



Common operations on collections

Kotlin library: extensions on collections

```
list.max|
```

```
λ max() for Iterable<T> in kotlin.collections Int  
λ maxBy {...} (selector: (Int) -> R) for Iterable<T> in kot... Int?  
λ maxWith(comparator: Comparator<in Int>) for Iterable<T> i... Int?  
^↓ and ^↑ will move caret down and up in the editor >>>
```

- filter
- map
- reduce
- count
- find
- any
- flatMap
- groupBy
- ...

Working with collections with lambdas

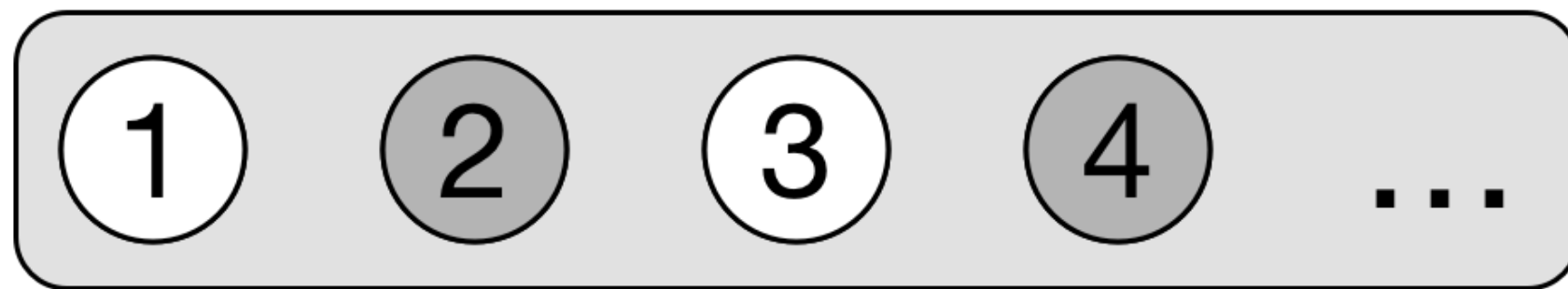
```
val employees: List<Employee>  
data class Employee(  
    val city: City, val age: Int  
)
```

What's an average
age of employees
working in Prague?

```
employees.filter { it.city == City.PRAGUE }.map { it.age }.average()
```

