

An Introduction to Elixir

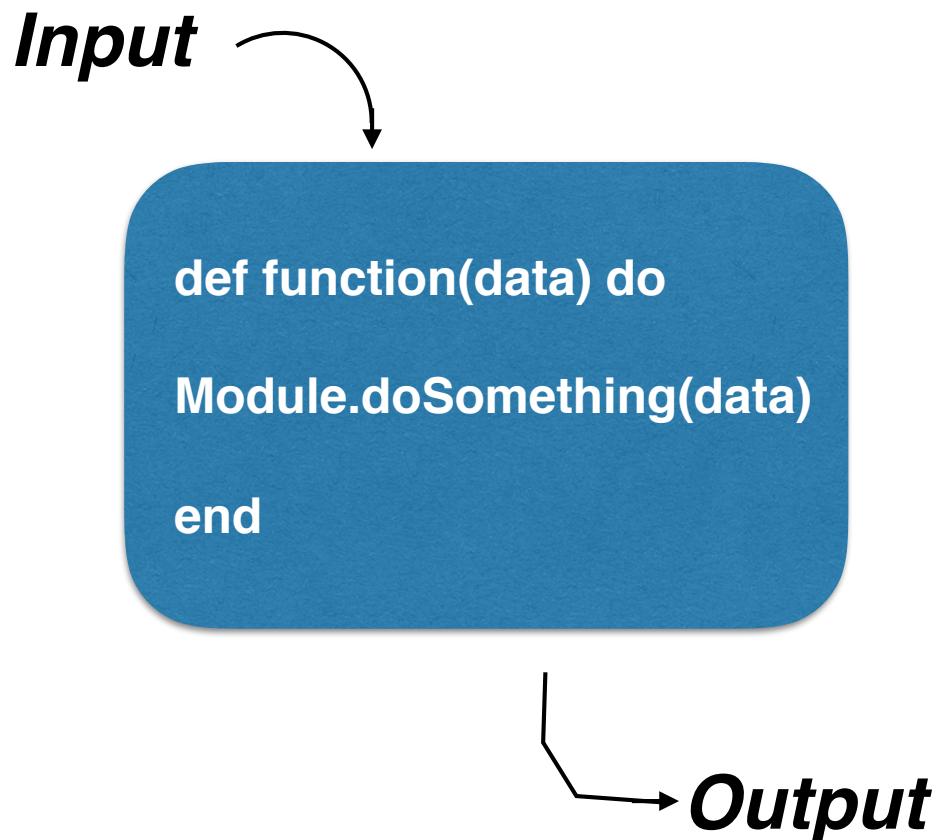
Daisy Mølving

HELLO
MY NAME IS



Functional Programming

- Functions
- Immutable Data
- Pure Functions
- Reduced Side Effects



Let's get started!

- mix `mix new project_name`
- iex (Interactive Elixir REPL)
- Help and documentation
- Helpful error messages

Module.function(data)

Module ->

function_one(data)

function_two(data)

function_three(data)

function_four(data)

Module.function(data)

String ->

`capitalize("hello")`

`contains?("hello", "lo")`

`duplicate("elixir", 2)`

`reverse("blue")`

**Functions
act upon
inputs
(data) that
are strings**

list = [1, 2, 3, 4, 5]

