1.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>

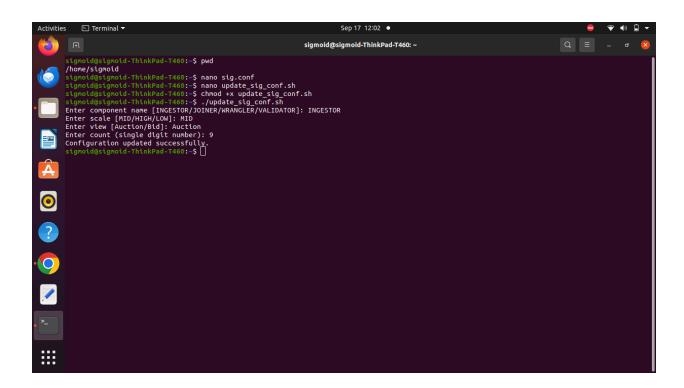
Note:- vdopiasample stands for Auction & vdopiasample-bid is for Bid

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
# File to be updated
CONF_FILE="sig.conf"
# Function to validate component name
validate component() {
  local component=$1
  case $component in
    INGESTOR|JOINER|WRANGLER|VALIDATOR)
       return 0
       ;;
       echo "Invalid component name. Please choose from [INGESTOR, JOINER,
WRANGLER, VALIDATOR]."
       return 1
  esac
}
# Function to validate scale
validate scale() {
  local scale=$1
  case $scale in
    MID|HIGH|LOW)
       return 0
       echo "Invalid scale. Please choose from [MID, HIGH, LOW]."
```

```
return 1
  esac
# Function to validate view
validate_view() {
  local view=$1
  case $view in
     Auction|Bid)
       return 0
       ;;
       echo "Invalid view. Please choose from [Auction, Bid]."
       return 1
  esac
}
# Function to validate count (single digit number)
validate_count() {
  local count=$1
  if [[ count = ^[0-9] ]]; then
     return 0
  else
     echo "Invalid count. Please enter a single digit number (0-9)."
  fi
}
# Prompt user for inputs and validate them
get_inputs() {
  while true; do
     read -p "Enter component name [INGESTOR/JOINER/WRANGLER/VALIDATOR]: "
component
     validate_component $component && break
  done
  while true; do
     read -p "Enter scale [MID/HIGH/LOW]: " scale
     validate_scale $scale && break
  done
  while true; do
```

```
read -p "Enter view [Auction/Bid]: " view
     validate_view $view && break
  done
  while true; do
     read -p "Enter count (single digit number): " count
     validate count $count && break
  done
}
# Update conf file
update_conf_file() {
  local component=$1
  local scale=$2
  local view=$3
  local count=$4
  # Determine the view key
  local view_key=""
  if [[ $view == "Auction" ]]; then
     view key="vdopiasample"
  elif [[ $view == "Bid" ]]; then
     view key="vdopiasample-bid"
  fi
  # Construct the new line
  local new_line="$view; $scale; $component; ETL; $view_key= $count"
  # Find and replace the appropriate line in the conf file
  # Matching lines containing the component, scale, and view
  sed -i "/$view.*$scale.*$component/c\\$new line" $CONF FILE
  echo "Configuration updated successfully."
}
# Main script starts here
if [[!-f $CONF FILE]]; then
  echo "Configuration file $CONF_FILE not found!"
  exit 1
fl
# Get inputs from user
get inputs
# Update the conf file with the provided inputs
update_conf_file $component $scale $view $count
```



The configuration file where only a single line was modified:-

