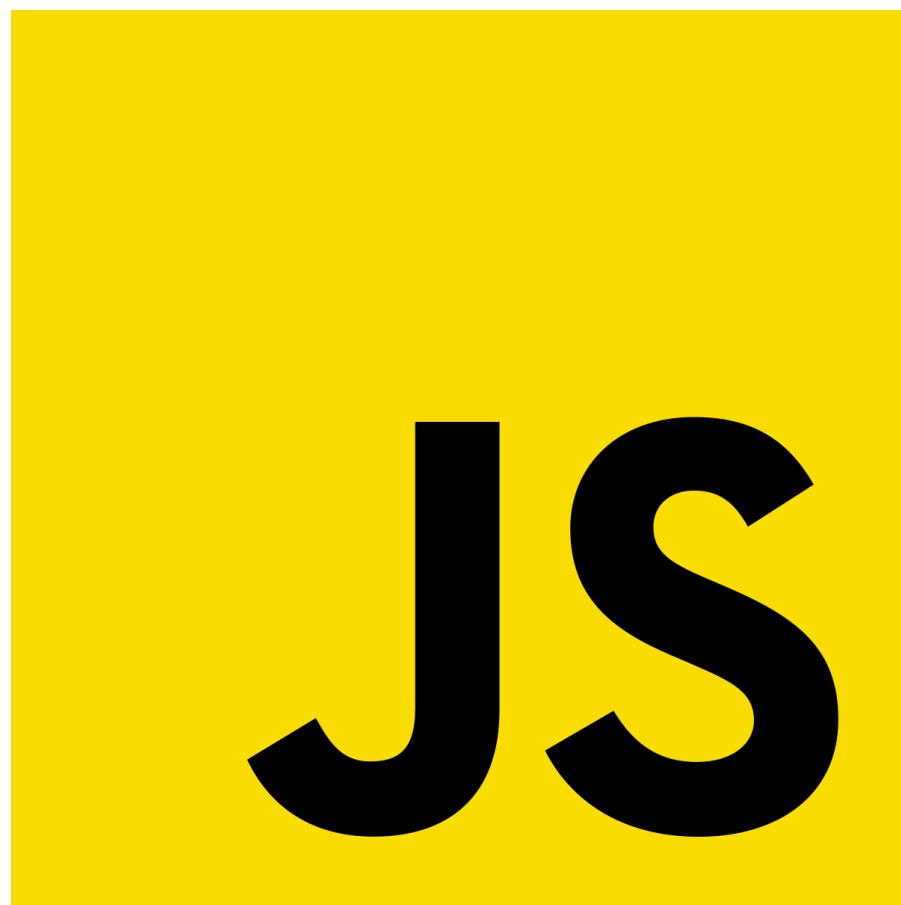
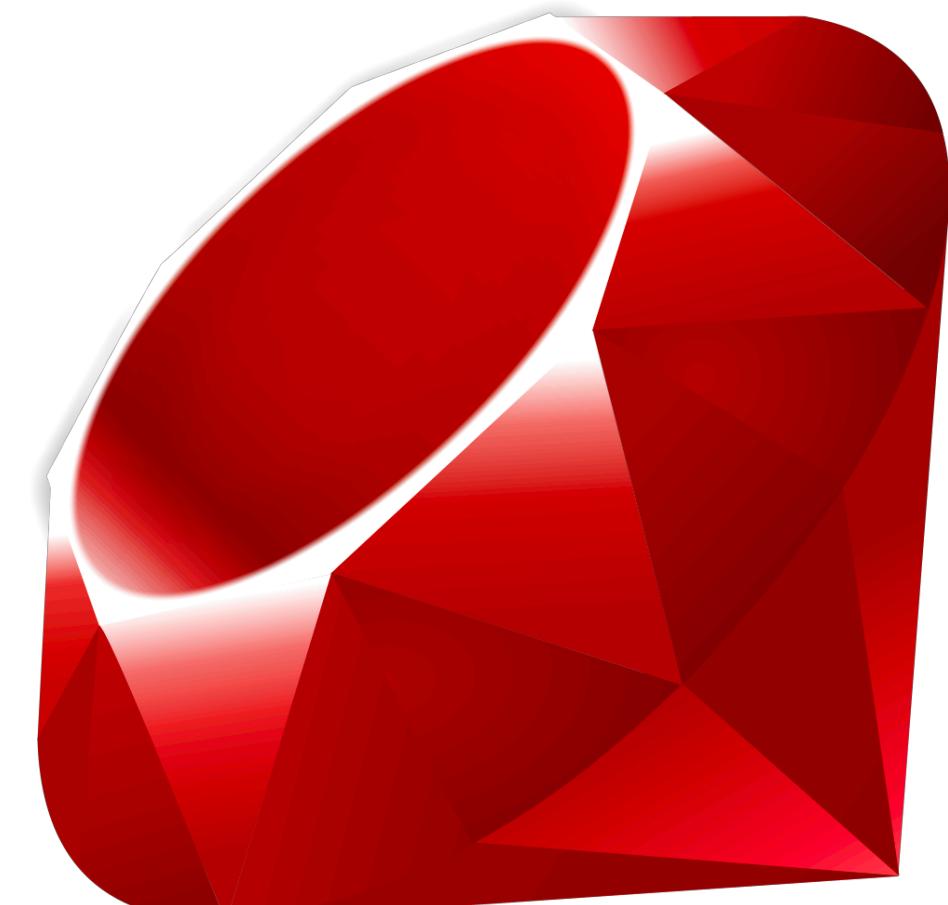


