

# What does `${2:-${1}}` mean in Bash?

What does the following bash snippet do exactly? `${2:-${1}}`

12

bash

asked Jul 22 '09 at 5:03



[Ben McCann](#)

10.3k

18

63

88

5

## 4 Answers

"Use the second argument, but if none then the first one".

18

answered Jul 22 '09 at 5:05



[Alex Martelli](#)

648k

132

1057

1288



`${var:-default}` evaluates to the value of `$var`, unless `$var` isn't set in which case it evaluates to the text "default". `$1`, `$2`, et al are the command-line arguments to your program (or function). Putting the two together it means to return `$2` if there were two arguments passed, otherwise return `$1`.

11

answered Jul 22 '09 at 5:09



[John Kugelman](#)

255k

56

415

466

It means "Use the second argument if the first is undefined or empty, else use the first". The form `"${2:-${1}}"` (no `:'`) means "Use the second if the first is not defined (but if the first is defined as empty, use it)".

2

answered Jul 28 '09 at 22:50

[Kevin Little](#)