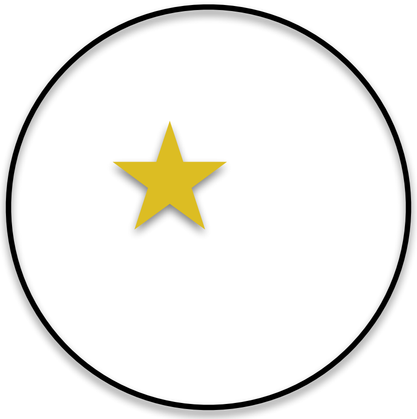
Improving Code with Functional Programming

Jessica Kerr
jessitron@jessitron.com
@jessitron

pluralsight
hardcore developer training

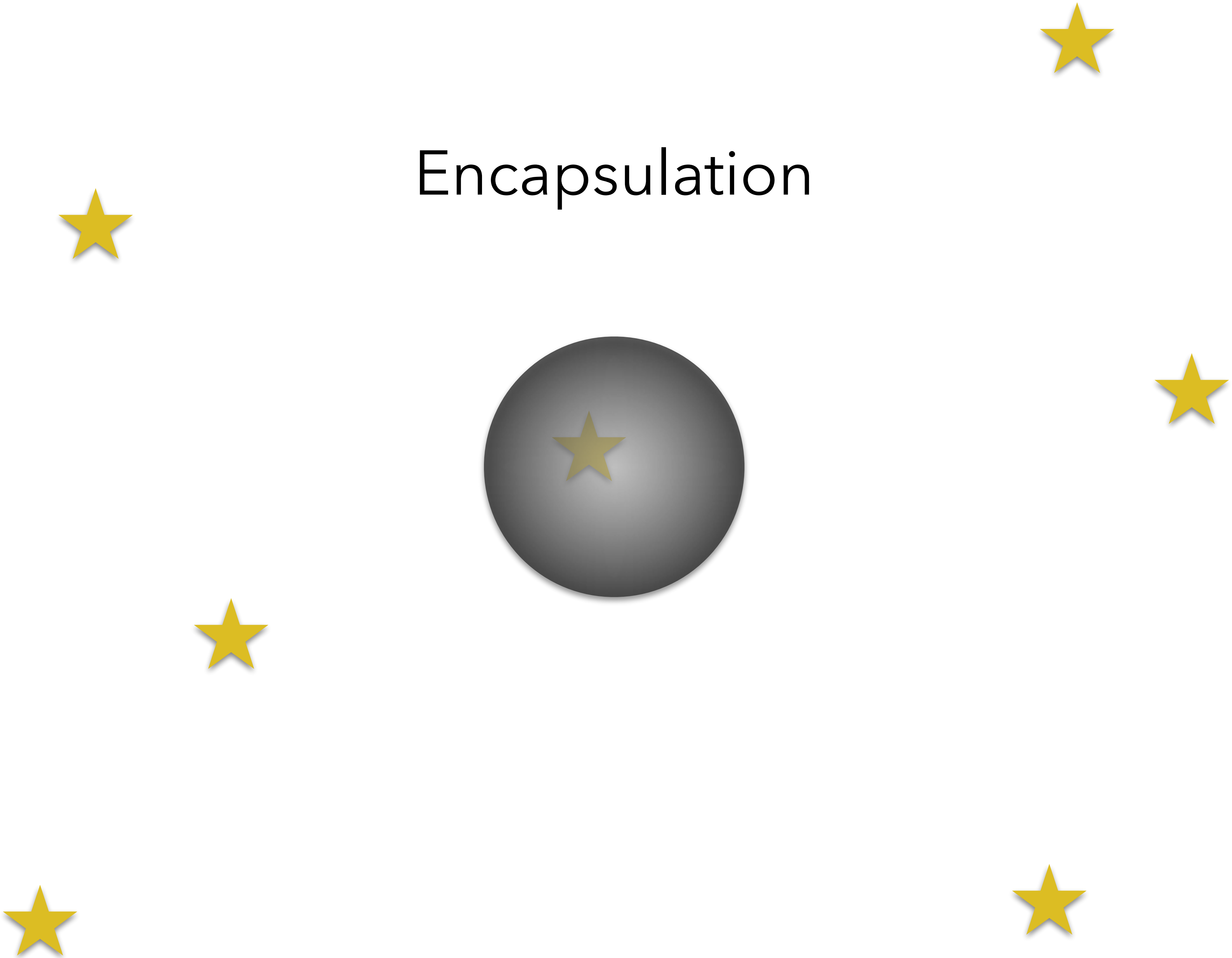
Current Mission:
to **remove code duplication**.