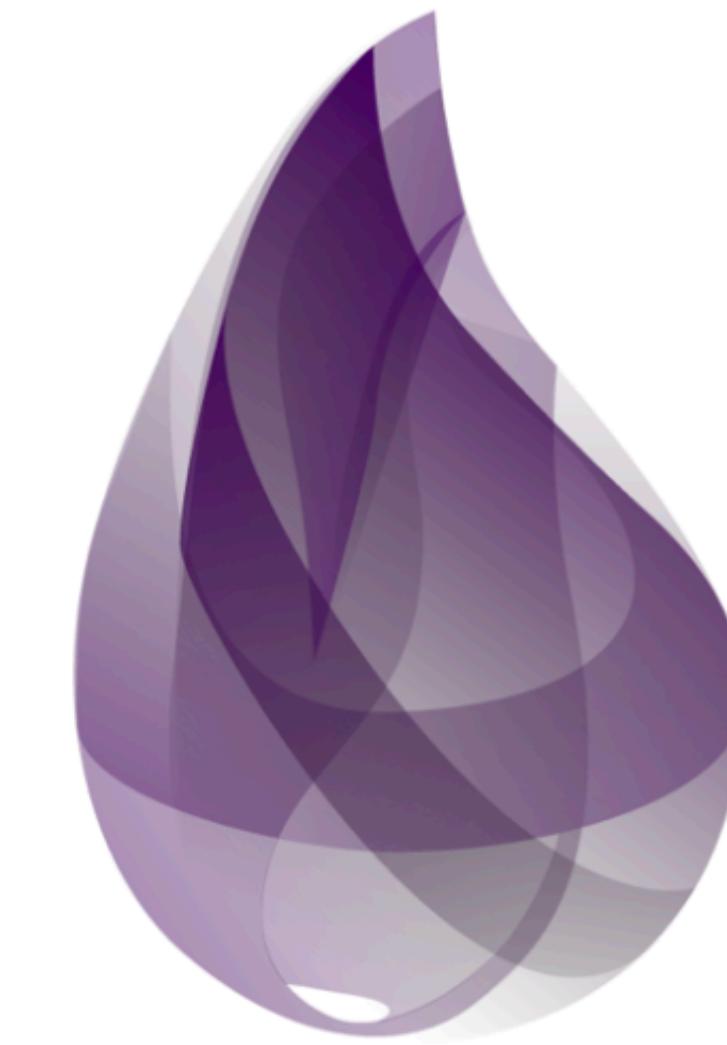
ELIXIR 
▷ CASUALLY



|>



|>

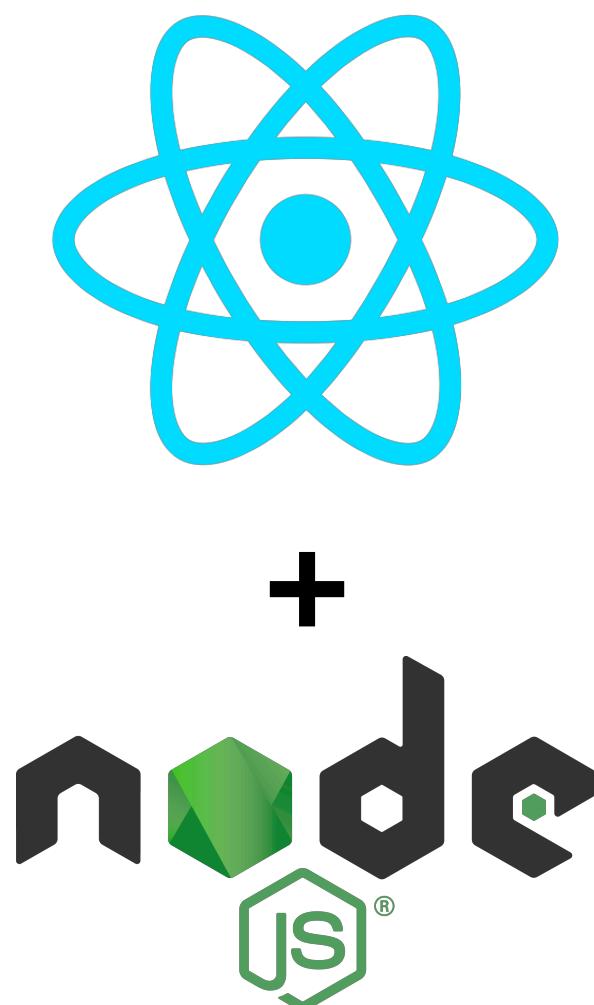
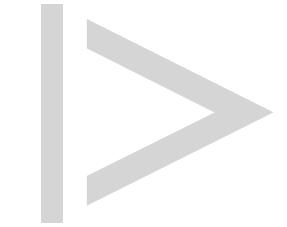


My Journey

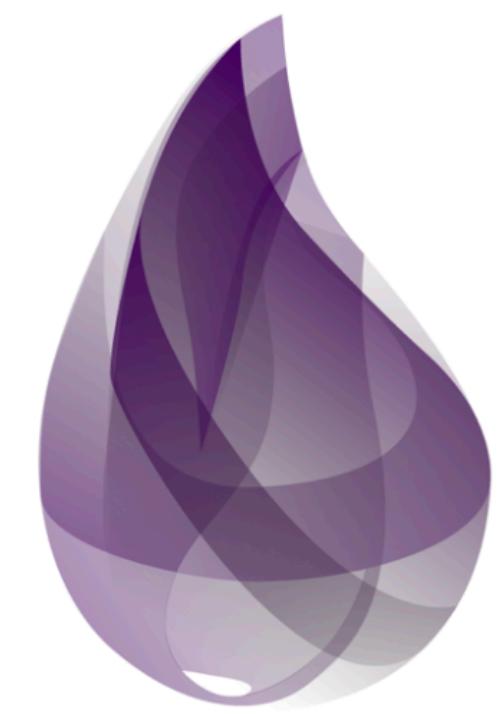
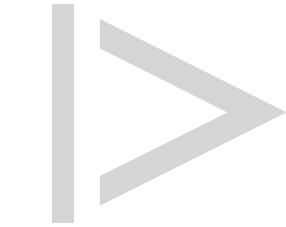


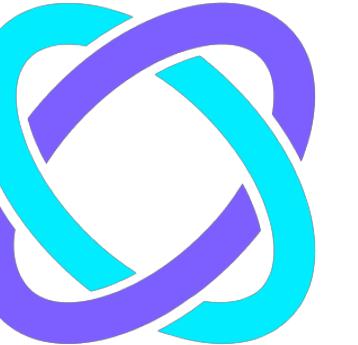
+

django



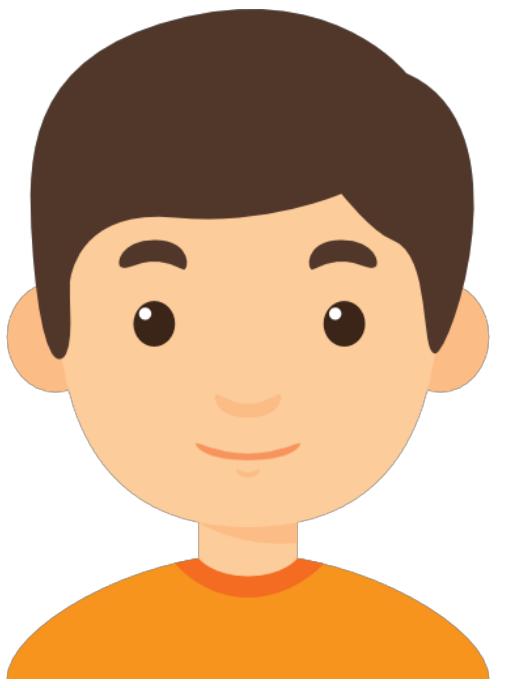
+



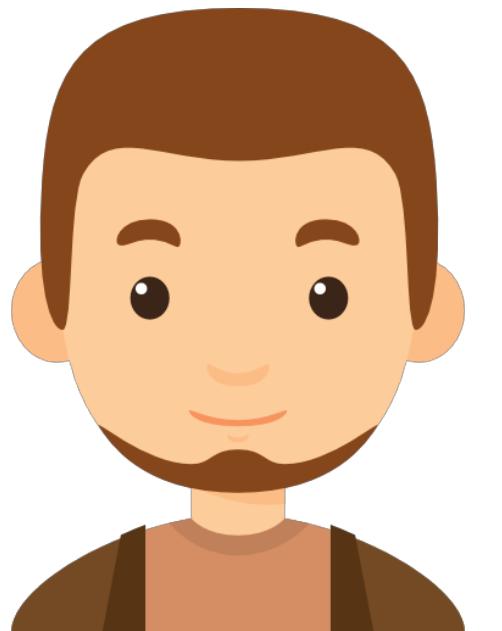
Why  flipay ❤️  ?