extension functions

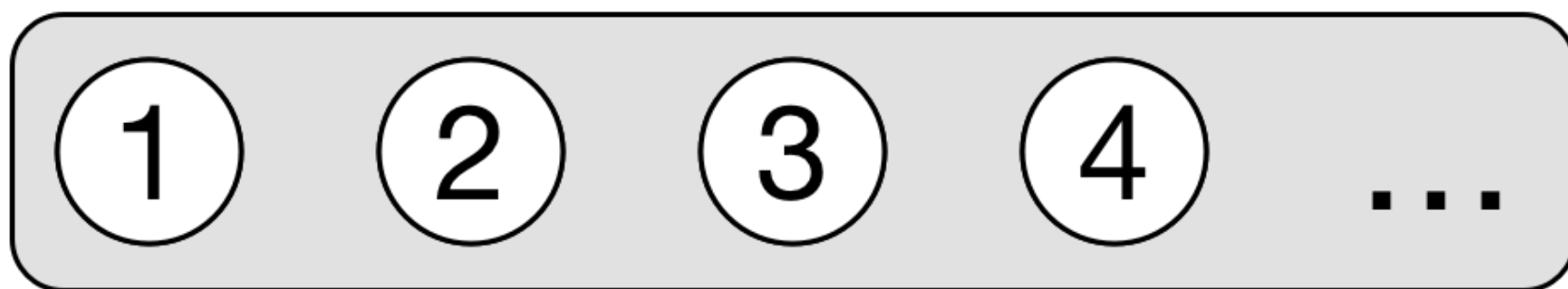
filter



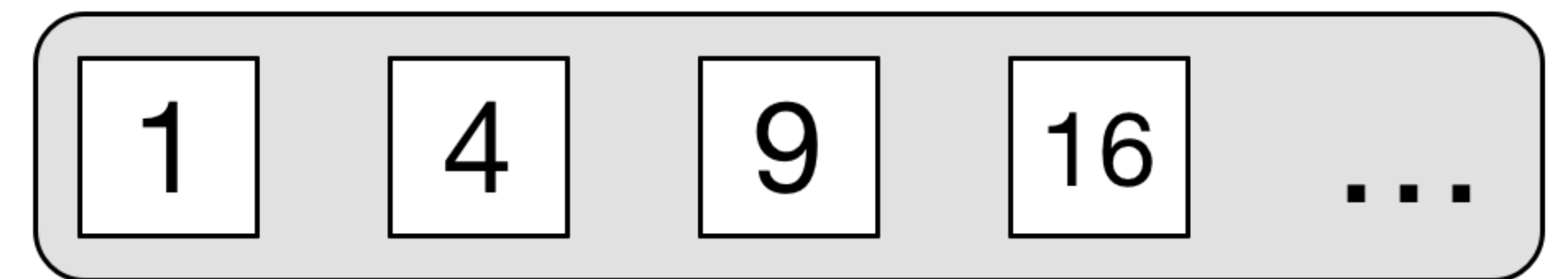
$\{ it \% 2 == 0 \}$
→



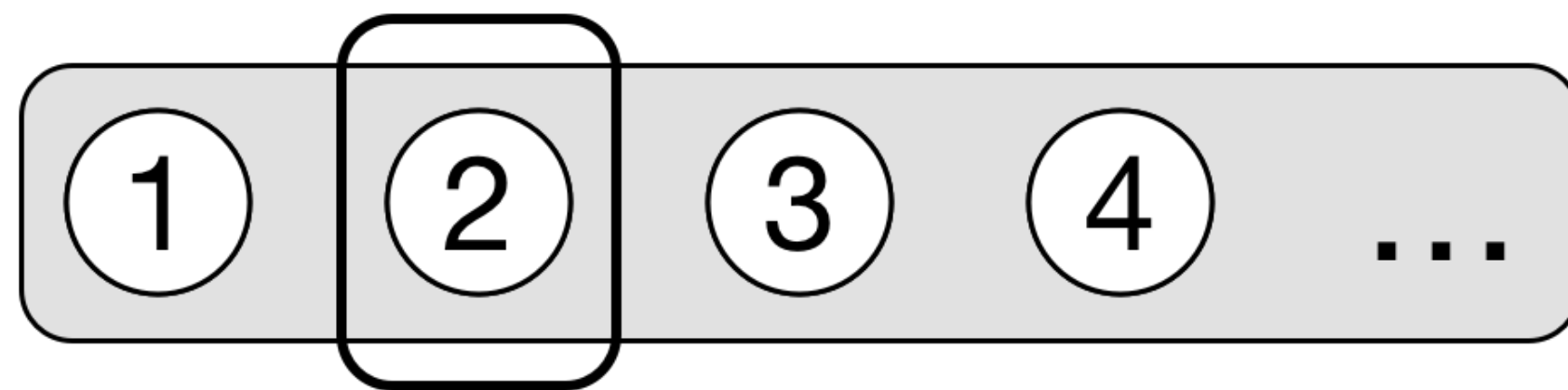
map



$\{ it * it \}$
→



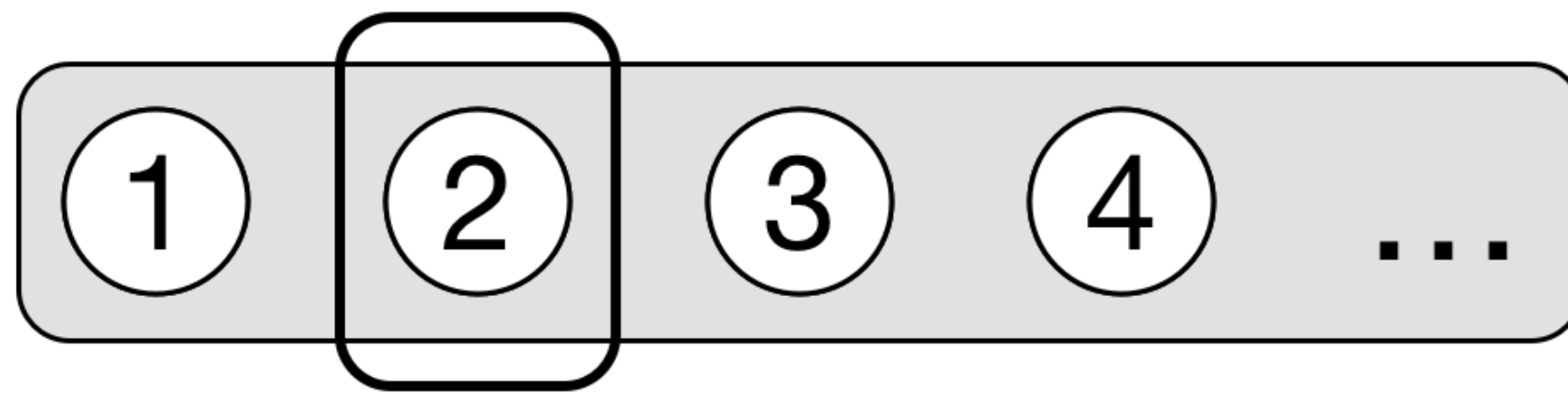
any (all, none)