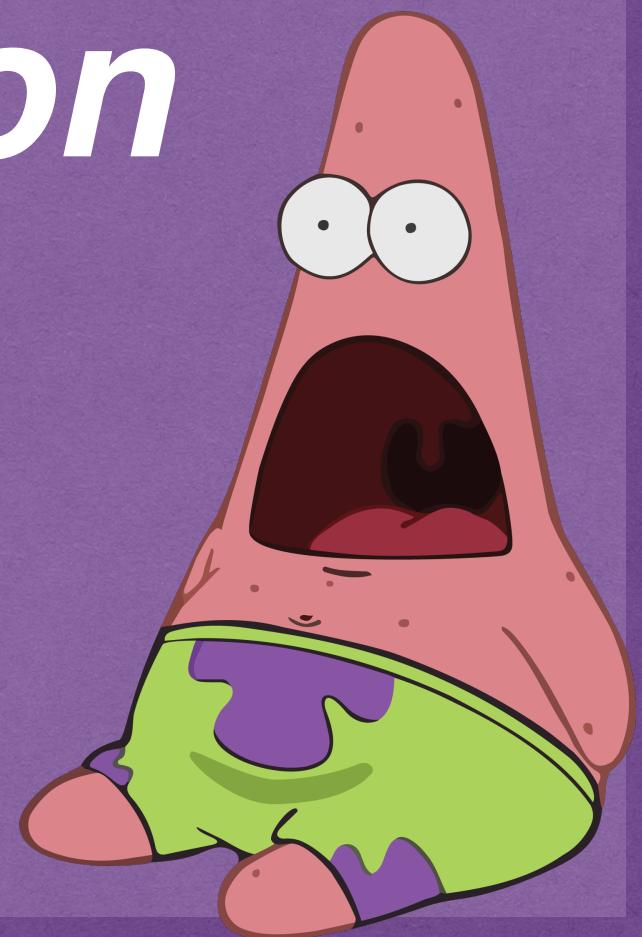
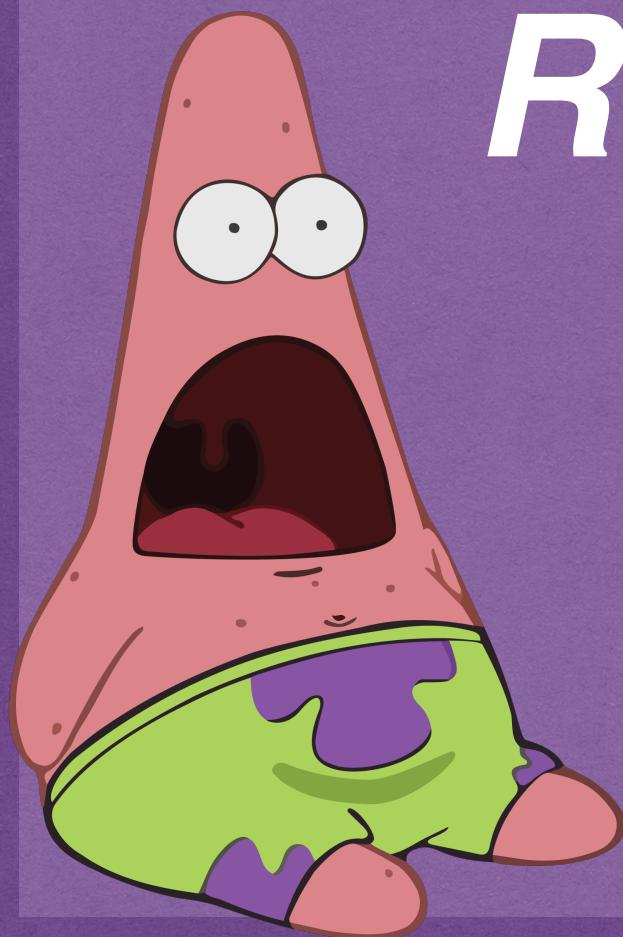


sum(list)



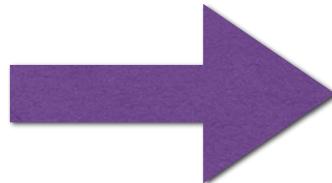
1 + 2 + 3 + 4 + 5

Recursion



LOOPING

```
def loop_greet
    num = 0
    while num <= 5 do
        puts "hello!"
        num += 1
    end
end
```



```
def recurse(num)
    if num == 5
        puts "hello!"
    else
        puts "hello!"
        recurse(num + 1)
    end
end
```