Concurrency in NodeJS



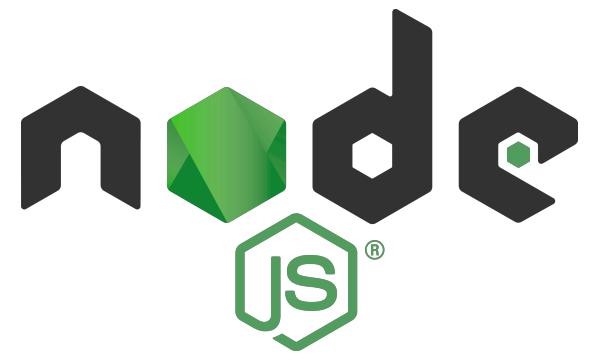
Request



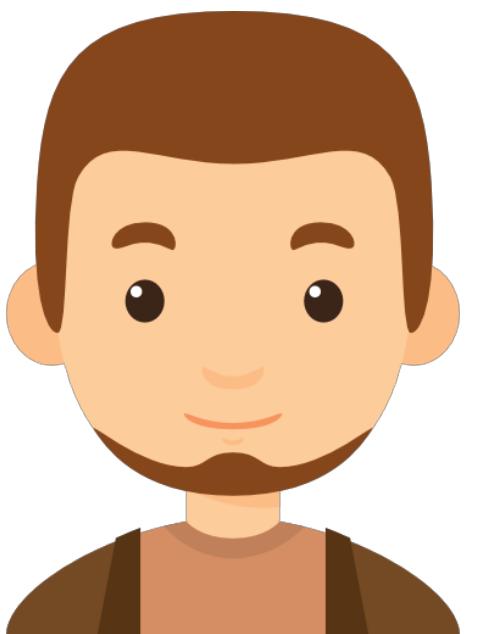
Request



icon from <https://www.freepik.com>



Sit & Wait

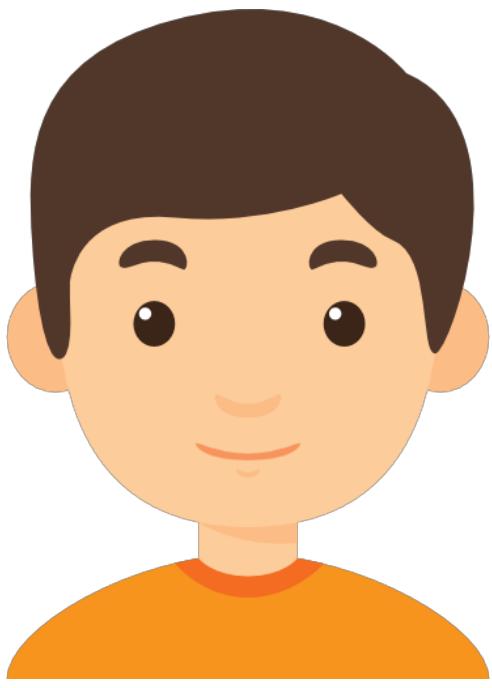


Sit & Wait

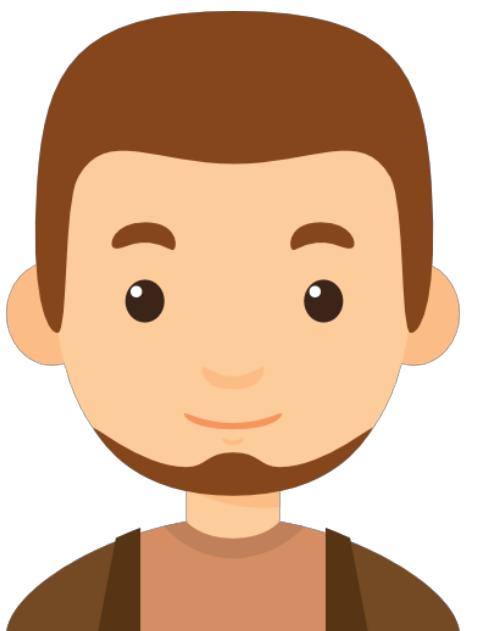
Request
Request



icon from <https://www.freepik.com>



Sit & Wait



Sit & Wait

Response

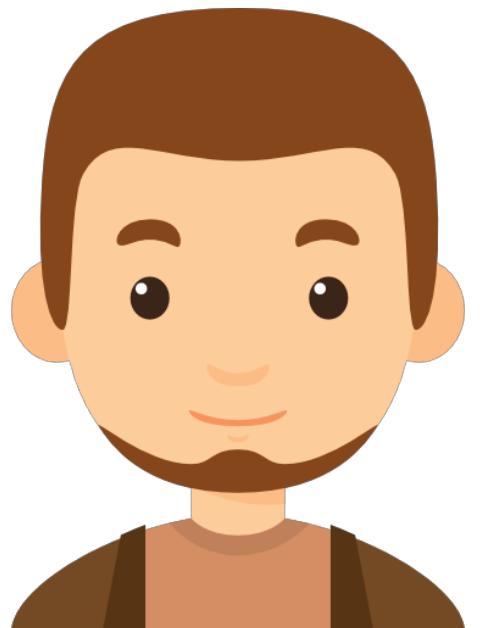
Request



icon from <https://www.freepik.com>



Response

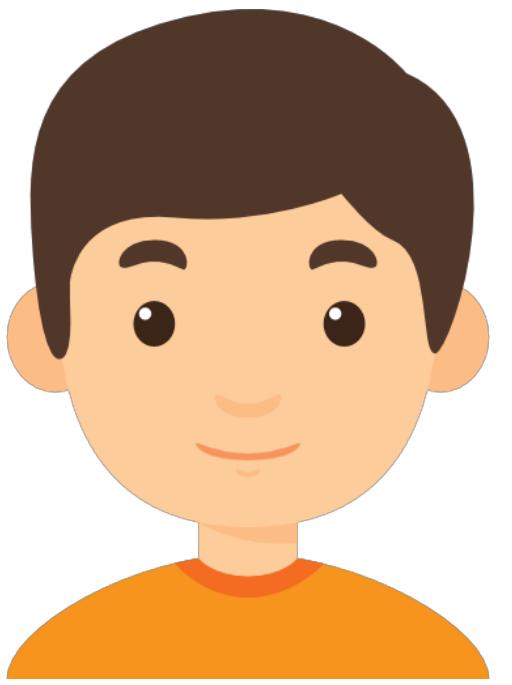


Sit & Wait

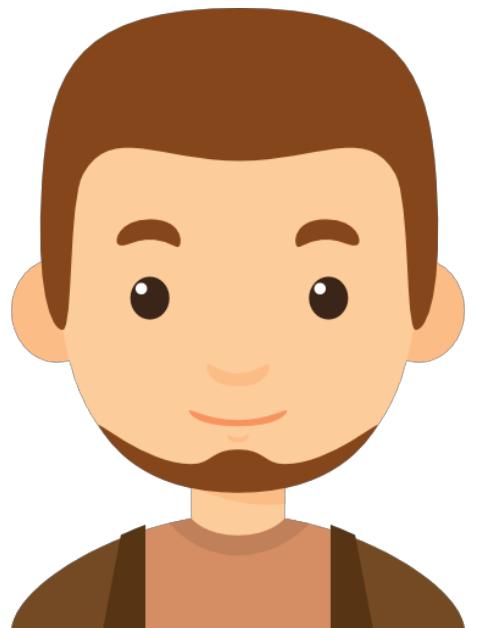
Response



icon from <https://www.freepik.com>



Response



Response