$\{ it \% 2 == 0 \}$
→

true

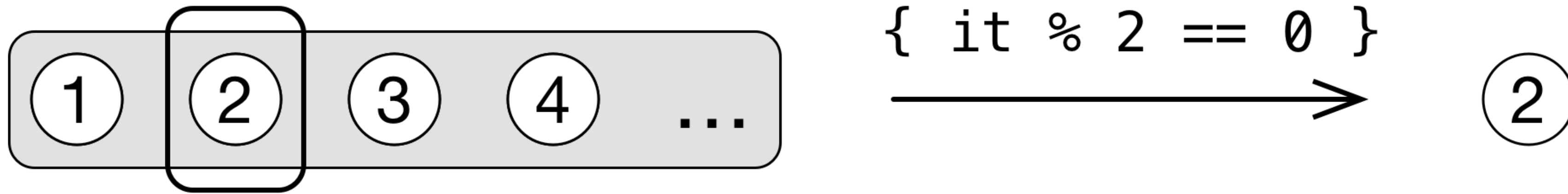
find



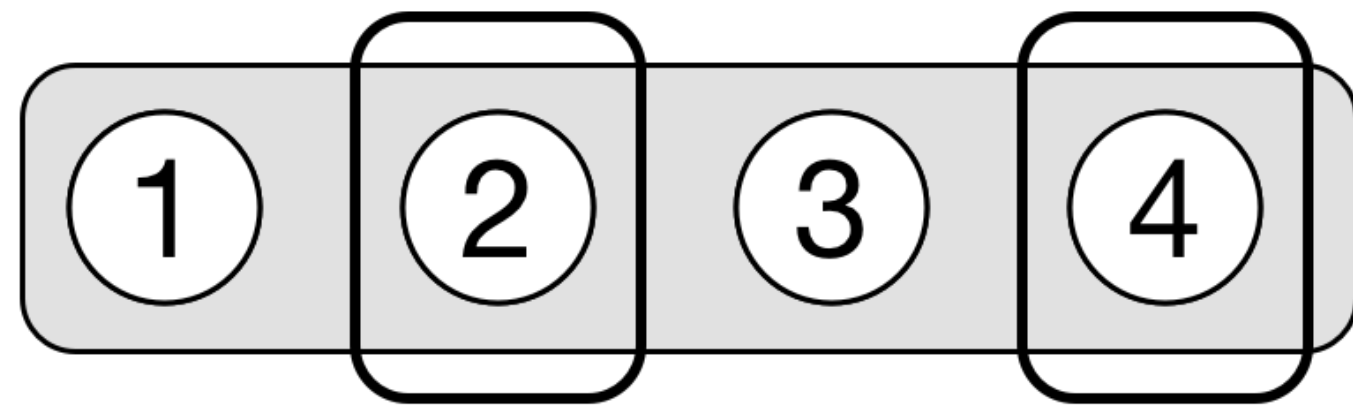
$\{ it \% 2 == 0 \}$