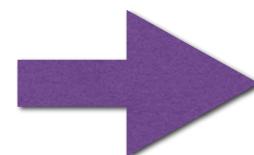
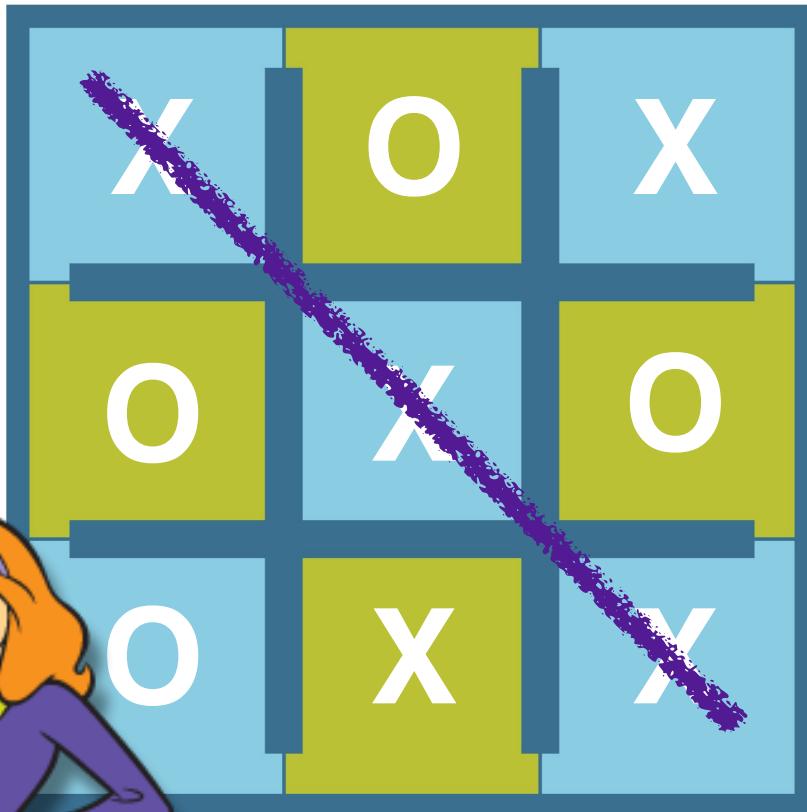


Pattern-Matching



=
=





```
current_board = [  
    "X", "O", "X",  
    "O", "X", "O",  
    "O", "X", "X"]
```



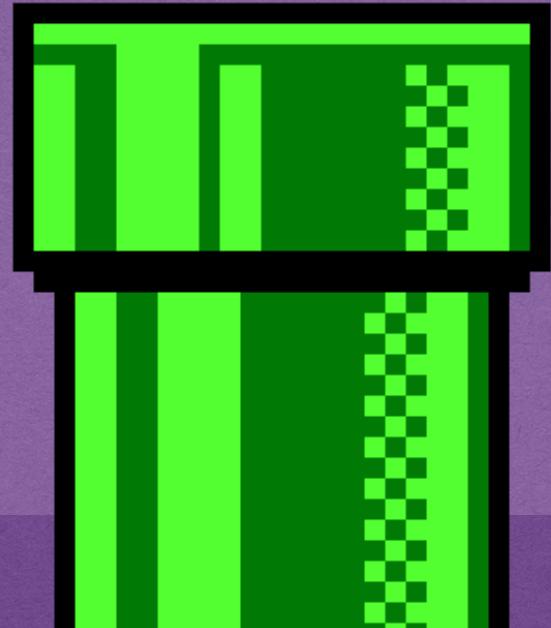
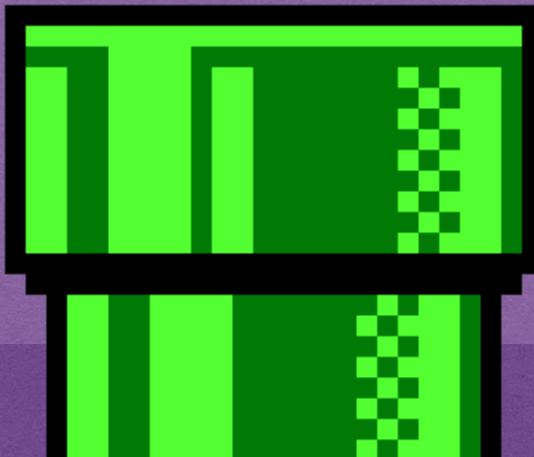
Diagonals?

[head | tail] → [first | rest]



Pipe Operator

|>



```
List.last(Integer.digits(String.to_integer("17")))
```

“17”

|> String.to_integer

|> Integer.digits

|> List.last



Resources

<https://github.com/elixirkoans/elixir-koans>

Screenshot of the GitHub repository page for `elixirkoans / elixir-koans`.