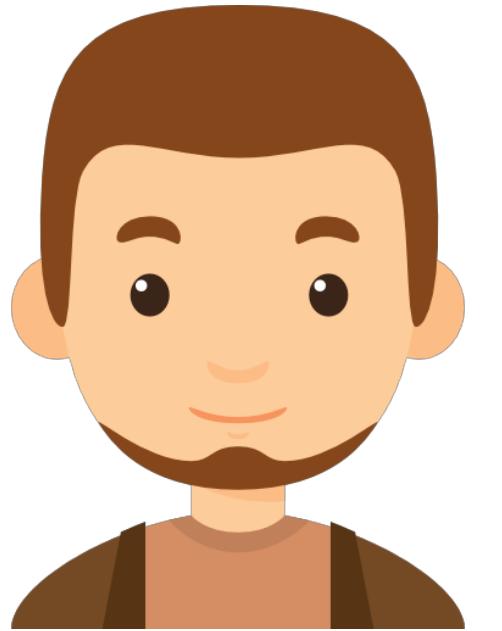
icon from <https://www.freepik.com>



Concurrency in Rails



Request



Request



icon from <https://www.freepik.com>



Sit & Wait



Request

Request





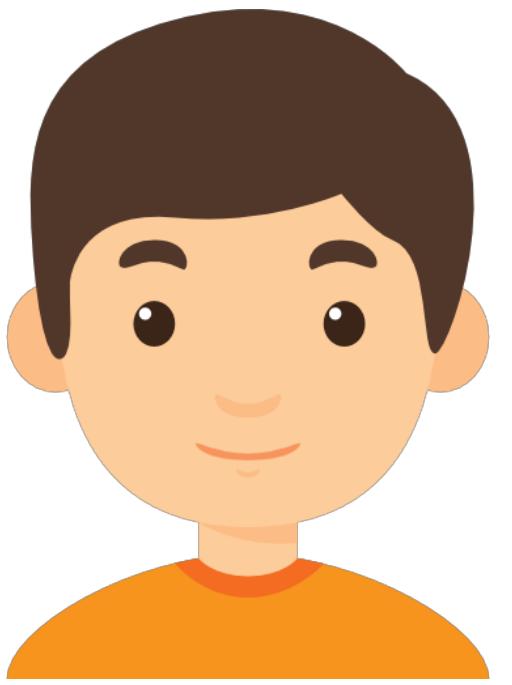
Sit & Wait



Request

Response





Sit & Wait

Response



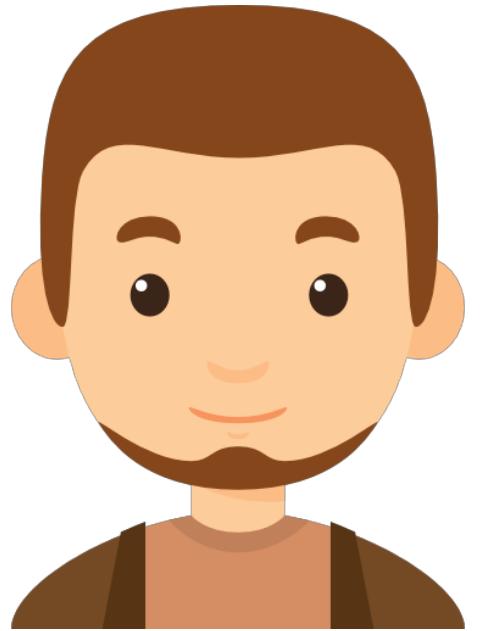
Request



icon from <https://www.freepik.com>



Response

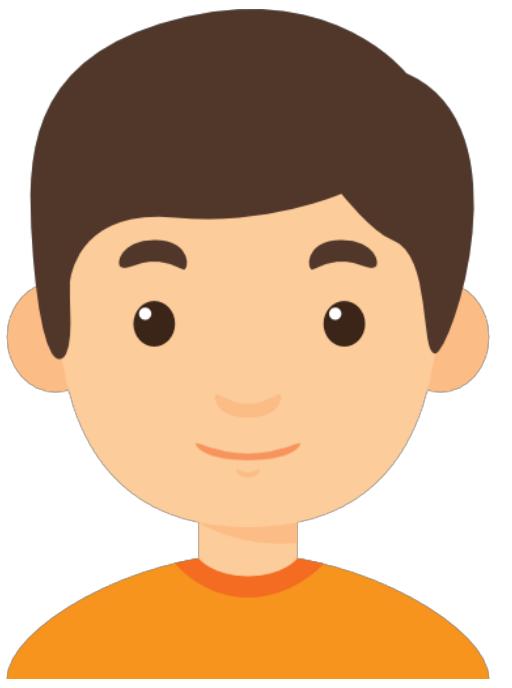


Sit & Wait

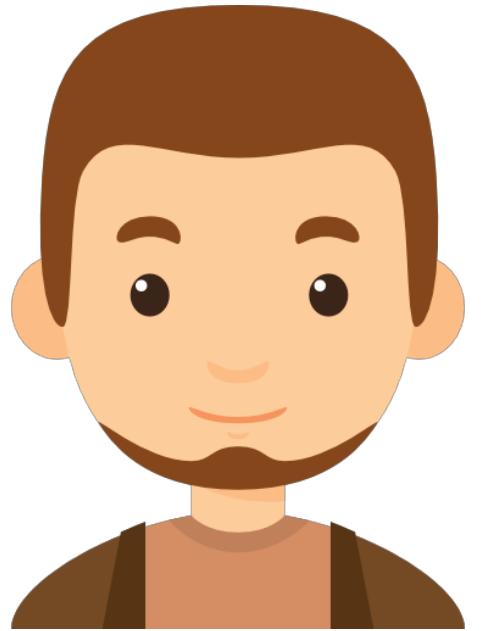
Response



icon from <https://www.freepik.com>



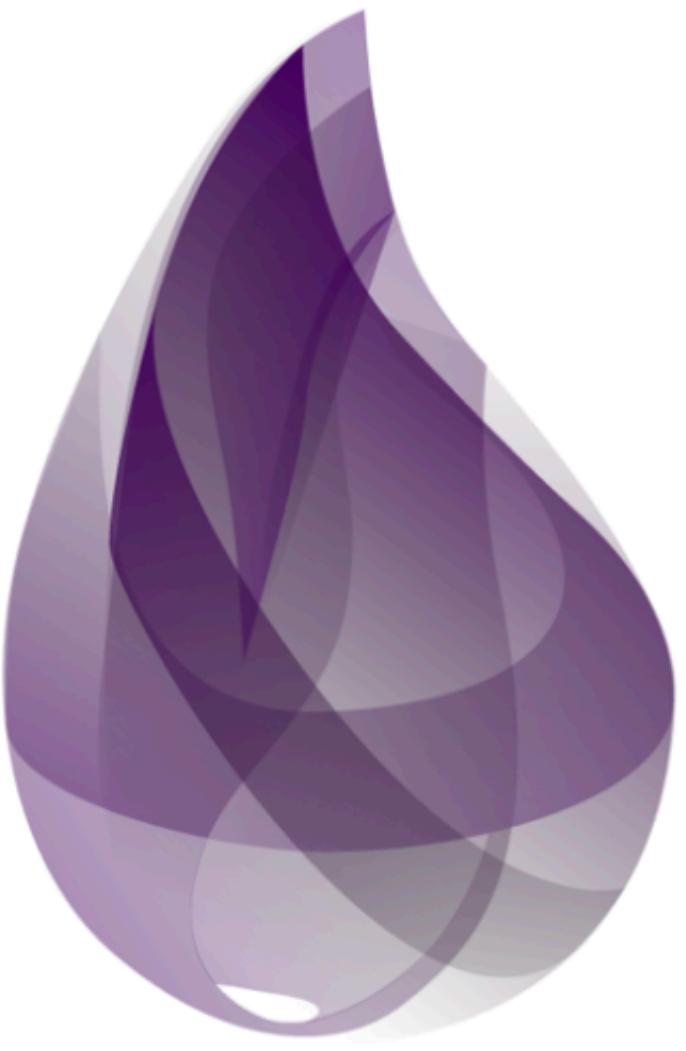
Response



Response



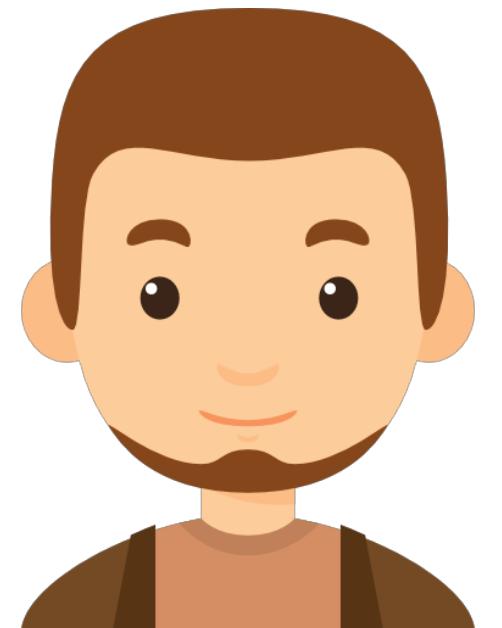
