

**NAME**

**which** - locate a program file in the user's path

**SYNOPSIS**

**which** [-as] *program* ...

**DESCRIPTION**

The **which** utility takes a list of command names and searches the path for each executable file that would be run had these commands actually been invoked.

The following options are available:

- a List all instances of executables found (instead of just the first one of each).
- s No output, just return 0 if all of the executables are found, or 1 if some were not found.

Some shells may provide a builtin **which** command which is similar or identical to this utility. Consult the builtin(1) manual page.

**EXAMPLES**

Locate the ls(1) and cp(1) commands:

```
$ /usr/bin/which ls cp
/bin/ls
/bin/cp
```

Same as above with a specific PATH and showing all occurrences:

```
$ PATH=/bin:/rescue /usr/bin/which -a ls cp
/bin/ls
/rescue/ls
/bin/cp
/rescue/cp
```

**which** will show duplicates if the same executable is found more than once:

```
$ PATH=/bin:/bin /usr/bin/which -a ls
/bin/ls
/bin/ls
```

Do not show output. Just exit with an appropriate return code:

```
$ /usr/bin/which -s ls cp
```

```
$ echo $?
```

```
0
```

```
$ /usr/bin/which -s fakecommand
```

```
$ echo $?
```

```
1
```

## SEE ALSO

builtin(1), csh(1), find(1), locate(1), whereis(1)

## HISTORY

The **which** command first appeared in FreeBSD 2.1.

## AUTHORS

The **which** utility was originally written in Perl and was contributed by Wolfram Schneider <[wosch@FreeBSD.org](mailto:wosch@FreeBSD.org)>. The current version of **which** was rewritten in C by Daniel Papasian <[dpapasia@andrew.cmu.edu](mailto:dpapasia@andrew.cmu.edu)>.