


Agenda

- ① Higher Order Function & Callback
- ② Pure & Impure Functions.
- ③ Writing Clean code w/ HOF
- ④ Example Array Methods -

~~Recap~~

- ① Lexical Scope
- ② Closures

Execution Context

Classification of Programming Language (Paradigm)

- ① Procedural Language (C) → low level language
 - ↳ how does it work
- ② Object Oriented language
 - (Java, C++, Python)
 - ↳ What is the way it works
- ③ Functional Programming Language
 - (Haskell)

Multi-Paradigm Language

JS

Object-Oriented ✓ (does not allow
Inheritance)

Functional ✓

Callback Function

- * These are functions which are received as arguments to other functions & "called back" later.

Pure Functions

Any function that does not contain any side-effect.

Any impact which is unintended.

DRY

→ DON'T

REPEAT

YOURSELF

Modular

Higher Order Functions

↳ Those functions where. →

- ✓ ① Functions are passed as arguments.

OR

- ✓ ② Functions are returned from them.

The built HOFs for Array

MAP

[1

2

3

4

5

(3)

ProcessFunction

E

X

y

4

B

c

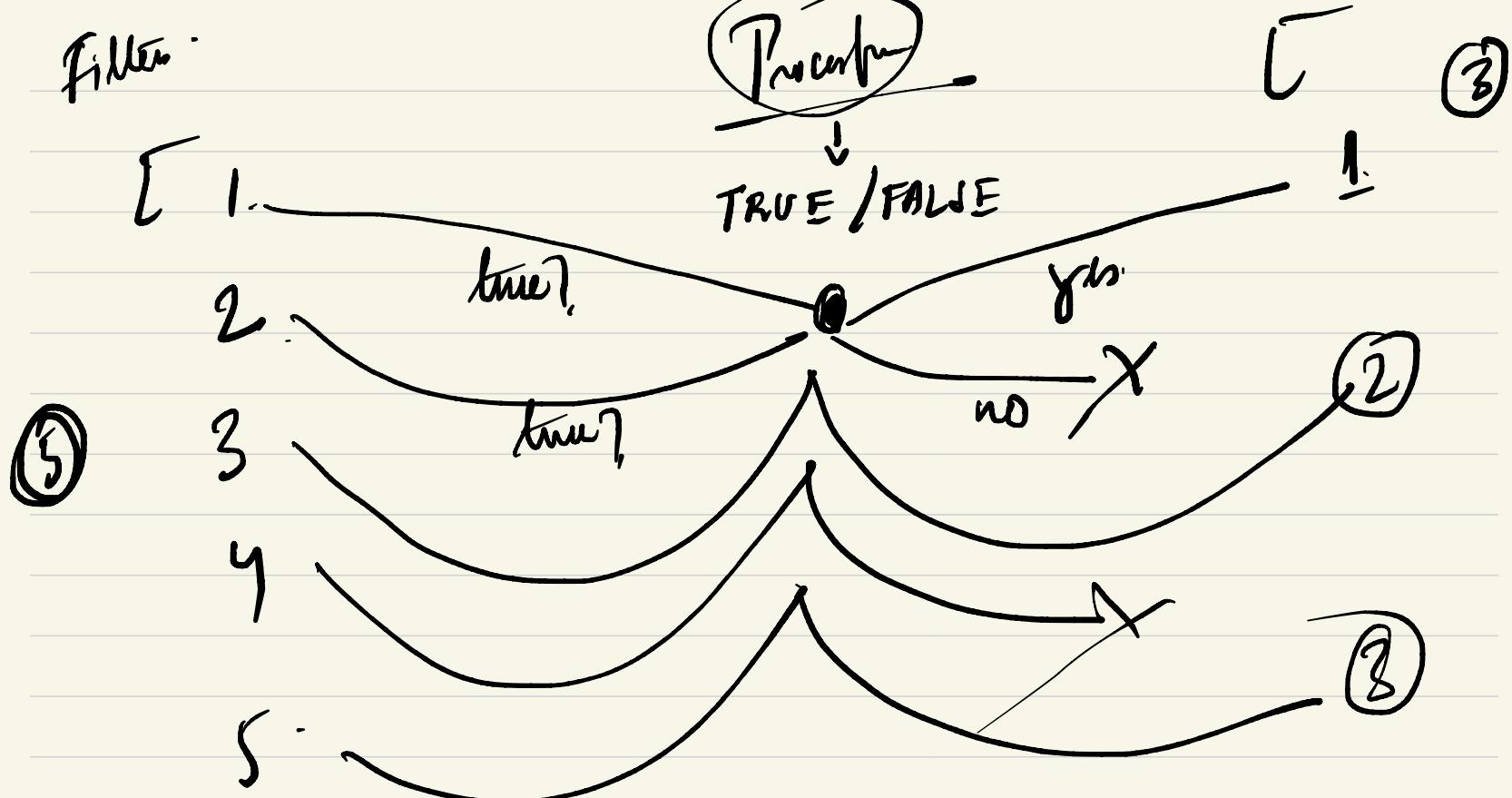
(5)

Fallen

Prozessor

C

③



Every value in JS can beTruthy OR Falsey

if (condition)
 Truey
 Falsey

arr-reduce $(\cdot \text{ (acc, item)}) = \{$ DRY RUN

acc = acc + item

return acc

3, 0)

[1, 2, 3, 4, 5]

initial value of acc = 0

0 1st

2nd

3rd

acc = 0
item = 1
acc = 1

1 + 2

3 + 3

1
2
3
4
5

return 1

1 + 4

10

3

5

10 + 5

15

If no initial value is passed

↳ the first element becomes our value

↳ Array loop runs from the second

our

Reduce

T

2

3

4

5

growing

arc

Tidal
arc

