Advanced Concepts of Functional Programming

Adhithi Ravichandran

Software Consultant | Author | Speaker

@AdhithiRavi | www.adhithiravichandran.com

Recursion

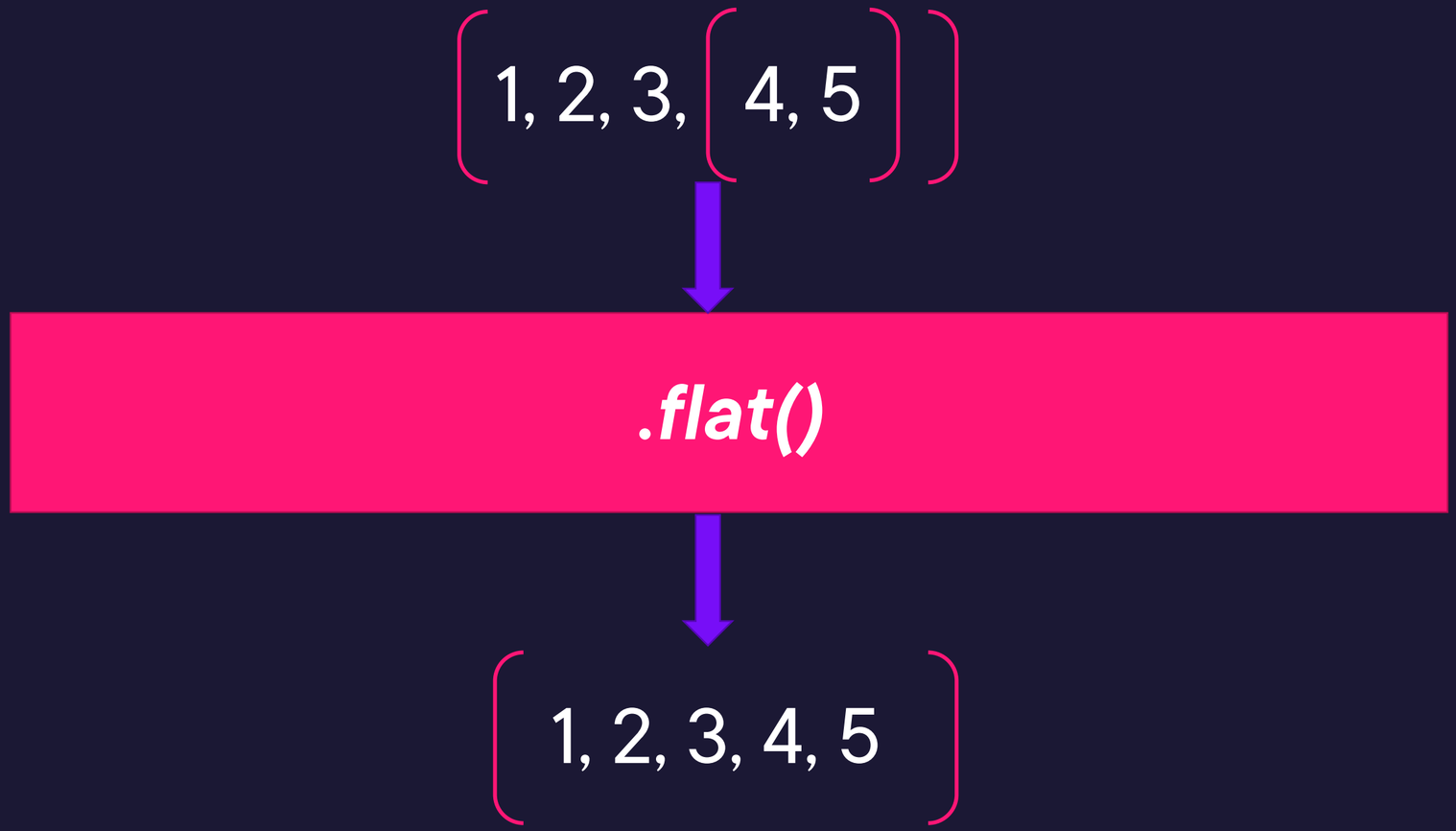
Recursion

A function calling itself is recursion.

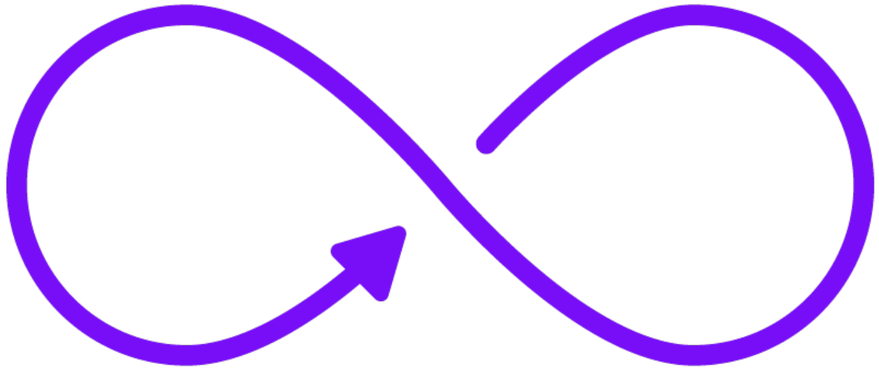
Needs a stopping point, if not ends up calling itself forever!

Could be a harder concept to understand.

Flat Is a Recursive Function



Recursive functions let you perform a unit of work multiple times.



