

# The until loop

## Synopsis

```
until <LIST1> ; do
    <LIST2>
done
```

## Description

The until-loop is relatively simple in what it does: it executes the command list `<LIST1>` and if the exit code of it was **not** 0 (FALSE) it executes `<LIST2>` . This happens again and again until `<LIST1>` returns TRUE.

This is exactly the opposite of the while loop.

⚠ Like all loops (both `for` -loops, `while` and `until` ), this loop can be

- terminated (broken) by the `break` command, optionally as `break N` to break `N` levels of nested loops
- forced to immediately do the next iteration using the `continue` command, optionally as `continue N` analog to `break N`

## Return status

The return status is the one of the last command executed in `<LIST2>` , or 0 ( TRUE ) if none was executed.

## Examples

## Portability considerations

## See also

- Internal: The while loop



## Discussion

[syntax/ccmd/until\\_loop.txt](#) Last modified: 2011/01/02 10:50 by thebonsai

---

This site is supported by Performing Databases - your experts for database administration

---

Bash Hackers Wiki

---



Except where otherwise noted, content on this wiki is licensed under the following license:  
GNU Free Documentation License 1.3