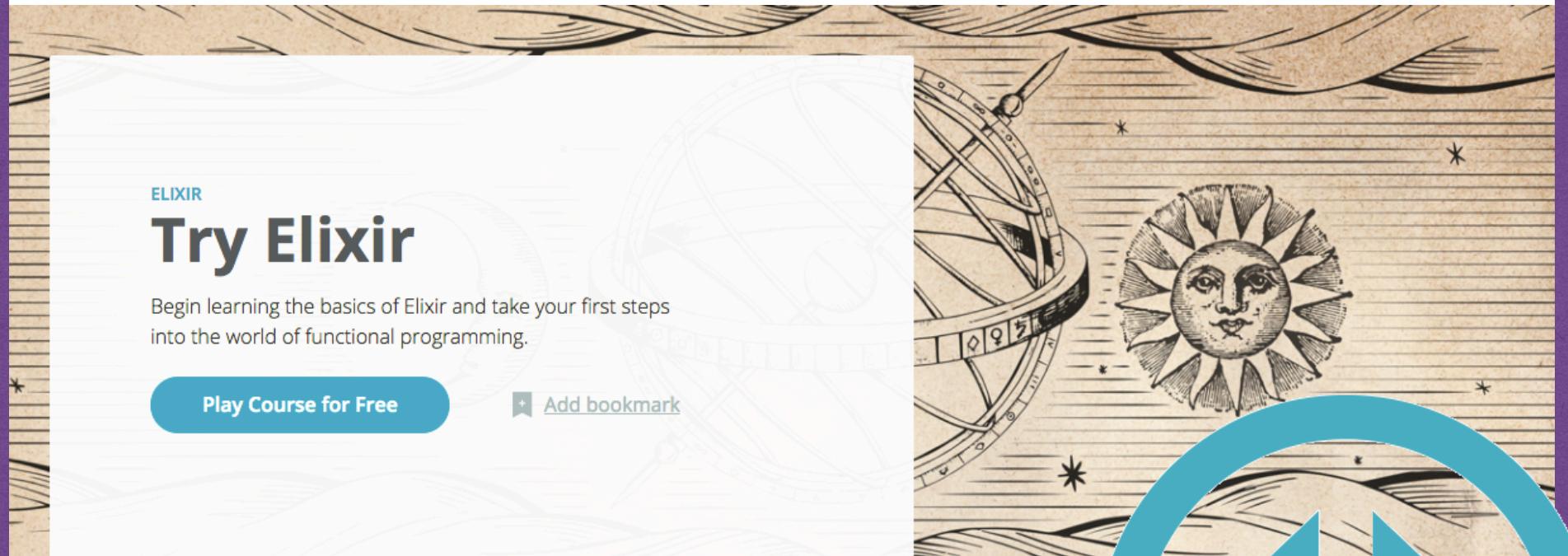
The repository has the following statistics:

- Code: 471 commits
- Issues: 14
- Pull requests: 2
- Projects: 0
- Pulse
- Graphs

The repository has 27 watchers, 497 stars, and 144 forks.

A large GitHub cat mascot is prominently displayed on the right side of the page.