The Background:
timing sections of code.

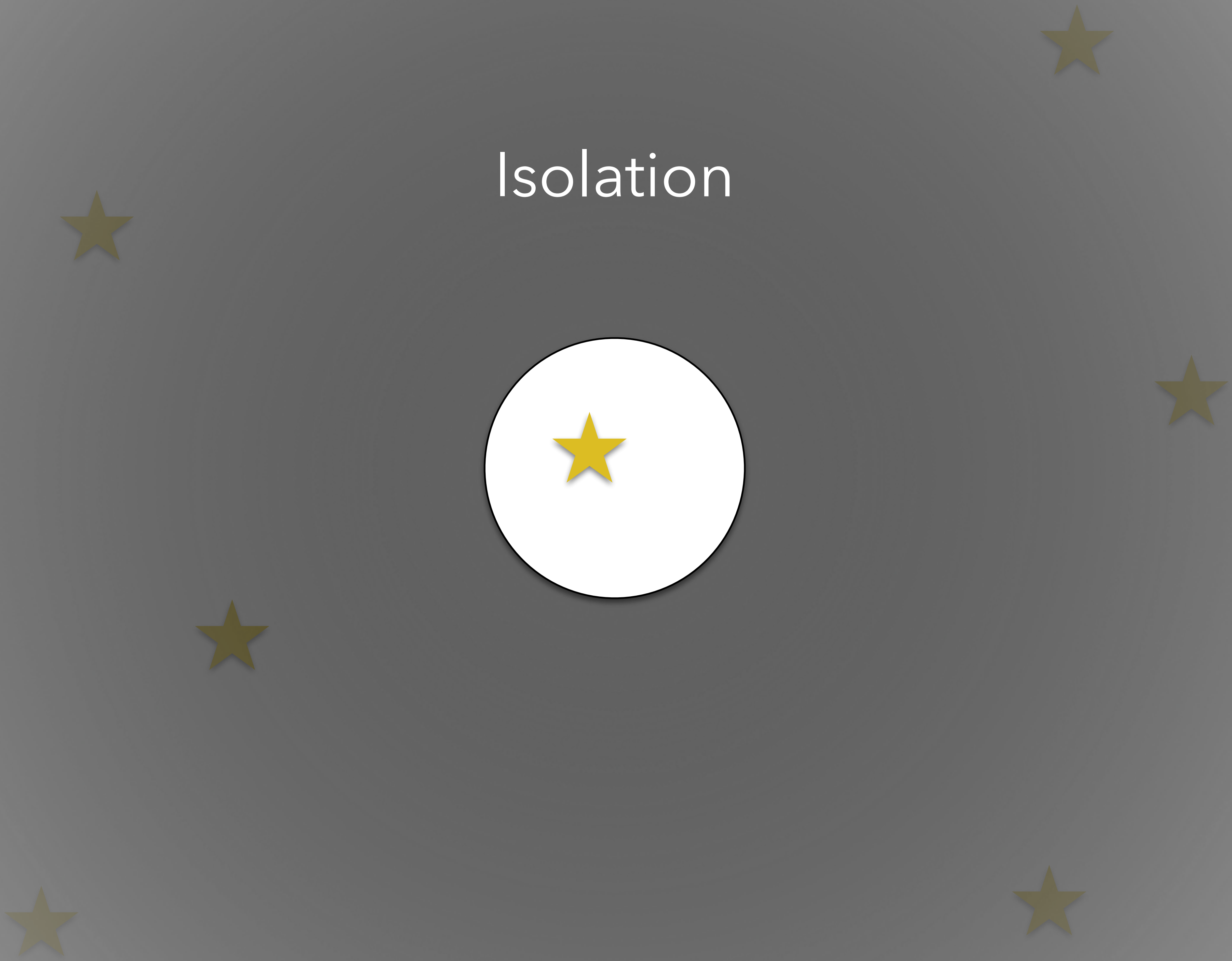


Encapsulation

Encapsulation



Isolation



✓ Specific Types

✓ Data In, Data Out

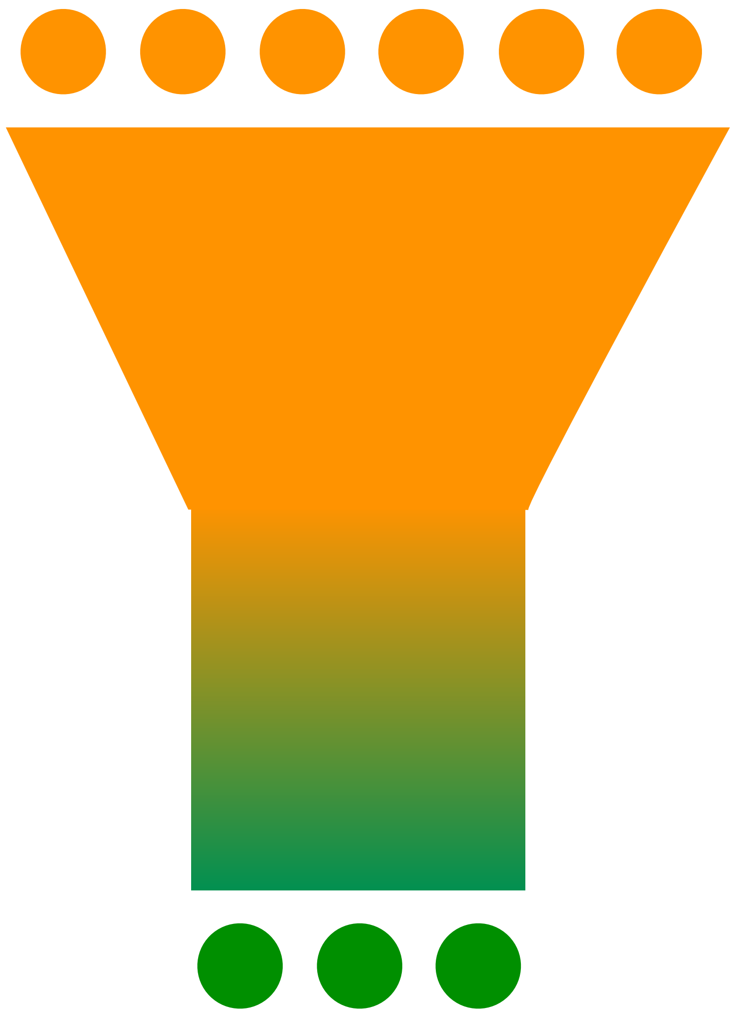
✓ Tested

filter

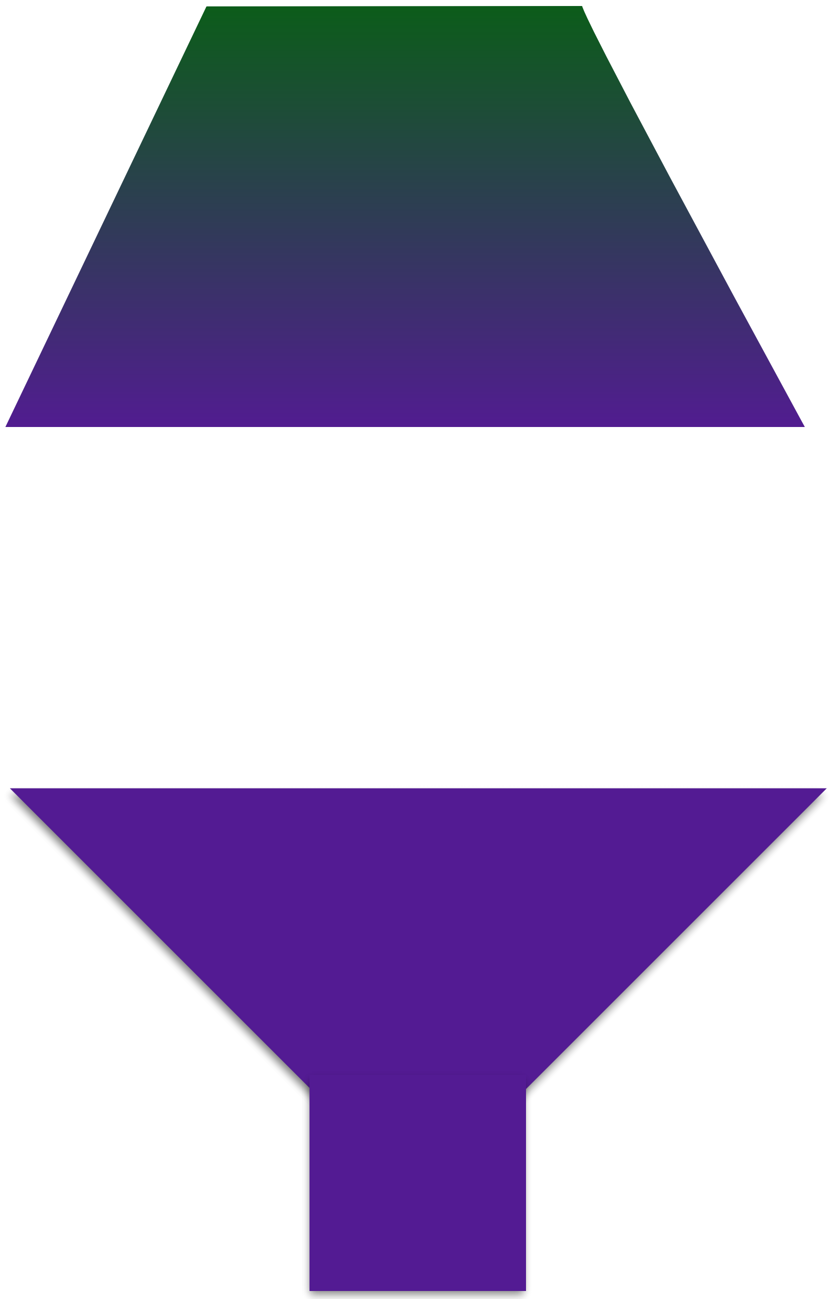
transform

