Until loop

- **What is until**
- **Use cases and syntax**

What is until loop

The **until** loop is used to execute a given set of commands as long as the given condition evaluates to false.

The Bash **until** loop takes the following form:

```
until [CONDITION]
do
  [COMMANDS]
done
```

The condition is evaluated before executing the commands. If the condition evaluates to false, commands are executed. Otherwise, if the condition evaluates to true the loop will be terminated and the program control will be passed to the command that follows.

Until sample

In the example below, on each iteration the loop prints the current value of the variable counter and Increments the variable by one.

```bash
#!/bin/bash

counter=0

until [ $counter -gt 5 ]
do
  echo Counter: $counter
  ((counter++))
done
```

The loop iterates as long as the counter variable has a value greater than four. The script will produce the following output:

```
[root@localhost tmp]# bash until_sample.sh
Counter: 0
Counter: 1
Counter: 2
Counter: 3
Counter: 4
Counter: 5
[root@localhost tmp]#
```

Until examples

The following script may be useful when your git host has downtime, and instead of manually typing **git pull** multiple times until the host is online, you can run the script once. It will try to pull the repository until it is successful.

```bash
#!/bin/bash

until git pull &> /dev/null
do
    echo "Waiting for the git host ..."
    sleep 1
done

echo -e "\nThe git repository is pulled."
```

The script will print "Waiting for the git host …" and sleep for one second until the git host goes online. Once the repository is pulled, it will print "The git repository is pulled.".

```
[root@localhost tmp]# bash until_git_pull.sh
Waiting for the git host ...
Waiting for the git host ...

The git repository is pulled.

[root@localhost tmp]# █
```

Until examples

An improved script from **While** video, which tests for available disk space.

If not enough disk space is available, remove pictures from the previous months:

```bash
#!/bin/bash

while true; do
    DISKFUL=$(df -h $WEBDIR | grep -v File | awk '{print $5 }' | cut -d "%" -f1 -)

    until [ $DISKFUL -ge "90" ]; do

            DATE=`date +%Y%m%d`
            HOUR=`date +%H`
            mkdir $WEBDIR/"$DATE"

            while [ $HOUR -ne "00" ]; do
                    DESTDIR=$WEBDIR/"$DATE"/"$HOUR"
                    mkdir "$DESTDIR"
                    mv $PICDIR/*.jpg "$DESTDIR"/
                    sleep 3600
                    HOUR=`date +%H`
            done

    DISKFULL=$(df -h $WEBDIR | grep -v File | awk '{ print $5 }' | cut -d "%" -f1 -)
    done

    TOREMOVE=$(find $WEBDIR -type d -a -mtime +30)
    for i in $TOREMOVE; do
        rm -rf "$i";
    done
done
```

Thanks for watching!