<https://www.codeschool.com/courses/try-elixir>



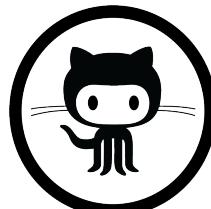
Elixir School

Lessons about the Elixir
programming language

▼ Basics

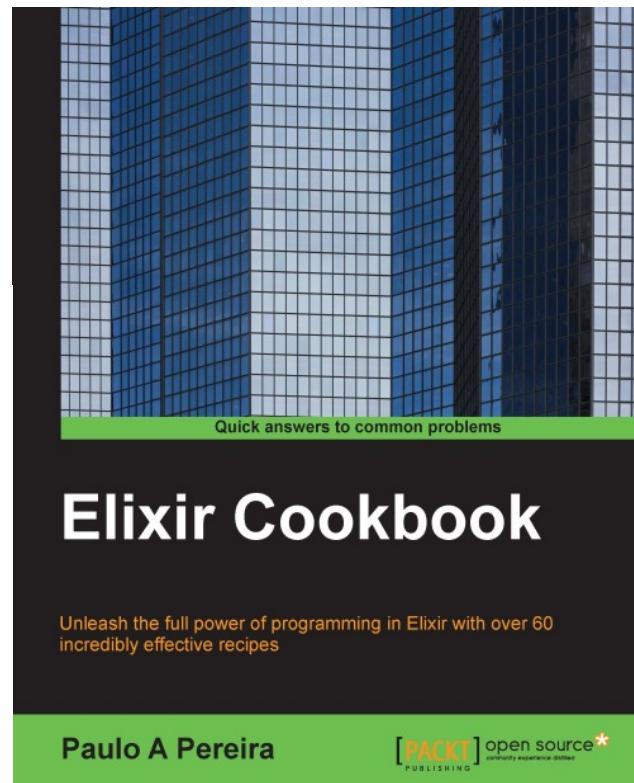
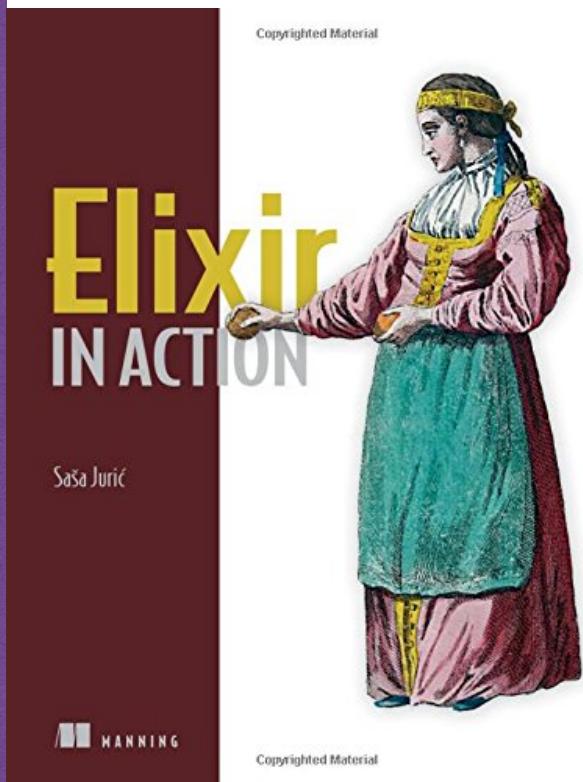
1. Basics
2. Collections
3. Enum
4. Pattern Matching
5. Control Structures
6. Functions
7. Pipe Operator
8. Modules
9. Mix
10. Sigils

<http://elixirschool.com/>



/doomspong/elixir-school

Books!



Programming Elixir

Functional
|> Concurrent
|> Pragmatic
|> Fun

Dave Thomas

Foreword by
José Valim,
Creator of Elixir

edited by Lynn Beighley





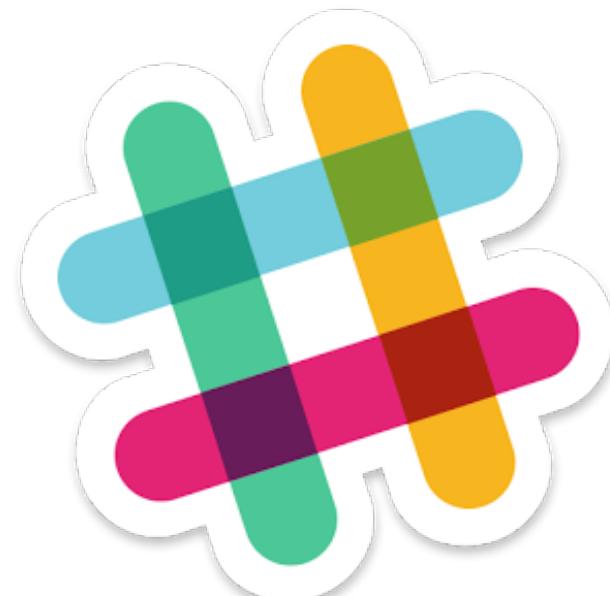
Elixir Community

THE ELIXIR FOUNTAIN

hosted by
Johnny Winn



elixir  forum



Thank
You!