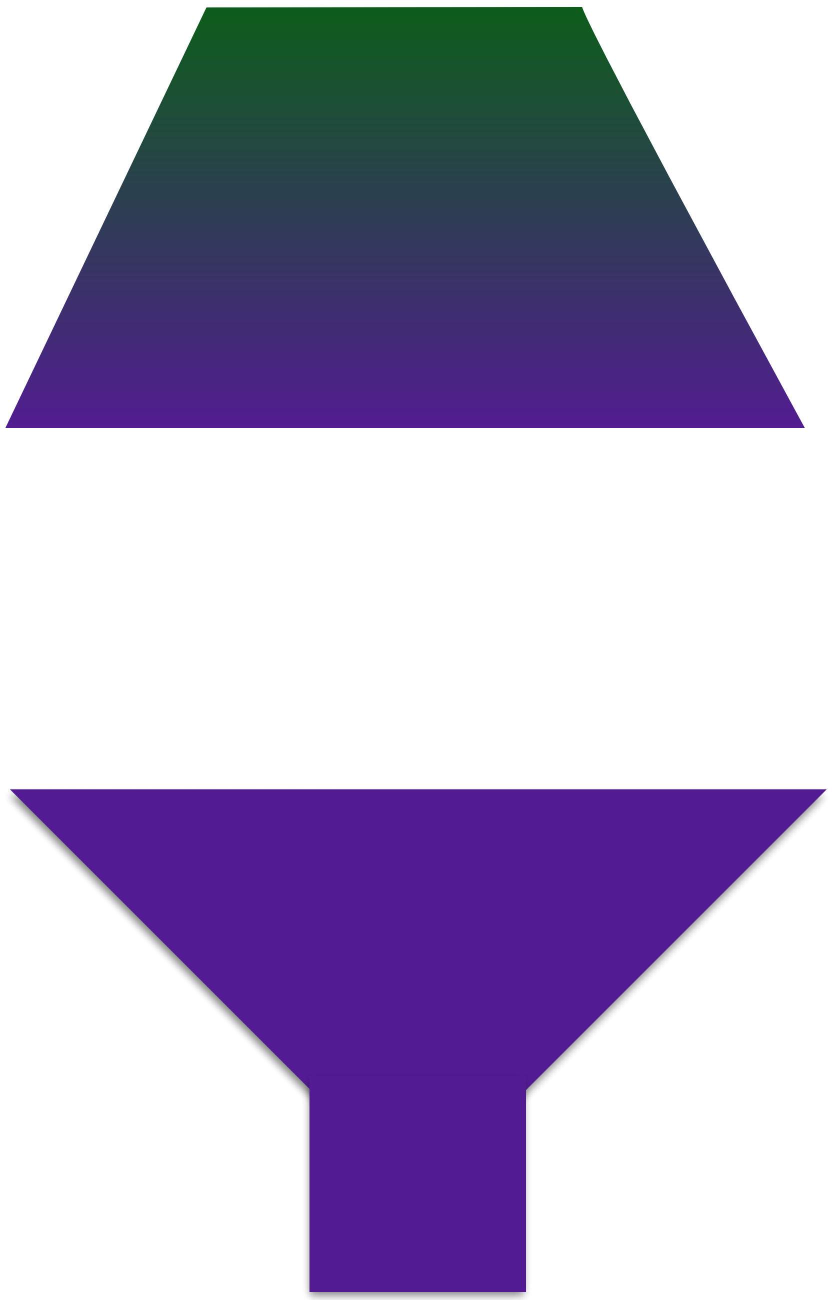
filter

transform



"little trees of broccoli"





✓ Java 8

✓ Java 6

Java 8

Java 6

External Iteration

Internal Iteration

FluentIterable

filter

transform

Joiner

Internal Iteration

FluentIterable	Stream
filter	filter
transform	map
Joiner	reduce

“carrots”

“bananas”

