Spicy Exercises

Exercises with problem, solution, and explanation.

Time for some exercises! While doing them take a look at Ramda's docs to see what functions can help you.

Remember point-free means your data's not visible.

```
// sum is NOT point-free...
// nums parameter is showing
const sum = (nums) => nums.reduce((x, y) => x + y, 0);

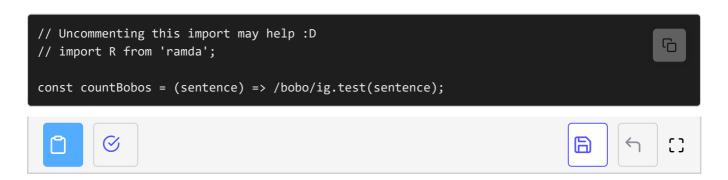
// sum is point-free
import { add, reduce } from 'ramda';
const sum = reduce(add, 0);

// This is okay too,
// but inner function (x, y) => {} isn't point-free
import { reduce } from 'ramda';
const sum = reduce((x, y) => x + y, 0);
```

Exercise 1

This code tells you if a given sentence contains "Bobo", regardless of case. Refactor it to be point-free.

https://ramdajs.com/docs/#test



Exercise 2

This code tells you if someone should consider a tech career. Refactor it to be point-free.

https://ramdajs.com/docs/#ifElse https://ramdajs.com/docs/#where

Exercise 3

This code returns everyone's age. Refactor it to be point-free.

https://ramdajs.com/docs/#map https://ramdajs.com/docs/#prop https://ramdajs.com/docs/#pluck



Exercise 4

This code rejects everyone under 18, and over 25. Refactor it to be point-free.

https://ramdajs.com/docs/#filter https://ramdajs.com/docs/#propSatisfies



Exercise 5

Create a function called defaultTo. It takes two parameters:

- defaultVal: A default value
- val: The value to return

If val is null or undefined, return defaultVal. Else, return val.

Curry it to allow preloading arguments.