→

2

first / firstOrNull



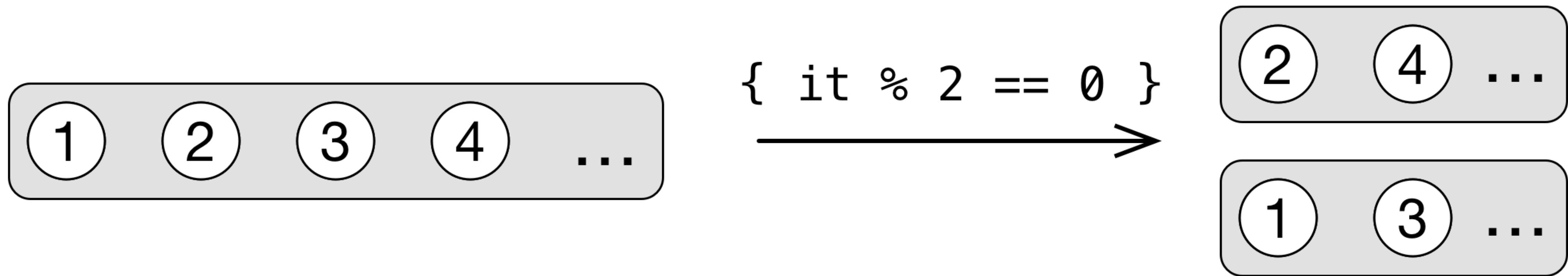
count



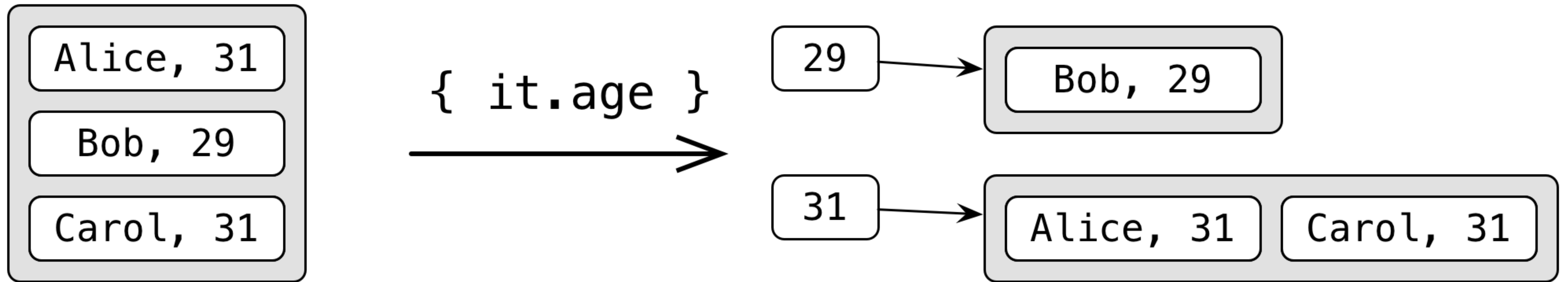
$\{ \text{it \% 2} == 0 \}$
