Linux shell scripting Hands on assignment

Write a shell script to change the values in a file(i.e sig.conf) according to the input passed to the script. The script should ask for all four inputs from the user & also validate the input.

Below are the details of input. In full bracket options are given, you have to restrict the user pass single value for each input from the provided options in the full bracket.

Input:-

1) Component Name **[INGESTOR/JOINER/WRANGLER/VALIDATOR]**

2) Scale **[MID/HIGH/LOW]**

3) View **[Auction/Bid]**

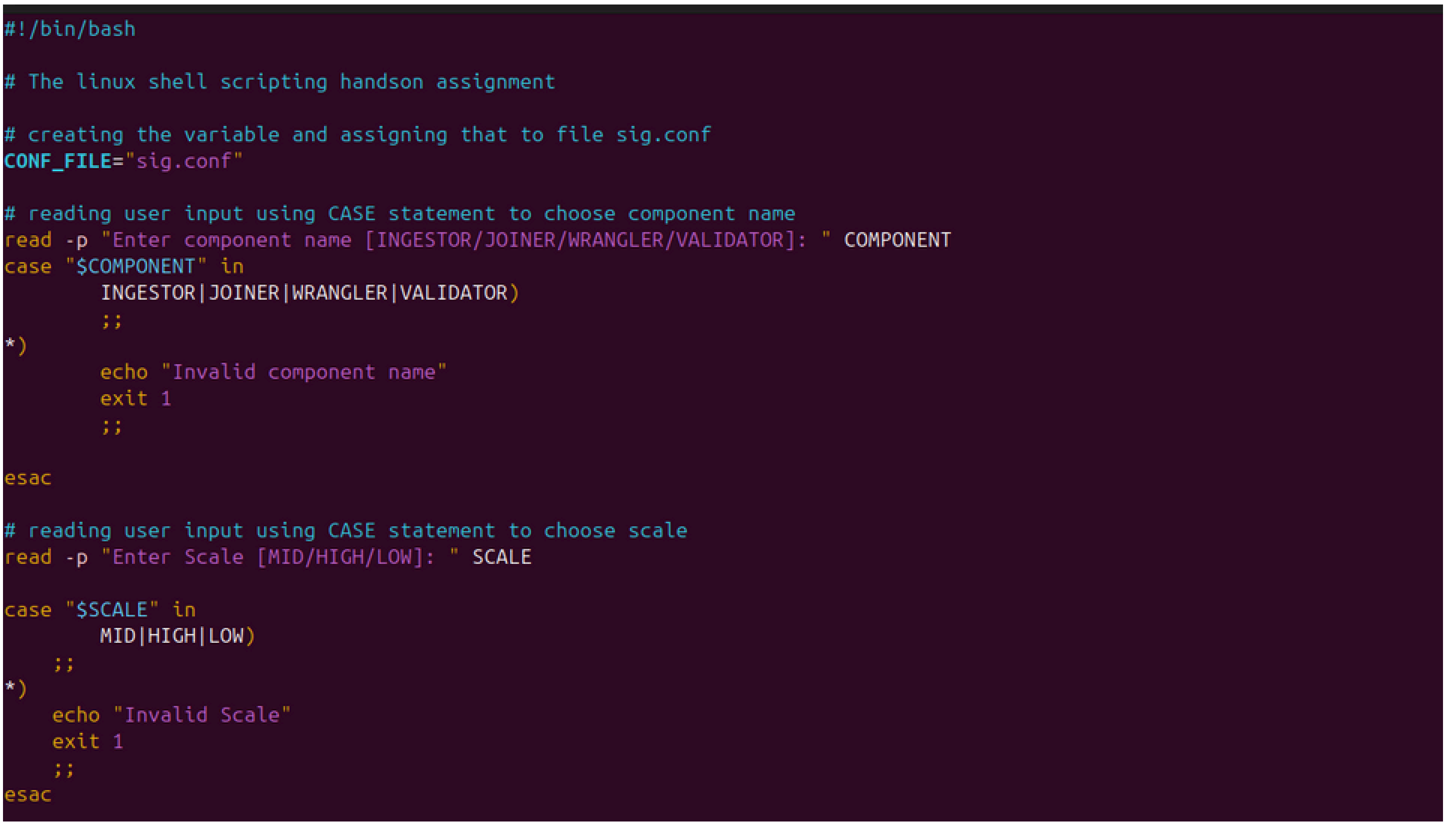
4) Count **[single digit number]**

Explanation of a conf file line.

