

Question:

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 to pass a single value for each input from the provided options.

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.

Shell Script:

```
#!/bin/bash
```

```
CONF_FILE="sig.conf"
```

```
# Function to validate input
```

```
validate_input() {  
    local input="$1"  
    shift  
    local options=("$@")  
    for opt in "${options[@]}; do  
        if [[ "$input" == "$opt" ]]; then  
            return 0  
        fi  
    done  
    return 1  
}
```

```
# Prompt and validate Component Name
```

```
while true; do  
    read -p "Enter Component Name [INGESTOR/JOINER/WRANGLER/VALIDATOR]: " component  
    if validate_input "$component" "INGESTOR" "JOINER" "WRANGLER" "VALIDATOR"; then  
        break  
    else  
        echo "Invalid component. Choose from [INGESTOR/JOINER/WRANGLER/VALIDATOR]"  
    fi  
done
```

```
# Prompt and validate Scale
```

```

while true; do
    read -p "Enter Scale [MID/HIGH/LOW]: " scale
    if validate_input "$scale" "MID" "HIGH" "LOW"; then
        break
    else
        echo "Invalid scale. Choose from [MID/HIGH/LOW]"
    fi
done

# Prompt and validate View
while true; do
    read -p "Enter View [Auction/Bid]: " view
    if validate_input "$view" "Auction" "Bid"; then
        break
    else
        echo "Invalid view. Choose from [Auction/Bid]"
    fi
done

# Prompt and validate Count
while true; do
    read -p "Enter Count [0-9]: " count
    if [[ "$count" =~ ^[0-9]$ ]]; then
        break
    else
        echo "Invalid count. Enter a single digit [0-9]."
    fi
done

# Map view to internal config name
if [[ "$view" == "Auction" ]]; then
    conf_view="vdopiasample"
else
    conf_view="vdopiasample-bid"
fi

# Prepare the new config line
newline="$view ; $scale ; $component ; ETL ; $conf_view-etl= $count"

# Temporary file for safe editing
tmp_file=$(mktemp)
found=0

# Read original config and write to temp with one replacement
while IFS= read -r line; do
    if [[ $found -eq 0 && "$line" == *"$view"* && "$line" == *"$scale"* && "$line" == *"$component" ]]; then
        echo "$newline" >> "$tmp_file"
        found=1
    else
        echo "$line" >> "$tmp_file"
    fi
done < "$CONF_FILE"

# If no match was found, append the new line

```

```
if [[ $found -eq 0 ]]; then
    echo "$newline" >> "$tmp_file"
fi

# Replace the original config file with the modified one
mv "$tmp_file" "$CONF_FILE"

echo "Configuration updated successfully in $CONF_FILE."
```