→

2

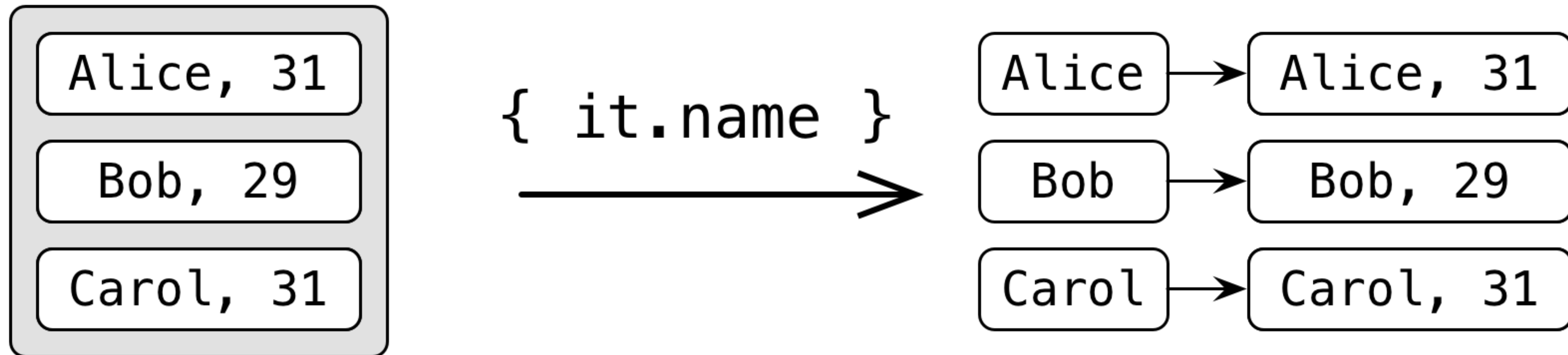
partition



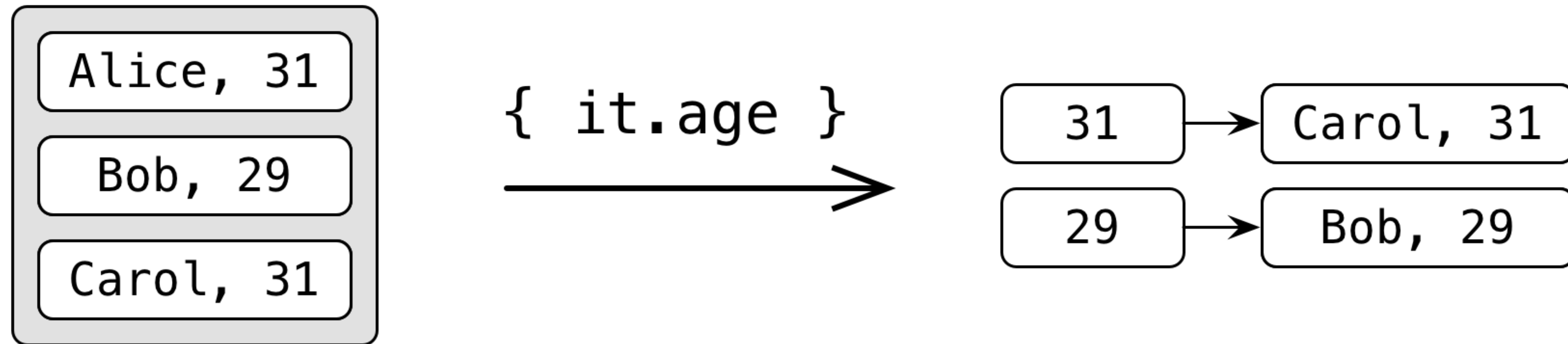
groupBy