<view> ; <scale> ; <component name> ; ETL ; vdopia-etl= <count>

Note:- vdopiasample stands for Auction & vdopiasample-bid is for Bid

The script should change the values in the file according to the input provided. At a time only one line of the conf file should be altered.



```
#!/bin/bash

# The linux shell scripting handson assignment

# creating the variable and assigning that to file sig.conf
CONF_FILE="sig.conf"

# reading user input using CASE statement to choose component name
read -p "Enter component name [INGESTOR/JOINER/WRANGLER/VALIDATOR]: " COMPONENT
case "$COMPONENT" in
        INGESTOR|JOINER|WRANGLER|VALIDATOR)
        ;;
*)
        echo "Invalid component name"
        exit 1
        ;;

esac

# reading user input using CASE statement to choose scale
read -p "Enter Scale [MID/HIGH/LOW]: " SCALE

case "$SCALE" in
        MID|HIGH|LOW)
    ;;
*)
    echo "Invalid Scale"
    exit 1
    ;;
esac
```

Creating a variable CONF_FILE and assign that to file sig.conf.

Creating three CASE statement for taking choice of component , scale and view from user .

Taking int input of count which ranges from 0-9.

Creating new variable NEW_LINE to store all the inputs in certain format.

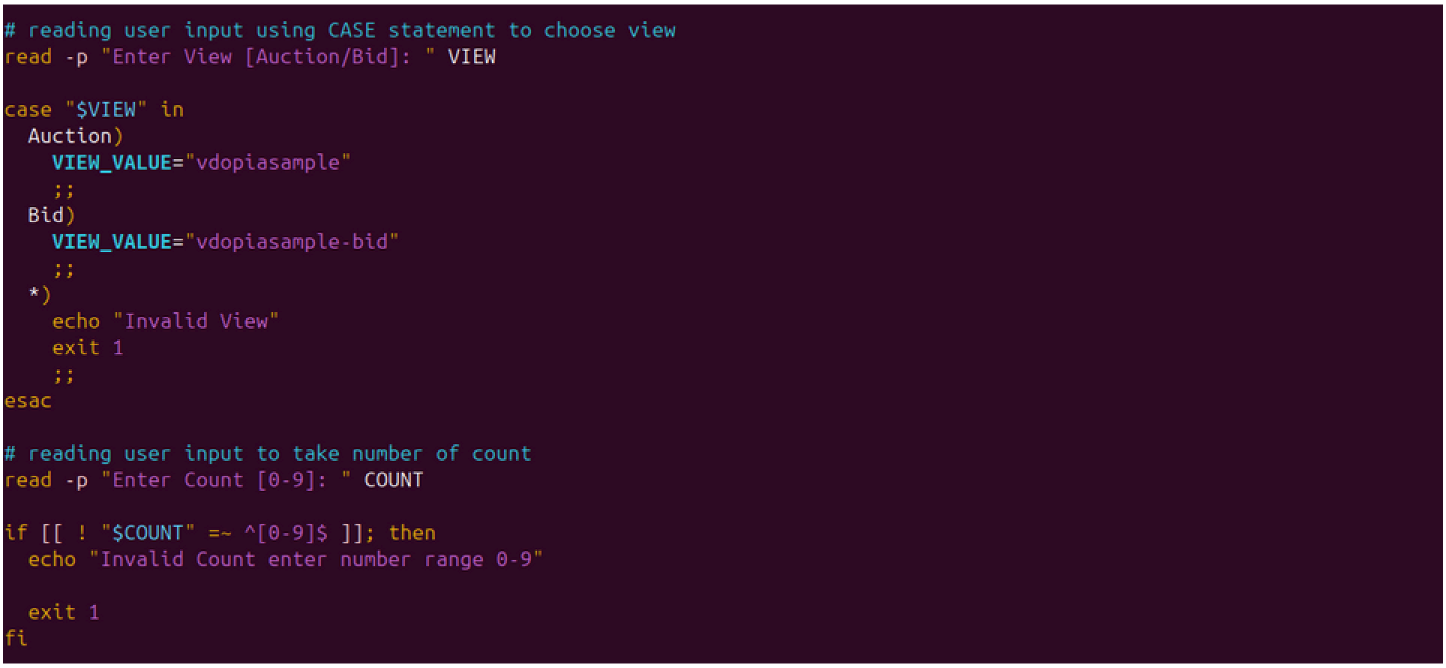
Copy that and create backup

using awk to scan the sig.conf file line by line and find the first line that exactly matches the given view, scale, and component values provided by the user once the first matching line is found, it replaces that line with the new configuration and prints all remaining lines without any further changes.

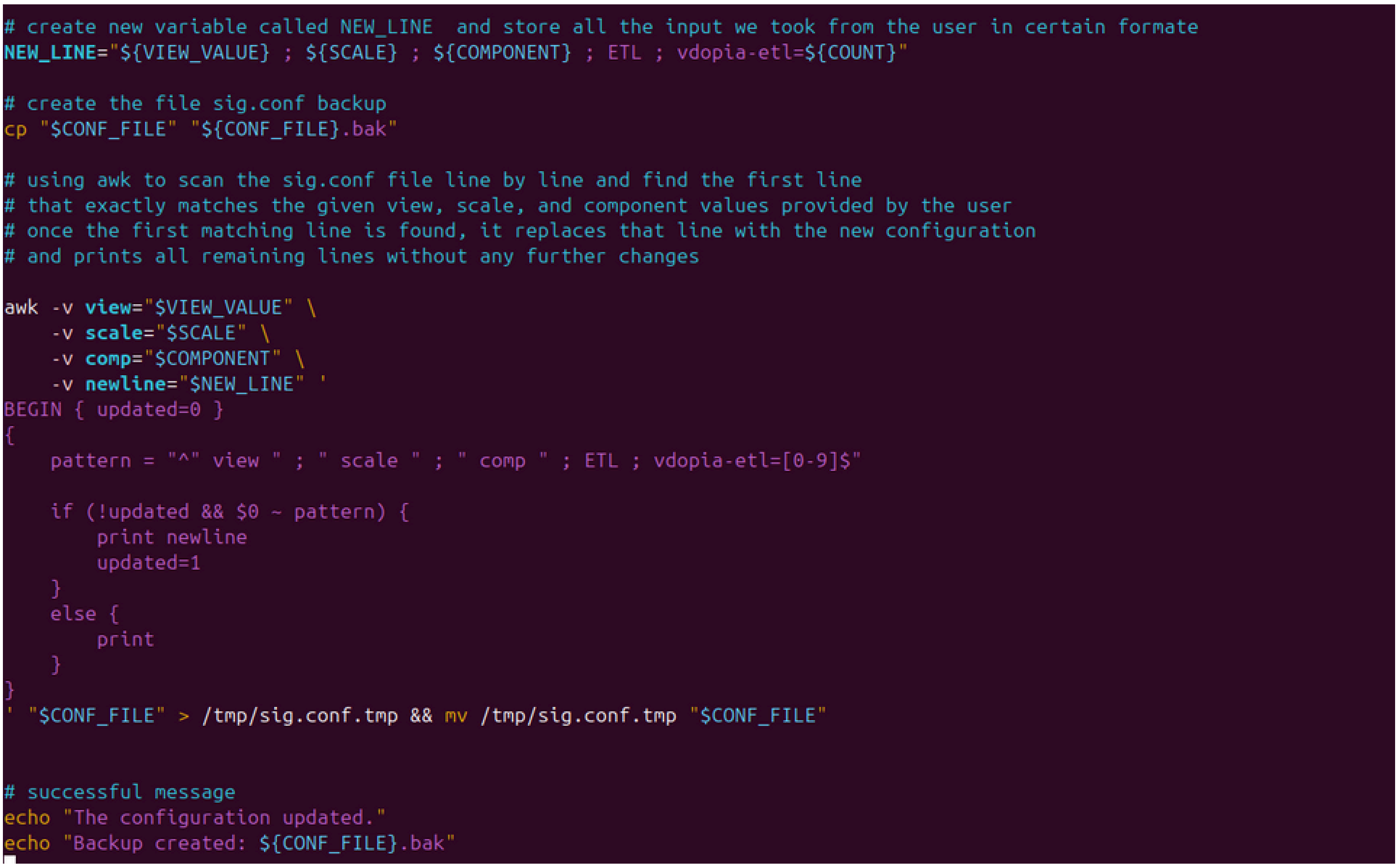
Also tried to use sed command i so some drawbacks like its changing multiple lines and deleting some lines so i shift to AWK

