

A tour of Nix

10 / 35 Functions: the @-pattern

prev

next

Ellipsis

Functions can be called with an `attribute set`, as we have seen. This `attribute set` must contain all required 'function arguments'.

However, such an `Attribute set` can contain additional attributes but you have to add `...` like this:

```
func2 = args@{a, b, c, ...}: a+b+c+args.d;
```

Note: `...` is called `ellipsis`.

@-pattern

Inside a function, those `attributes` can be accessed with the `@-pattern`.

Now complete the arguments `attribute set`:

- **foo** must evaluate to 'foo'
- **foobar** must evaluate to 'foobar'

Note: See video [@youtube](#)

```
1 let
2   arguments = {a="f"; b="o"; c=X; d=X;}; #only modify this line
3   func = {a, b, c, ...}: a+b+c;
4   func2 = args@{a, b, c, ...}: a+b+c+args.d;
5 in
6 {
7   #the argument d is not used
8   foo = func arguments;
9   #now the argument d is used
10  foobar = func2 arguments;
11 }
12
```

reset

solution

run

```
let
  arguments = {a="f"; b="o"; c="o"; d="bar";}; #only modify this line

  func = {a, b, c, ...}: a+b+c;
  func2 = args@{a, b, c, ...}: a+b+c+args.d;
in
{
  #the argument d is not used
  foo = func arguments;
  #now the argument d is used
  foobar = func2 arguments;
}
```