icon from <https://www.freepik.com>



Concurrency in Elixir OTP



Request

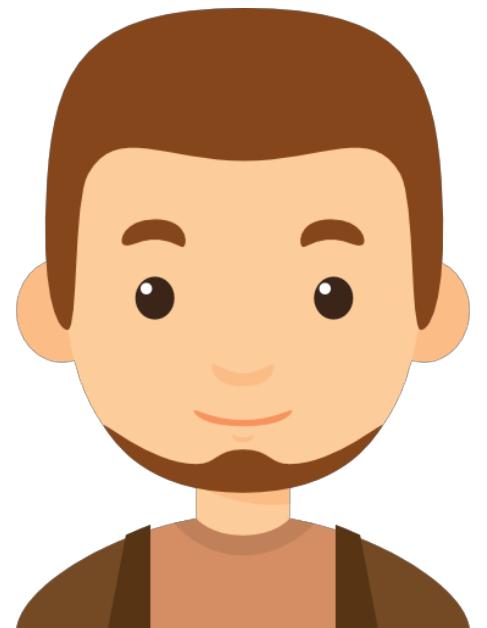


Request





Sit & Wait



Sit & Wait

Request

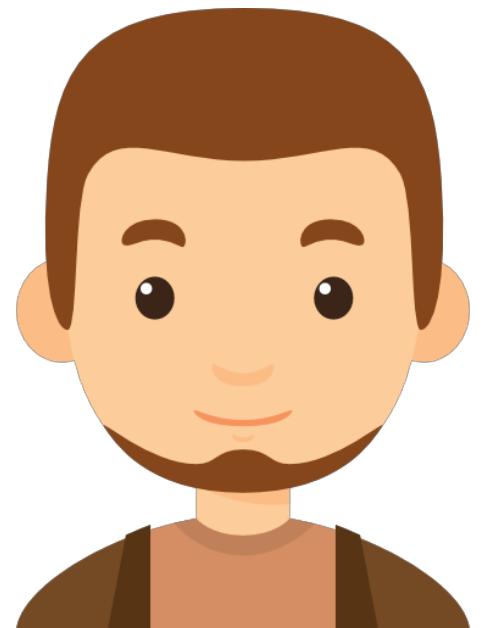


Request





Sit & Wait



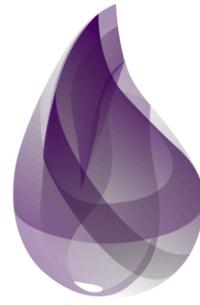
Sit & Wait

Response

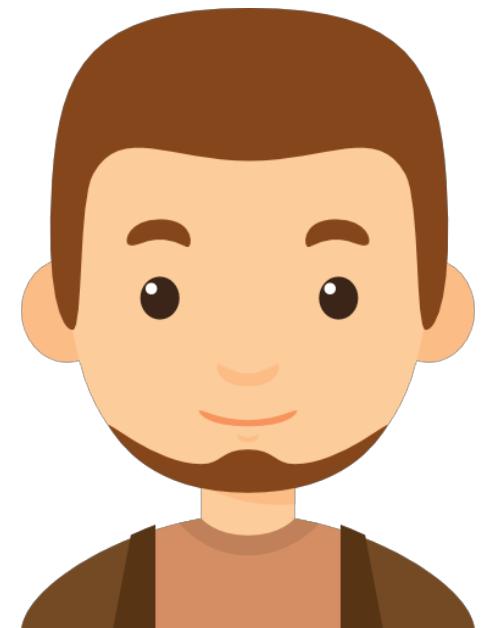


Response





Response

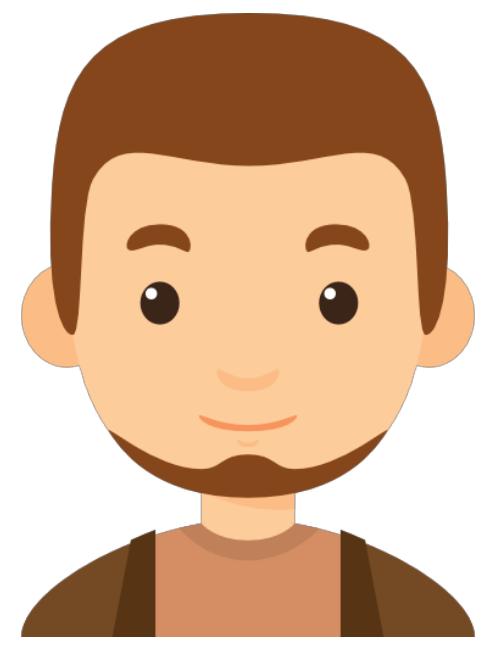


Response





Response



Response

1-10 MB/process



Process



Process



Supervisor

:observer.start

nonode@nohost

System Load Charts Memory Allocators Applications Processes Ports Table Viewer Trace Overview