The script treats component, scale, and view as identifiers for a config entry. when it find exact match it will replace it and make sure that only one line change no multiple file changes.

Component, view, and scale together uniquely identify a configuration entry. Changing them would mean modifying the identity of the service itself, not just its configuration value.

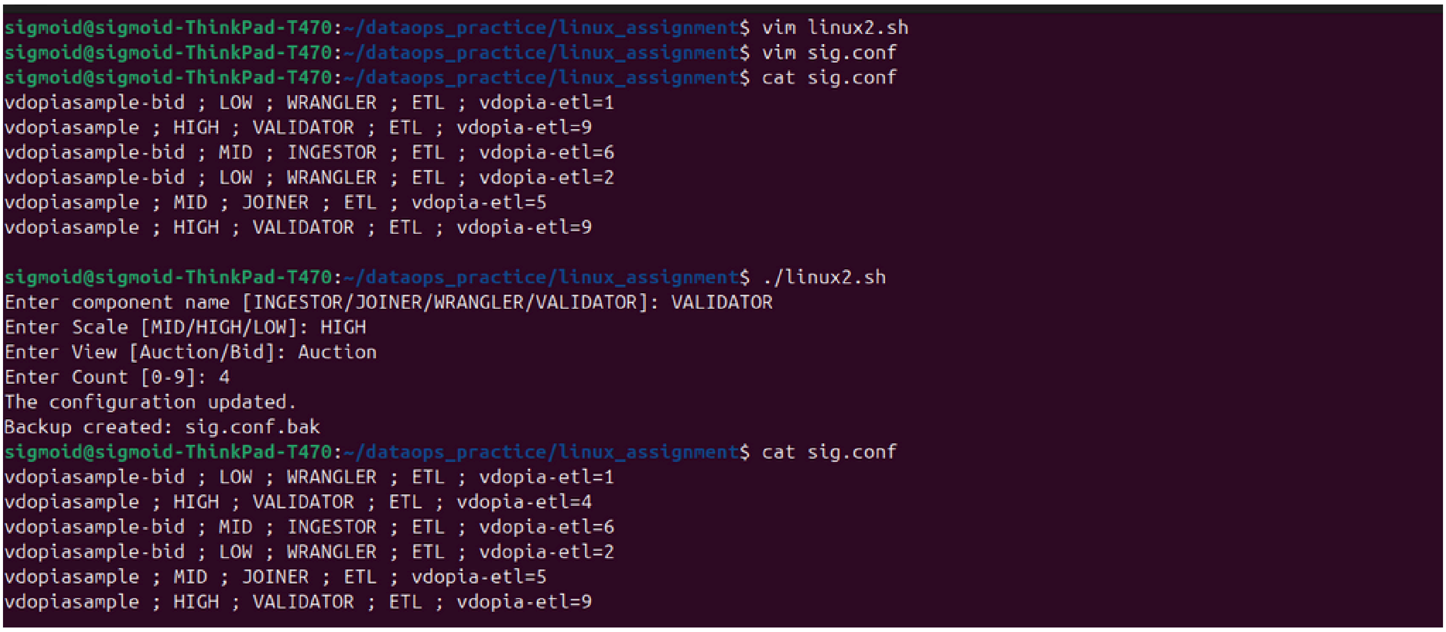


```
# reading user input using CASE statement to choose view
read -p "Enter View [Auction/Bid]: " VIEW

case "$VIEW" in
  Auction)
    VIEW_VALUE="vdopiasample"
    ;;
  Bid)
    VIEW_VALUE="vdopiasample-bid"
    ;;
  *)
    echo "Invalid View"
    exit 1
    ;;
esac

# reading user input to take number of count
read -p "Enter Count [0-9]: " COUNT

if [[ ! "$COUNT" =~ ^[0-9]$ ]]; then
  echo "Invalid Count enter number range 0-9"

  exit 1
fi
```



```
# create new variable called NEW_LINE  and store all the input we took from the user in certain formate
NEW_LINE="${VIEW_VALUE} ; ${SCALE} ; ${COMPONENT} ; ETL ; vdopia-etl=${COUNT}"

# create the file sig.conf backup
cp "$CONF_FILE" "${CONF_FILE}.bak"

# using awk to scan the sig.conf file line by line and find the first line
# that exactly matches the given view, scale, and component values provided by the user
# once the first matching line is found, it replaces that line with the new configuration
# and prints all remaining lines without any further changes

awk -v view="$VIEW_VALUE" \
    -v scale="$SCALE" \
    -v comp="$COMPONENT" \
    -v newline="$NEW_LINE" '
BEGIN { updated=0 }
{
    pattern = "^" view " ; " scale " ; " comp " ; ETL ; vdopia-etl=[0-9]$"

    if (!updated && $0 ~ pattern) {
        print newline
        updated=1
    }
    else {
        print
    }
}
' "$CONF_FILE" > /tmp/sig.conf.tmp && mv /tmp/sig.conf.tmp "$CONF_FILE"


# successful message
echo "The configuration updated."
echo "Backup created: ${CONF_FILE}.bak"
```

