

A tour of Nix

9 / 35 Functions: attribute sets arguments

[prev](#)[next](#)

When passing an `attribute set` to `function`, you can set default values.

Doing so, allows the function to be called without that parameter, making it optional.

Note: A default value is defined with a '?'.
Example: `foo ? "foo"`

Now:

- Change the `function` 'func' in a way that **foobar** is evaluated to **'foobar'**.

Note: See video [@youtube](#)

```
1 let
2   f = "f";
3   o = "o";
4   b = "b";
5   func = {a ? f, b , c } : a+b+c; #only modify this line!
6 in
7 rec {
8   foo = func {b="o"; c=o;}; #must evaluate to "foo"
9   bar = func {a=b; c="r";}; #must evaluate to "bar"
10  foobar = func {a=foo;b=bar;}; #must evaluate to "foobar"
11 }
12
```

[reset](#)[solution](#)[run](#)

```
let
  f = "f";
  o = "o";
  b = "b";
  func = {a ? f, b ? "a", c ? ""}: a+b+c; #Only modify this line!
in
rec {
  foo = func {b="o"; c=o;}; #should be foo
  bar = func {a=b; c="r";}; #should be bar
  foobar = func {a=foo;b=bar;}; #should be foobar
}
```