Pid	Name or Initial Func	Reds	Memory	MsgQ	Current Function
<0.19975.0>	Elixir.FlipServices.OrderStatu...	33778	7637616	0	gen_server:loop/7
<0.122.0>	ssl_manager	1	6665260	0	gen_server:loop/7
<0.314.0>	Elixir.ExchangeGateway.Server	1	6665260	0	gen_server:loop/7
<0.27554.1>	tls_connection:init/1	233910	4349404	0	gen_statem:loop_receive/3
<0.225.0>	Elixir.Phoenix.CodeReloader.S...	1	1945636	0	gen_server:loop/7
<0.21212.2>	tls_connection:init/1	404184	1716288	0	gen_statem:loop_receive/3
<0.42.0>	application_controller	1	1116340	0	gen_server:loop/7
<0.17844.0>	Elixir.ExchangeGateway.Ticker	42446664	1115360	0	gen:do_call/4
<0.20664.0>	erlang:apply/2	437862	1115064	0	observer_pro_wx:table_holder/1
<0.31541.1>	appmon_info	779198	832516	0	gen_server:loop/7
<0.20652.0>	observer_alloc_wx:init/1	45457	689912	0	wx_object:loop/6
<0.489.0>	Elixir.Postgrex.TypeServer:init/1	2	427096	0	gen_server:loop/7
<0.8.0>	erl_prim_loader	1	426524	0	erl_prim_loader:loop/3
<0.109.0>	Elixir.Mix.ProjectStack	1	426524	0	gen_server:loop/7
<0.48.0>	code_server	1	284668	0	code_server:loop/1
<0.20658.0>	observer_app_wx:init/1	2	264076	0	wx_object:loop/6
<0.56.0>	file_server_2	1	197084	0	gen_server:loop/7
<0.372.0>	supervisor:ranch_acceptors_s...	12	147256	0	gen_server:loop/7
<0.20642.0>	observer	38	143240	0	wx_object:loop/6
<0.20647.0>	observer_sys_wx:init/1	2	101524	0	wx_object:loop/6
<0.355.0>	Elixir.FlipServices.Supervisor	11	88760	0	gen_server:loop/7
<0.51.0>	global_name_server	1	55608	0	gen_server:loop/7
<0.313.0>	Elixir.ExchangeGateway.Super...	1	55228	0	gen_server:loop/7
<0.20643.0>	wxe_server:init/1	747092	48472	0	gen_server:loop/7
<0.20858.0>	observer_trace_wx:init/1	2	47320	0	wx_object:loop/6
<0.1.0>	erts_code_purger	1	45112	0	erts_code_purger:wait_for_request/0
<0.475.0>	supervisor:Elixir.Supervisor.D...	2	42848	0	gen_server:loop/7
<0.363.0>	Elixir.FlipServices.PubSub	11	37456	0	gen_server:loop/7
<0.124.0>	tls_connection_sup	1	34644	0	gen_server:loop/7
<0.340.0>	jose_server	1	34532	0	gen_server:loop/7
<0.490.0>	hackney_pool:init/1	14525	34444	0	gen_server:loop/7
<0.53.0>	erlang:apply/2	1	29468	0	global:loop_the_registrar/0
<0.356.0>	Elixir.FlipServicesWeb.Endpoint	11	26716	0	gen_server:loop/7

Number of Processes: 318

- + Functional
- + Concurrency
- + Beautiful Syntax
- + Development Exp.
- + Fault tolerance
- + All-in-one Ecosystem
- + Documentations



- Small Community
- Learning Resources
- Erlang Learning



Paradigm Shift

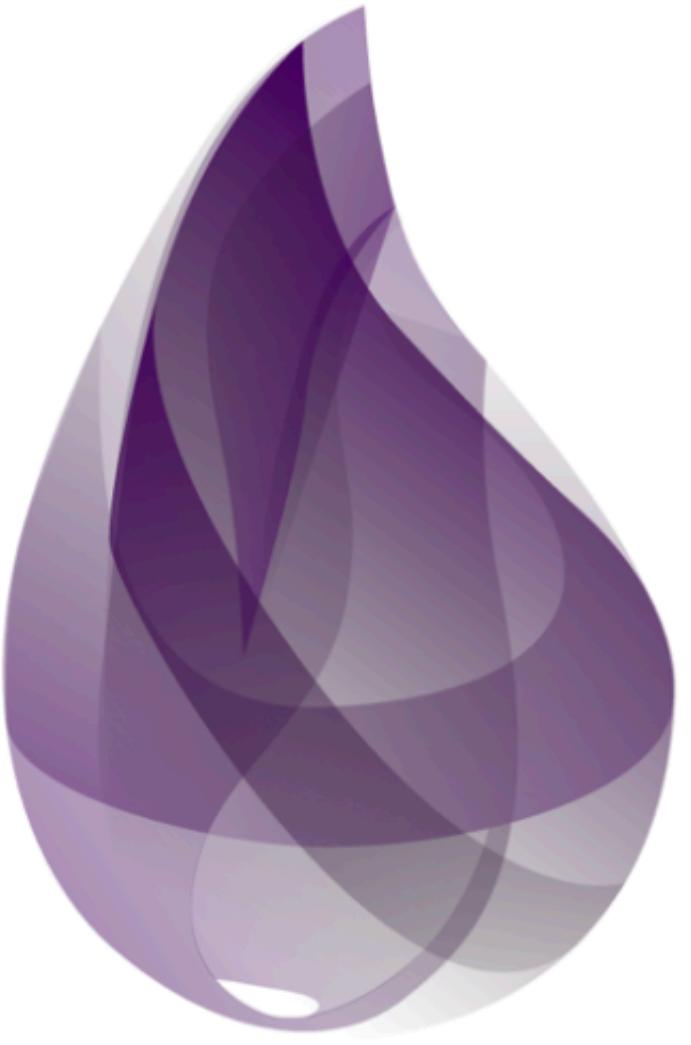


Elixir in Discord (2017)



5,000,000 concurrent users
1,000,000+ events/second





Functional Programming

OOP (JS)

```
const tank = new Tank({ position: 0 })
tank.move(50) // Tank{ position: 50 }
tank.position // 50
```

Functional & Immutable (Elixir)

```
tank = Tank.new(position: 0)
Tank.move(tank, 50) # %Tank{ position: 50 }
tank.position # 0
```



Turn left 90° → Move 200 → Fire





Turn left 90° → Move 200 → Fire

```
tank = Tank.fire(  
    Tank.move(  
        Tank.turn_left(tank, 90),  
        200  
    )  
)
```



