Bash Scripting: Loops with Cron Jobs and Commands (50 Examples)

Bash scripting is widely used for automating tasks in Linux. Loops in Bash allow us to execute a set of commands repeatedly. Bash provides three types of loops:

1. **For Loop**
2. **While Loop**
3. **Until Loop**

Cron jobs are scheduled tasks that run automatically at predefined times. Let's explore various Bash scripts using loops and integrate them with cron jobs.

---

1. FOR LOOP EXAMPLES

1.1 Print Numbers from 1 to 10

bash

CopyEdit

```
#!/bin/bash
for i in {1..10}
do
  echo "Number: $i"
done
```

1.2 Loop with Step Value

bash

CopyEdit

```
#!/bin/bash
for i in {1..10..2}
do
  echo "Odd Number: $i"
done
```

1.3 Loop Using C-Style Syntax

bash

CopyEdit

```
#!/bin/bash
for ((i=1; i<=5; i++))
do
  echo "Iteration: $i"
done
```

1.4 Loop Over an Array

bash

CopyEdit

```
#!/bin/bash
arr=("Ubuntu" "Debian" "CentOS" "Fedora")
for os in "${arr[@]}"
do
  echo "OS: $os"
done
```

1.5 Loop Through Files in a Directory

bash

CopyEdit

```
#!/bin/bash
for file in /home/user/*.txt
do
  echo "File: $file"
done
```

1.6 Loop Through Command Output

bash

CopyEdit

```
#!/bin/bash
for user in $(cut -d: -f1 /etc/passwd)
do
  echo "User: $user"
done
```

1.7 Using break in a For Loop

bash

CopyEdit

```
#!/bin/bash
for i in {1..10}
do
  if [ $i -eq 5 ]; then
    break
  fi
  echo "Count: $i"
done
```

1.8 Using continue in a For Loop

bash

CopyEdit

```
#!/bin/bash
for i in {1..5}
do
  if [ $i -eq 3 ]; then
    continue
  fi
  echo "Number: $i"
done
```

---

2. WHILE LOOP EXAMPLES

2.1 Print Numbers from 1 to 5

bash

CopyEdit

```
#!/bin/bash
i=1
while [ $i -le 5 ]
do
  echo "Count: $i"
  ((i++))
done
```

2.2 Read File Line by Line

bash

CopyEdit

```
#!/bin/bash
while read line
do
  echo "$line"
done < /etc/passwd
```

2.3 Infinite While Loop

bash

CopyEdit

```
#!/bin/bash
while true
do
  echo "Press CTRL+C to stop..."
  sleep 2
done
```

2.4 Read User Input Until Quit

bash

CopyEdit

```
#!/bin/bash
input=""
while [ "$input" != "quit" ]
do
  read -p "Enter something (type 'quit' to exit): " input
done
```

2.5 Using break in While Loop

bash

CopyEdit

```
#!/bin/bash
i=1
while [ $i -le 10 ]
do
  echo "Count: $i"
  ((i++))
  if [ $i -eq 5 ]; then
    break
  fi
done
```

2.6 Using continue in While Loop

bash

CopyEdit

```
#!/bin/bash
i=0
while [ $i -lt 5 ]
do
  ((i++))
  if [ $i -eq 3 ]; then
    continue
  fi
  echo "Number: $i"
done
```

---

3. UNTIL LOOP EXAMPLES

3.1 Basic Until Loop

bash

CopyEdit

```
#!/bin/bash
i=1
until [ $i -gt 5 ]
do
  echo "Count: $i"
  ((i++))
```

```
done
```

3.2 Until Loop With User Input

bash

CopyEdit

```
#!/bin/bash
password=""
until [ "$password" == "12345" ]
do
  read -s -p "Enter Password: " password
  echo
done
echo "Access Granted"
```

---

4. CRON JOB EXAMPLES

4.1 Schedule a Script Every Minute

Edit the crontab using:

bash

CopyEdit

```
crontab -e
```

Add:

arduino

CopyEdit

```
* * * * * /home/user/script.sh
```

4.2 Run Script Every 5 Minutes

ruby

CopyEdit

```
*/5 * * * * /home/user/script.sh
```

4.3 Run Script at Midnight

arduino

CopyEdit

```
0 0 * * * /home/user/midnight_task.sh
```

4.4 Backup Files Every Day at 2 AM

arduino

CopyEdit

```
0 2 * * * tar -czf /backup/home.tar.gz /home/user
```

---

5. COMMANDS IN LOOP

5.1 Ping Multiple Hosts

bash

CopyEdit

```
#!/bin/bash
hosts=("google.com" "yahoo.com" "bing.com")
for h in "${hosts[@]}"
do
  ping -c 2 $h
done
```

5.2 Check If Services Are Running

bash

CopyEdit

```
#!/bin/bash
services=("nginx" "mysql" "apache2")
for service in "${services[@]}"
```

```
do
  systemctl is-active --quiet $service && echo "$service is running" || echo "$service is stopped"
done
```

5.3 Delete Old Log Files

bash

CopyEdit

```
#!/bin/bash
find /var/log -type f -name "*.log" -mtime +7 -exec rm {} \;
```

5.4 Monitor Disk Usage

bash

CopyEdit

```
#!/bin/bash
while true
do
  df -h
  sleep 60
done
```

5.5 Restart a Service If It Fails

bash

CopyEdit

```
#!/bin/bash
while true
do
  systemctl is-active --quiet apache2 || systemctl restart apache2
  sleep 60
done
```

SUMMARY

- **For Loop:** Used for iterating over a range, array, or command output.
- **While Loop:** Runs while a condition is true.
- **Until Loop:** Runs until a condition becomes true.
- **Cron Jobs:** Schedule scripts to run at specific times.
- **Commands in Loops:** Automate system administration tasks.

Would you like me to provide more advanced automation examples? 🚀