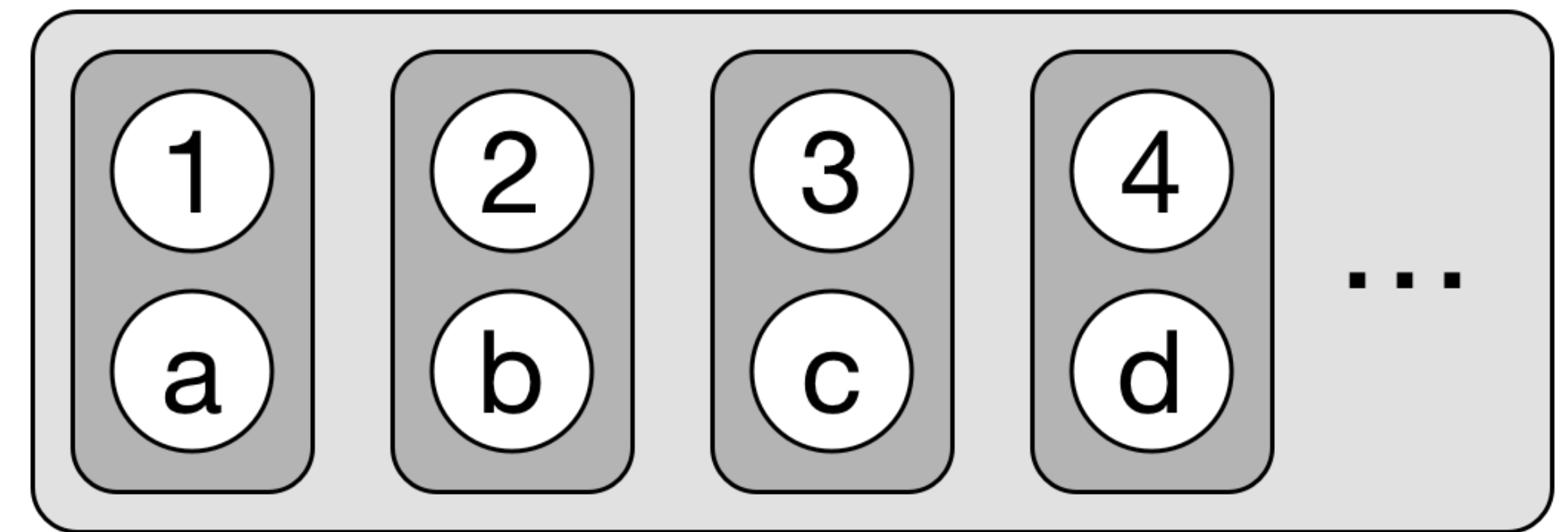
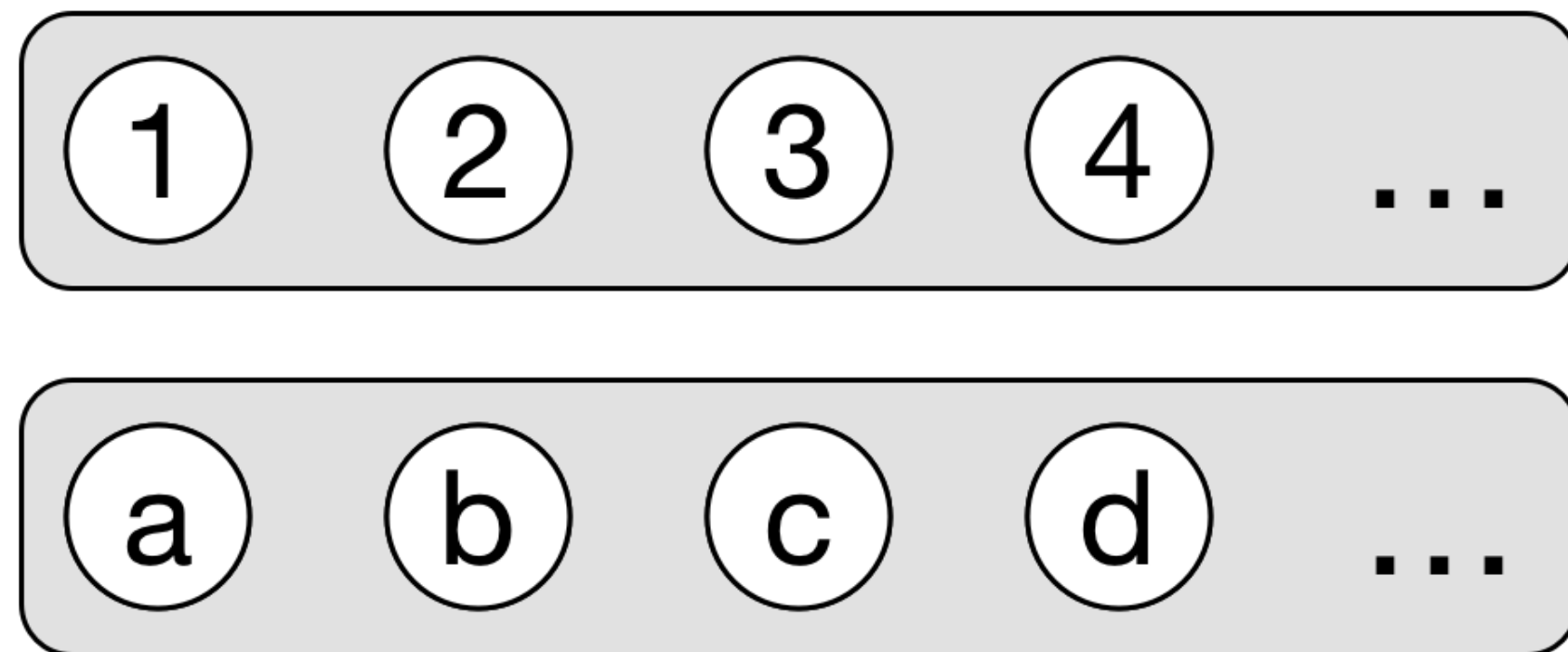
associateBy



associateBy (duplicates removed!)



zip



flatMap

