

A tour of Nix

34 / 35 Reimplementation: catAttrs

prev

next

Write the function `catAttrs`.

```
catAttrs "a" [{a = 1;} {b = 0;} {a = 2;}]; => [ 1 2 ]
```

Attribute sets that don't contain the named attribute are ignored.

Note: you can use the builtin function `concatLists`.

Note: the right-hand side of the `? operator` has the same syntax as the assignment in attrsets (i.e., `{foo=42;}` and `{"foo" = 42;}`)

Warning: This is hard!

Note: See video [@youtube](#)

```
1 with import <nixpkgs> { };
2 let
3   list = ["a"] ["b"] ["c"];
4
5   attrList = [{a = 1;} {b = 0;} {a = 2;}];
6   catAttrs = XXX
7 in
8 rec {
9   example = builtins.concatLists list; #is [ "a" "b" "c" ]
10  result = catAttrs "a" attrList; #should be [1 2]
11 }
12
```

reset

solution

run

```
with import <nixpkgs> { };
let
  list = ["a"] ["b"] ["c"];

  attrList = [{a = 1;} {b = 0;} {a = 2;}];
  catAttrs = name: List: builtins.concatLists (map (x: if x ? ${name} then [x.${name}]
else []) List);
in
rec {
  example = builtins.concatLists list; #is [ "a" "b" "c" ]
  result = catAttrs "a" attrList; #should be [1 2]
}
```