“pumpkins”

“broccoli”

“meat”

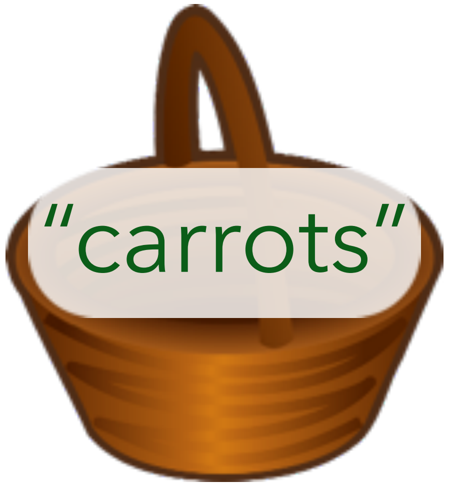
“carrots”

“bananas”

“pumpkins”

“broccoli”

“meat”



“carrots”

“bananas”

“pumpkins”

“broccoli”

“meat”

“carrots” “bananas”

“pumpkins”

“broccoli”

“meat”

“carrots & bananas”

“pumpkins”

“broccoli”

“meat”

“carrots & bananas”

“pumpkins”

“broccoli”

“meat”

“carrots & bananas” “pumpkins”

“broccoli”

“meat”

“carrots & bananas & pumpkins”

“broccoli”

“meat”

“carrots & bananas & pumpkins”

“broccoli”

“meat”

“carrots & bananas & pumpkins” “broccoli”

“meat”

“carrots & bananas & pumpkins & broccoli”