Demo

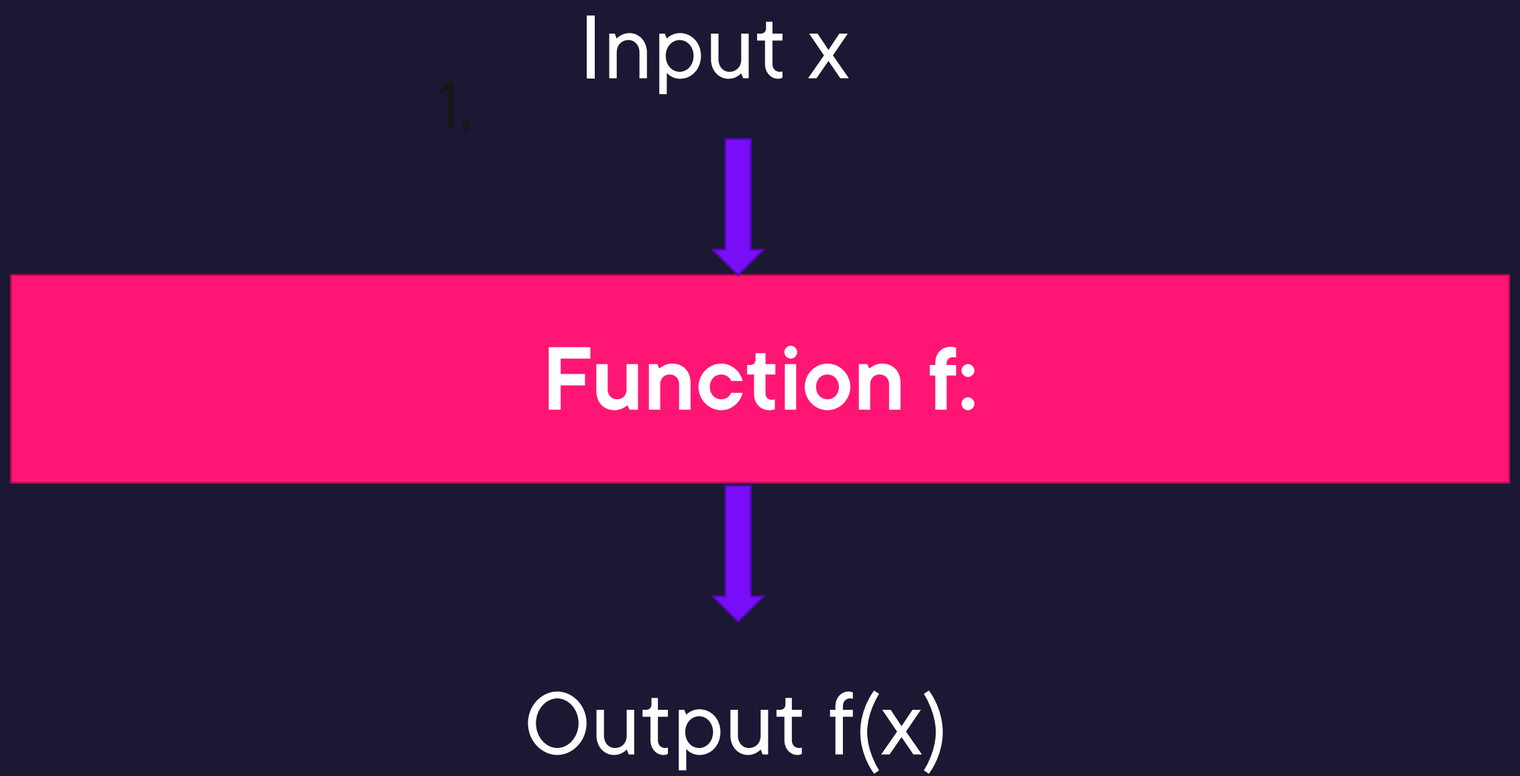
Recursive approach vs. Imperative approach

Higher-order Functions

Higher-order Functions

1. Accepts a function as an argument
2. and/or returns a function

Higher-order Functions



Built-in Higher-order Functions

.map()

.filter()

.reduce

```
const initialNumbers = [1, 2, 3, 4, 5];

const addTwo = (number) => number + 2;

const newNumbers = initialNumbers.map(addTwo);
```

Map Function

```
const isWordGreaterThanSeven = (word) => word.length > 7;

const result = words.filter(isWordGreaterThanSeven);
```

Filter Function

```
const getMax = (a, b) => Math.max(a, b);

const max = initialArray.reduce(getMax); // 5
```

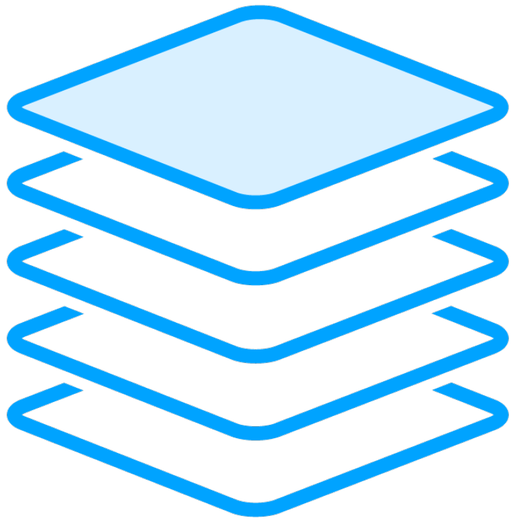
Reduce Function

Demo

Create your own custom Higher-order Function in JS

Currying

Currying



Transforms a function with multiple arguments into several functions containing a single argument.

Wraps a function inside a function.

Demo

Currying in JS

Summary

Recap

Advanced Concepts in JS:

- Recursion
- Higher-Order Functions
- Currying

Up Next:

JavaScript Frameworks