

Array ??

Learning

- [Bash Scripting – Indexed Array Explained With Examples](#)
- [Bash Scripting – Associative Array Explained With Examples](#)

List Array

- ?????? `${#str_list[@]}`
- ?????? `${str_list[@]}` ? `${str_list[*]}`

```
#!/bin/bash
str_list=("aaa" "bbb" "ccc" "ddd")
echo ${#str_list[@]}
for i in ${str_list[@]}
do
    echo "$i"
done
```

```
# Script to split a string based on the delimiter
my_string="Ubuntu;Linux Mint;Debian;Arch;Fedora"
IFS=';' read -ra my_array <<< "$my_string"

#Print the split string
for i in "${my_array[@]}"
do
    echo $i
done
```

```
# Script to split a string based on the delimiter
my_string="Ubuntu;Linux Mint;Debian;Arch;Fedora"
my_array=$(echo $my_string | tr ";" "\n")

#Print the split string
for i in "${my_array[@]}"
```

```
do
    echo $i
done
```

```
local n=0

until [[ $n -eq ${#asr_array[@]} ]]; do
    asr=${asr_array[$n]}
    echo $asr
    let n++
done
```

? printf ?????????? grep ??????????

```
# --- Ban active IPs not already banned ---
for ip in $active_ips; do
    if ! printf '%s\n' "${banned_ips[@]}" | grep -qw "$ip"; then
        log "Banning IP via Fail2Ban: $ip"
        fail2ban-client set "$JAIL" banip "$ip"
    fi
done
```

???????

```
# Delete the 3th element of the array
unset str_list[3]
```

String to Array

```
while read c1 c2 c3 c4 c5 c6;do
    day="$c1"
    ncalls="$c4"
    scalls="$c6"
    asr=$(echo "scale=2; $scalls * 100 / $ncalls" | bc)
    day_str+="$day "
    asr_str+="$asr "
    calls_str+="$ncalls "
done < $tmpfile
```

```
read -ra day_array <<< "$day_str"
read -ra asr_array <<< "$asr_str"
read -ra calls_array <<< "$calls_str"
```

mapfile

????????????????????

????

- ?? lines ?????
- `mapfile lines < file.txt`
- `mapfile -t lines < <(command)`

```
# File
mapfile MYFILE < example.txt
echo ${MYFILE[@]}
echo ${MYFILE[0]}

# Command
mapfile -t GEEKSFORGEEKS < <(printf "Item 1\nItem 2\nItem 3\n")
echo ${GEEKSFORGEEKS[@]}
```

????? ufw ??? IP

```
# --- Remove UFW rules for inactive IPs ---
for ip in $inactive_ips; do
    # Get UFW rules for this IP in reverse order to avoid shifting rule numbers on deletion
    mapfile -t rules < <(ufw status numbered | grep "$ip" | grep "DENY IN" | tac)
    for rule in "${rules[@]"; do
        rule_number=$(echo "$rule" | awk -F'[][]' '{print $2}')
        log "Removing UFW rule #$rule_number for IP: $ip"
        ufw --force delete "$rule_number"
    done
done
```