Turn left 90° → Move 200 → Fire

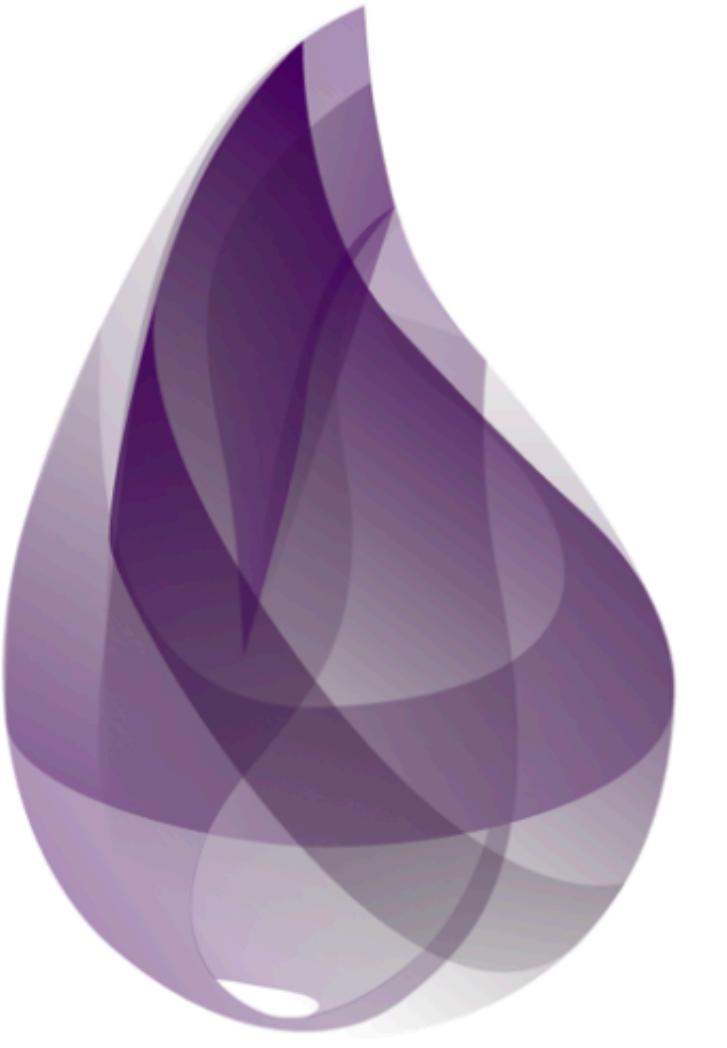
```
tank = Tank.turn_left(tank, 90)
tank = Tank.move(tank, 200)
tank = Tank.fire(tank)
```



Turn left 90° → Move 200 → Fire

```
tank
|> Tank.turn_left(90)
|> Tank.move(200)
|> Tank.fire()
```

Pipe Operator ❤️



Pattern Matching

= is actually pattern matching

```
iex> 1 = 1  
1
```

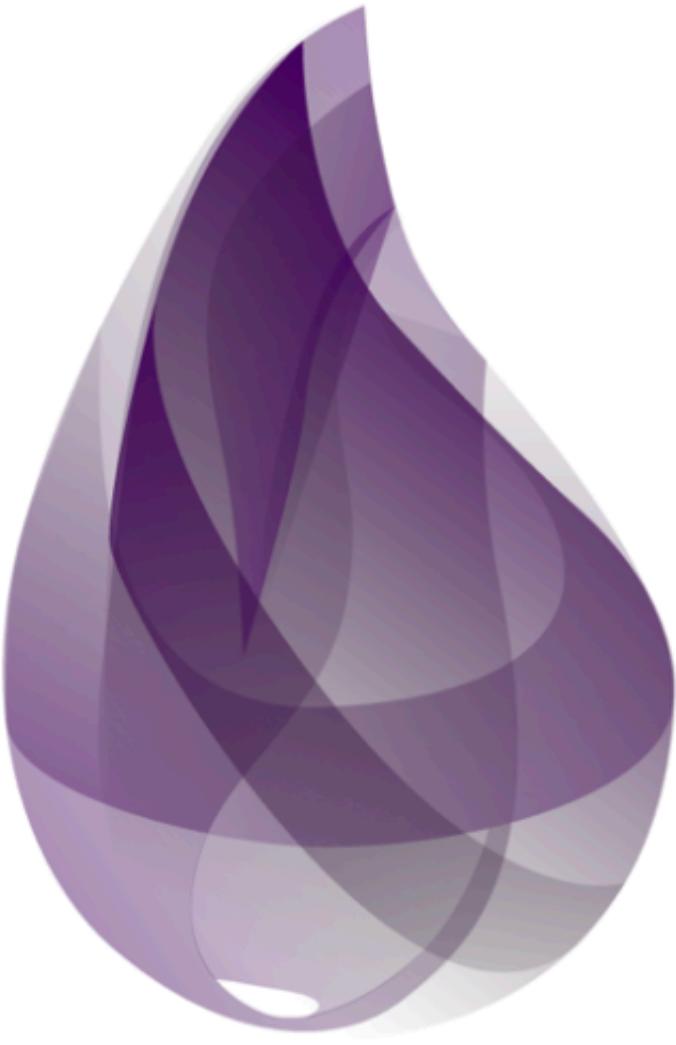
```
iex> [a, b] = [1, 2]  
[1, 2]
```

```
iex> [a, b, c] = [1, 2]  
** (MatchError)
```

```
iex> [first | rest] = [1,2,3,4]  
[1, 2, 3, 4]
```

```
iex> first  
1
```

```
iex> rest  
[2, 3, 4]
```



**New Ways of Solving
Problems Elegantly**

Develop lover match function



```
match(my_feeling, target_feeling)
```

my_feeling	target_feeling	output
“like”	“like”	“Yey, It’s a match!”
“like”	“dislike”	“Please give up”
“dislike”	“like”	“Passed”
“dislike”	“dislike”	“Passed”

JavaScript

```
function match(me, target) {  
    if (me === 'like') {  
  
    }  
    else if (me === 'dislike') {  
  
    }  
    else {  
  
    }  
}
```

JavaScript

```
function match(me, target) {  
    if (me === 'like') {  
        if (target === 'like') {  
            return 'Yey, It's a match!'  
        }  
        else {  
            return 'Please give up'  
        }  
    }  
    else if (me === 'dislike') {  
        return 'Passed'  
    }  
    else {  
        return false  
    }  
}
```

Elixir

```
def match(me, target) do
  case {me, target} do
    end
  end
```

Elixir

```
def match(me, target) do
  case {me, target} do
    {:like, :like} -> "Yey, you found a match"
    {:like, :dislike} -> "Please give up"
    {:dislike, _} -> "Passed"

  end
end
```

