $area"

cir=$( echo "2 * $pi * $radius" | bc )
echo "The Circumference of the circle = $cir"
```

5. **To take subject marks as input from user and display the result (I-division, II-division, III-division, fail)**

```
#!/bin/sh

echo "Enter Subject Marks : "
read marks
if [ $marks -gt 80 ]
then
    echo "DIVISION I"
elif [ $marks -lt 80 ] && [ $marks -gt 70 ]
then
    echo "DIVISION II"
elif [ $marks -lt 70 ] && [ $marks -gt 50 ]
then
    echo "DIVISION III"
else
    echo "FAIL"
fi
```

6. **Simple calculator for basic arithmetic operations. (take operand-1, operator and operand-2 from user as input)**

```
#!/bin/bash
echo "Enter first operator: "
read arg1
echo "Enter second operator: "
read arg2
echo "Enter operator(+,-,/,*,%)"
read orp
case "$orp" in
    '*') result=$(($arg1 * $arg2))
        ;;
    '/') result=$(($arg1 / $arg2))
        ;;
    '-') result=$(($arg1 - $arg2))
        ;;
    '%') result=$(($arg1 % $arg2))
    ;;
    '+') result=$(($arg1 + $arg2))
        ;;
    *) echo "Unknown operator '$op'"
        ;;
esac
echo "result = $result"
```

7. **Find the largest of 3 numbers.**

```
#!/bin/sh
echo "Enter Num1"
read num1
echo "Enter Num2"
read num2
echo "Enter Num3"
read num3
if [ $num1 -gt $num2 ] && [ $num1 -gt $num3 ]
then
    echo " number 1: $num1 is the greatest"
elif [ $num2 -gt $num1 ] && [ $num2 -gt $num3 ]
then
    echo " number 2: $num2 is the greatest"
else
    echo " number 3: $num3 is the greatest"
fi
```

**8) to display the result a student has scored (same as 5th program) using *case...esac* statement.**

```
#!/bin/sh

echo "Enter Subject Marks : "
read marks

case "marks" in
    [$marks -gt 80]) echo "DIVISION I"
        ;;
    [ $marks -lt 80 ] && [ $marks -gt 70 ]) echo "DIVISION II"
        ;;
    [ $marks -lt 70 ] && [ $marks -gt 50 ]) echo "DIVISION III"
        ;;
    *) echo "FAIL"
        ;;
Esac
```

**9) to check if a file exists if yes then write "hello world" in the file if no then create the file and then write "hello world" line.**

```
#!/bin/bash

echo "Please enter File name:"
read filename

if [ -e "$fileName" ]

 then
    echo "File exists"
    echo "Hello World!!" > "$fileName"
    cat "$fileName"
else
    echo "File does not exists"
    echo "Hello World!!" >> "$fileName"
    cat "$fileName"
fi
```

**10) Make a directory after checking of its existence.**

```
#!/bin/bash

echo "Please enter directory name"
read dirname

if [ ! -d "$dirname" ]

then
    echo "Directory does not exist. Creating a new directory now."

    mkdir ./$dirname

else
    echo "Directory exists"
fi
```

**11) To read the contents of a file and display it line by line in command prompt. (Hint: use *read* command to read the file contents.)**

```
#!/bin/bash

filename='Hello_142103005.txt'

while read line

do

echo "$line"

done < "$filename"
```

## 12) to check and display 10 leap years from year 2000.

```
#!/bin/bash

echo "Please enter the year : "
read year
i=0
echo "The Leap Years:"

while [ $i -ne 10 ]
do
        if [ `expr $year % 400` -eq 0 ]
        then
                echo "$year"
                i=`expr $i + 1`
        elif [ `expr $year % 100` -eq 0 ]
        then
                :
        elif [ `expr $year % 4` -eq 0 ]
        then
                echo "$year"
                i=`expr $i + 1`
        else
                :
        fi
        year=`expr $year + 1`
done
```

## 13) To check the given string is palindrome or not.

```
#!/bin/sh

echo Please enter the string
read str
echo $str > temp
rstr="$(rev temp)"
if [ $str = $rstr ]
then
echo "Yes, Its a Palindrome"

else
echo "No, Its not a Palindrome"
fi
```

**14) To find the factorial of a number.**

```
#!/bin/bash

echo "Please enter any number : "
read num
fact=1
i=2
while [ $i -le $num ]
do
        fact=`expr $fact \* $i`

        i=`expr $i + 1`
done
echo "Factorial of $num is $fact"
```

**15) to check two file contents are same. ( Hint: use *cmp or diff* command )**

```
#!/bin/bash

file1='hello.txt'
file2='Hellojake.txt'

if cmp -s "$file1" "$file2"; then
    printf 'The files "%s" and "%s" are same\n' "$file1" "$file2"
else
    printf 'The files "%s" and "%s" are different.\n' "$file1" "$file2"
fi
```

**16) to print a pyramid of numbers.**

```
#!/bin/bash

echo "Enter num of Rows:"
read rows
number=1
for((i=1; i<=rows; i++))
do
  for((j=1; j<=i; j++))
  do
    echo -n "$number "
    number=$((number + 1))
  done
  number=1
  echo
done
```

**17) To write the following contents in a file:- cat,dog,bear,hello,elephant,hello,tiger,hello,horse and then delete the lines containing the word "hello" and display the file. (Hint: use *grep* command)**

```
#1/bin/bash

echo "Please enter File name:"
read fileName

echo "cat" >> "$fileName"
echo "dog" >> "$fileName"
echo "bear" >> "$fileName"
echo "hello" >> "$fileName"
echo "elephant" >> "$fileName"
echo "hello" >> "$fileName"
echo "tiger" >> "$fileName"
echo "hello" >> "$fileName"
echo "horse" >> "$fileName"

sed -i "hello" "$fileName"
cat "$fileName
```