

# The while-loop

## Synopsis

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

## Description

The while-loop is relatively simple in what it does: it executes the command list `<LIST1>` and if the exit code of it was 0 (TRUE) it executes `<LIST2>`. This happens again and again until `<LIST1>` returns FALSE.

This is exactly the opposite of the until 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 until loop

- Internal: code examples of the read builtin command to see how you can loop over lines

## Discussion

---

 [syntax/ccmd/while\\_loop.txt](#)  Last modified: 2010/04/15 18:47 (external edit)

---

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