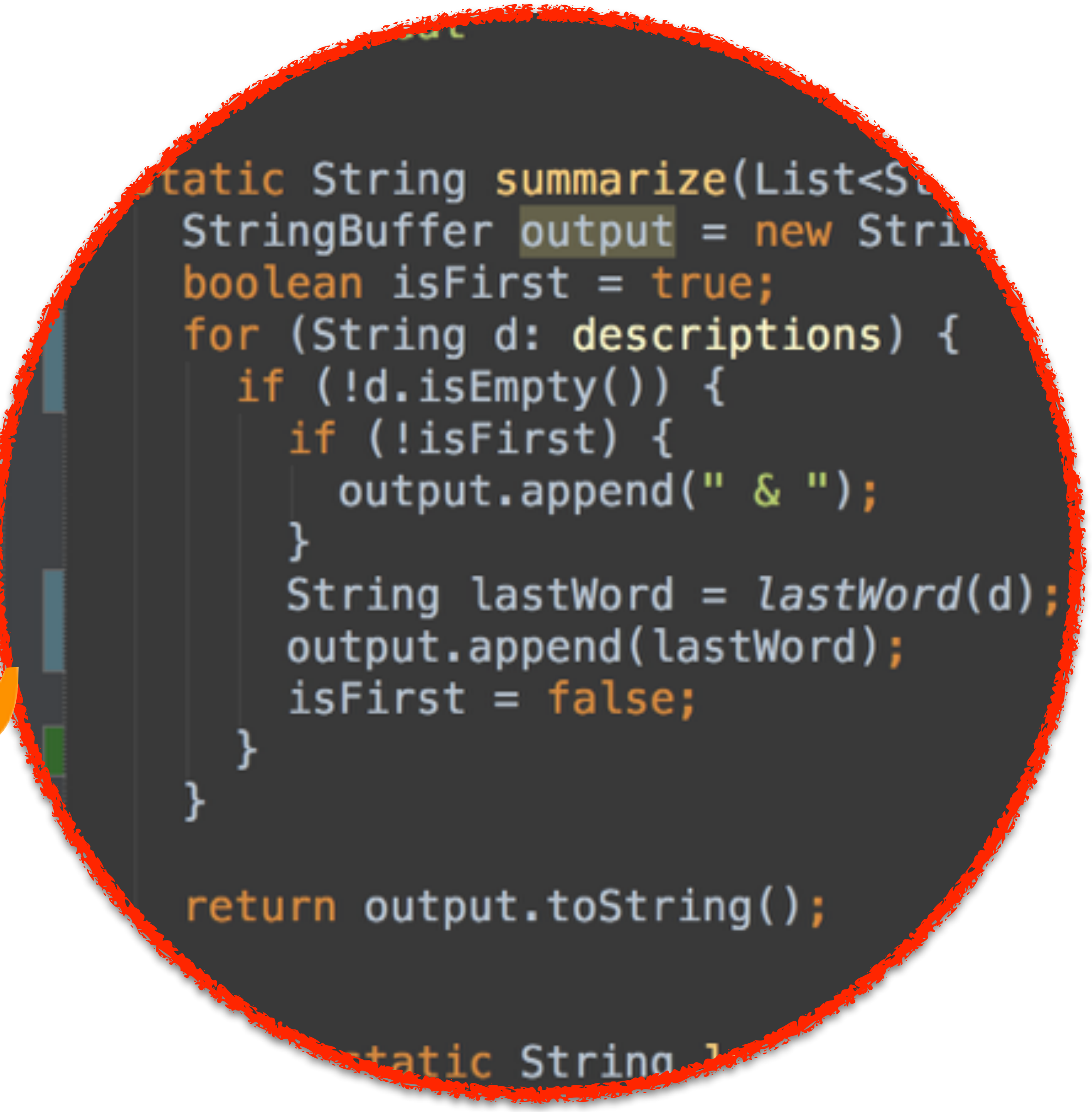
“meat”

“carrots & bananas & pumpkins & broccoli”   “meat”

“carrots & bananas & pumpkins & broccoli” “meat”

“carrots & bananas & pumpkins & broccoli & meat” “carrots & bananas & pumpkins & broccoli & meat”