Result



```
sigmoid@sigmoid-ThinkPad-T470:~/dataops_practice/linux_assignment$ vim linux2.sh
sigmoid@sigmoid-ThinkPad-T470:~/dataops_practice/linux_assignment$ vim sig.conf
sigmoid@sigmoid-ThinkPad-T470:~/dataops_practice/linux_assignment$ cat sig.conf
vdopiasample-bid ; LOW ; WRANGLER ; ETL ; vdopia-etl=1
vdopiasample ; HIGH ; VALIDATOR ; ETL ; vdopia-etl=9
vdopiasample-bid ; MID ; INGESTOR ; ETL ; vdopia-etl=6
vdopiasample-bid ; LOW ; WRANGLER ; ETL ; vdopia-etl=2
vdopiasample ; MID ; JOINER ; ETL ; vdopia-etl=5
vdopiasample ; HIGH ; VALIDATOR ; ETL ; vdopia-etl=9

sigmoid@sigmoid-ThinkPad-T470:~/dataops_practice/linux_assignment$ ./linux2.sh
Enter component name [INGESTOR/JOINER/WRANGLER/VALIDATOR]: VALIDATOR
Enter Scale [MID/HIGH/LOW]: HIGH
Enter View [Auction/Bid]: Auction
Enter Count [0-9]: 4
The configuration updated.
Backup created: sig.conf.bak
sigmoid@sigmoid-ThinkPad-T470:~/dataops_practice/linux_assignment$ cat sig.conf
vdopiasample-bid ; LOW ; WRANGLER ; ETL ; vdopia-etl=1
vdopiasample ; HIGH ; VALIDATOR ; ETL ; vdopia-etl=4
vdopiasample-bid ; MID ; INGESTOR ; ETL ; vdopia-etl=6
vdopiasample-bid ; LOW ; WRANGLER ; ETL ; vdopia-etl=2
vdopiasample ; MID ; JOINER ; ETL ; vdopia-etl=5
vdopiasample ; HIGH ; VALIDATOR ; ETL ; vdopia-etl=9
```

Verify the input i want to change the line which having files VALIDATOR,HIGH,Auction and change the count to 4

Even though we have same matching field at line 2 and line 6 it will modify only one line it will ignore rest.