

IP Validation Script

IP Validation Shell Script-

```
ipValidation (){  
    echo "Please enter your IP:"  
  
    read ip  
    if echo "$ip" | { IFS=. read a b c d e;  
        test "$a" -ge 0 && test "$a" -le 255 &&  
        test "$b" -ge 0 && test "$b" -le 255 &&  
        test "$c" -ge 0 && test "$c" -le 255 &&  
        test "$d" -ge 0 && test "$d" -le 255 &&  
        test -z "$e"; }; then echo ip is valid;  
    else  
        echo "ip is not valid"  
    fi  
}  
  
while true  
do
```

```
ipValidation  
  
read -p " Press any key to continue, or type 'q' to quit: " input  
  
if [ "$input" == "q" ]  
  
then  
  
    break  
  
fi  
  
done
```

Explanation;

The provided script is a Bash script that validates an IP address entered by the user. Here's a breakdown of how it works:

1. The script defines a function named **`ipValidation`** that prompts the user to enter an IP address.
2. The user's input is read and stored in the **`ip`** variable.
3. The script uses a combination of **`echo`**, **`read`**, and **`test`** commands to validate the IP address. It checks if each octet (separated by periods) falls within the valid range of 0 to 255. If any octet is outside this range or if there is a fifth component (e.g., an extra period), the IP address is considered invalid.
4. If the IP address is valid, the script displays the message **"ip is valid."** Otherwise, it displays the message **"ip is not valid."**
5. The script then enters a **`while true`** loop, repeatedly calling the **`ipValidation`** function.
6. After each validation, the script prompts the user to press any key to continue or type **'q'** to quit.

7. If the user enters 'q', the loop breaks, and the script terminates.

By running this script, you can validate IP addresses by entering them interactively. The script will indicate whether the entered IP address is valid or not. You can continue validating IP addresses until you choose to quit by entering 'q' at the prompt.