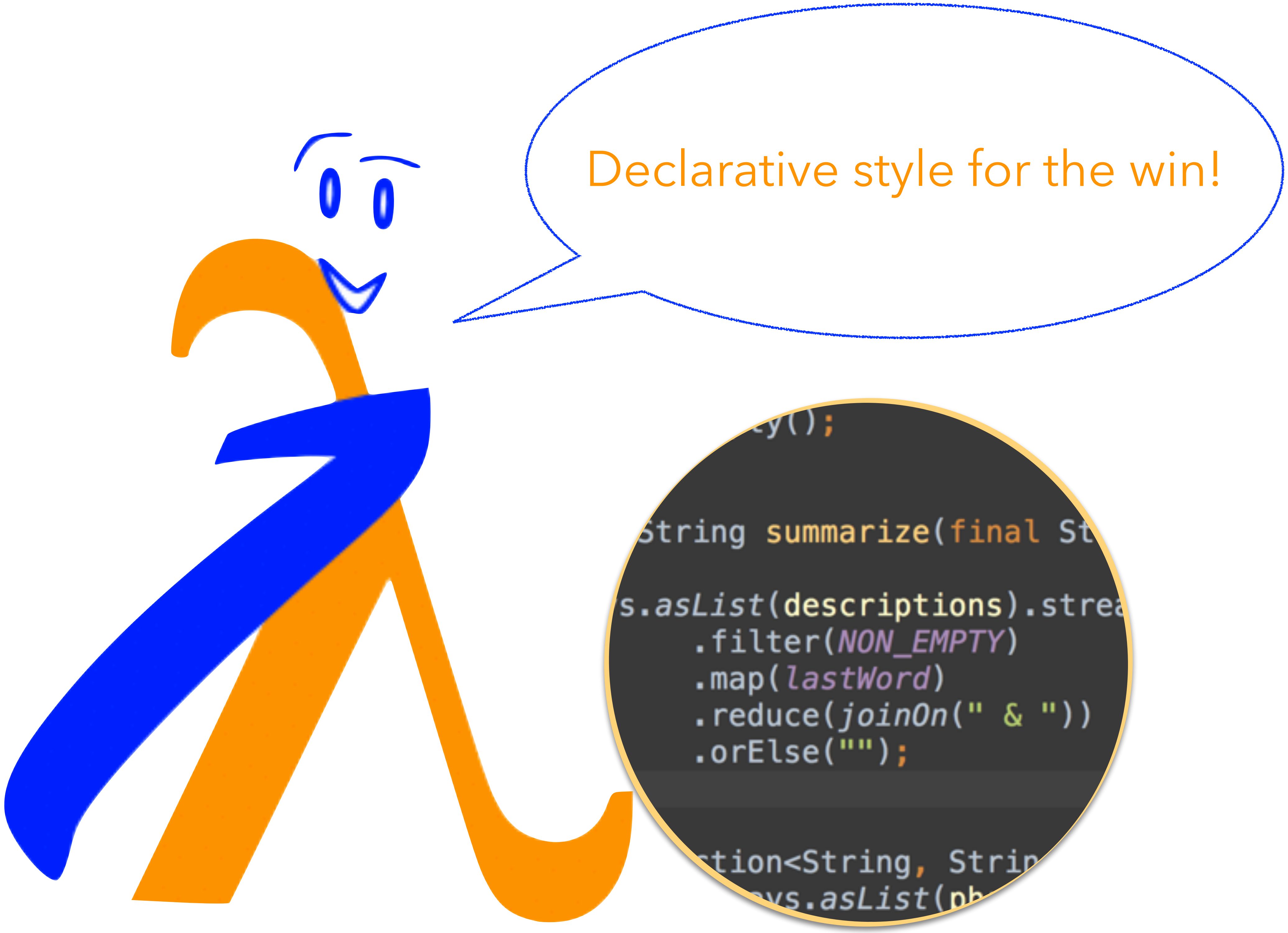
Well, it’s better than “null.”



```
static String summarize(List<St
  StringBuffer output = new Stri
  boolean isFirst = true;
  for (String d: descriptions) {
    if (!d.isEmpty()) {
      if (!isFirst) {
        output.append(" & ");
      }
      String lastWord = lastWord(d);
      output.append(lastWord);
      isFirst = false;
    }
  }

  return output.toString();


        static String l
```



```
   ty();

String summarize(final St

s.asList(descriptions).strea
      .filter(NON_EMPTY)
      .map(lastWord)
      .reduce(joinOn(" & "))
      .orElse("");


tion<String, Strin
   ys.asList(ph
```