Elixir

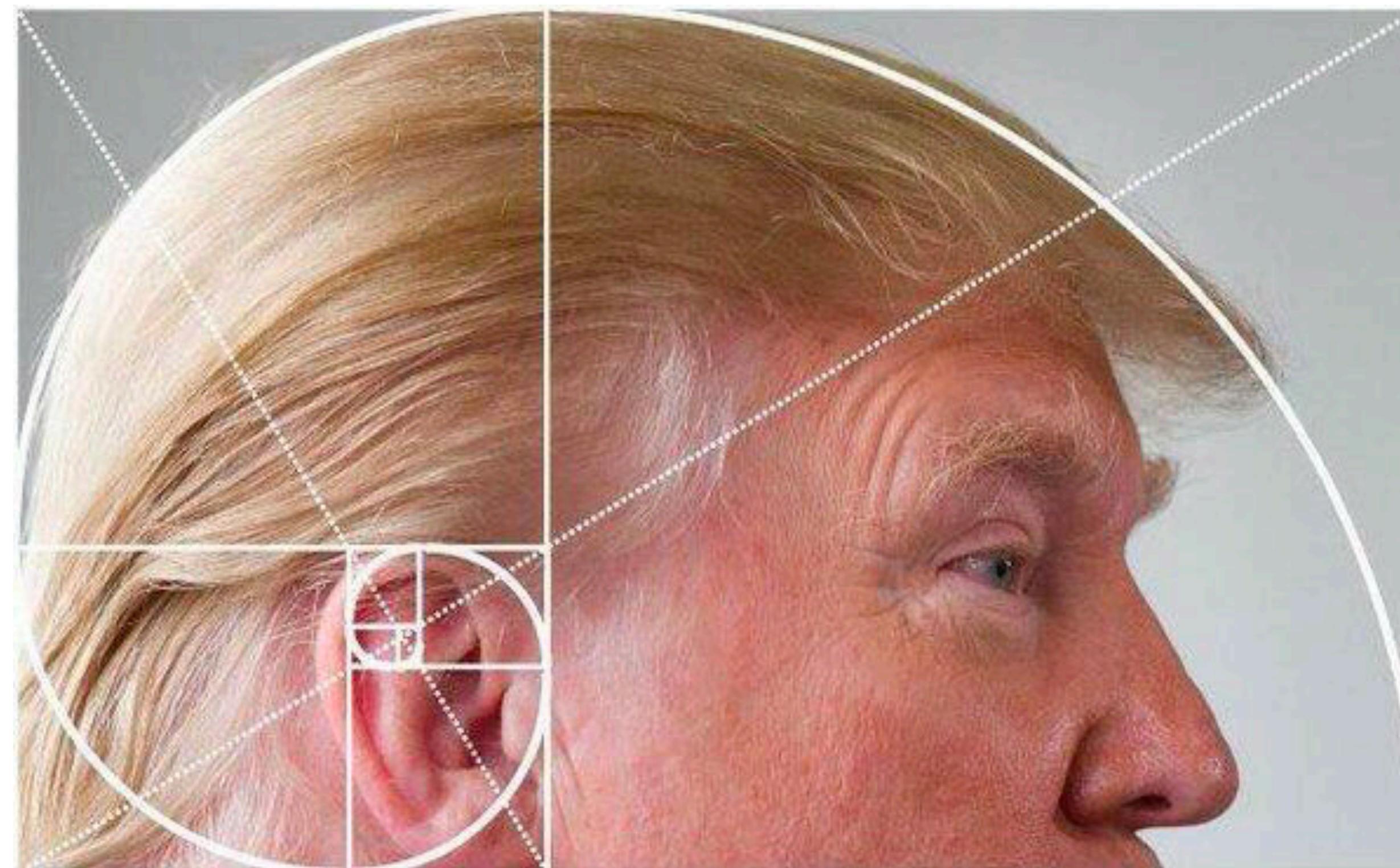
```
def match(me, target) do
  case {me, target} do
    {:like, :like} -> "Yey, you found a match"
    {:like, :dislike} -> "Please give up"
    {:dislike, _} -> "Passed"
    _ -> nil
  end
end
```

Elixir (Alternative)

```
def match(:like, :like), do: "Yey, you found a match"  
def match(:like, :dislike), do: "Please give up"  
def match(:dislike, _), do: "Passed"  
def match(_, _), do: nil
```

Develop Fibonacci

0, 1, 1, 2, 3, 5, 8, 13, ...



Fibonacci

```
def fibonacci(0), do: 0
def fibonacci(1), do: 1
def fibonacci(2), do: 1
def fibonacci(3), do: 2
```

Fibonacci

```
def fibonacci(0), do: 0
def fibonacci(1), do: 1

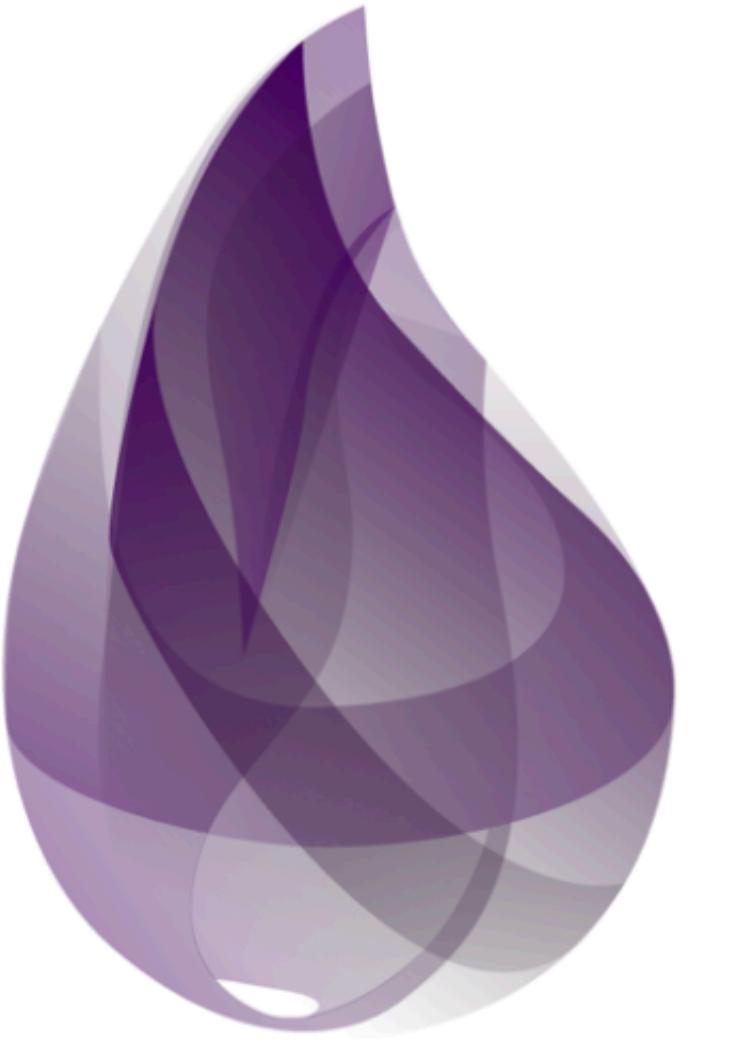
def fibonacci(n) do
  fibonacci(n - 1) + fibonacci(n - 2)
end
```

Fibonacci

```
def fibonacci(n) when n < 2, do: n  
  
def fibonacci(n) do  
  fibonacci(n - 1) + fibonacci(n - 2)  
end
```

Fibonacci (JS Functional)

```
function fibonucci(n) {  
    if (n < 2) return n  
  
    return fibonucci(n - 1) + fibonucci(n - 2)  
}
```



Testing

Built-in BDD testing

```
defmodule FibonacciTest do
  use ExUnit.Case

  describe "fibonacci/1" do
    test "return 13 for input 7" do
      assert Fibonacci.fibonacci(7) == 13
    end
  end
end
```

Doctest

```
@doc """
## Examples

iex> Fibonacci.fibonacci(7)
13

"""
def fibonacci(n) do
  ...
end
```

Powerful assert

```
object = %{a: 2, b: 3.9}

# Normal assert
assert object == %{a: 2, b: 3.9}

# Pattern matching assert
assert object = %{a: 2}
```

Tada! ✨ auto generated doc with `mix docs`

elixir_casually_intro

v0.1.0



Search

PAGES

MODULES

ElixirCasuallyIntro

Fibonacci

Top

Summary

+ Functions

Tinder

Summary

Functions

fibonacci(n)

Examples

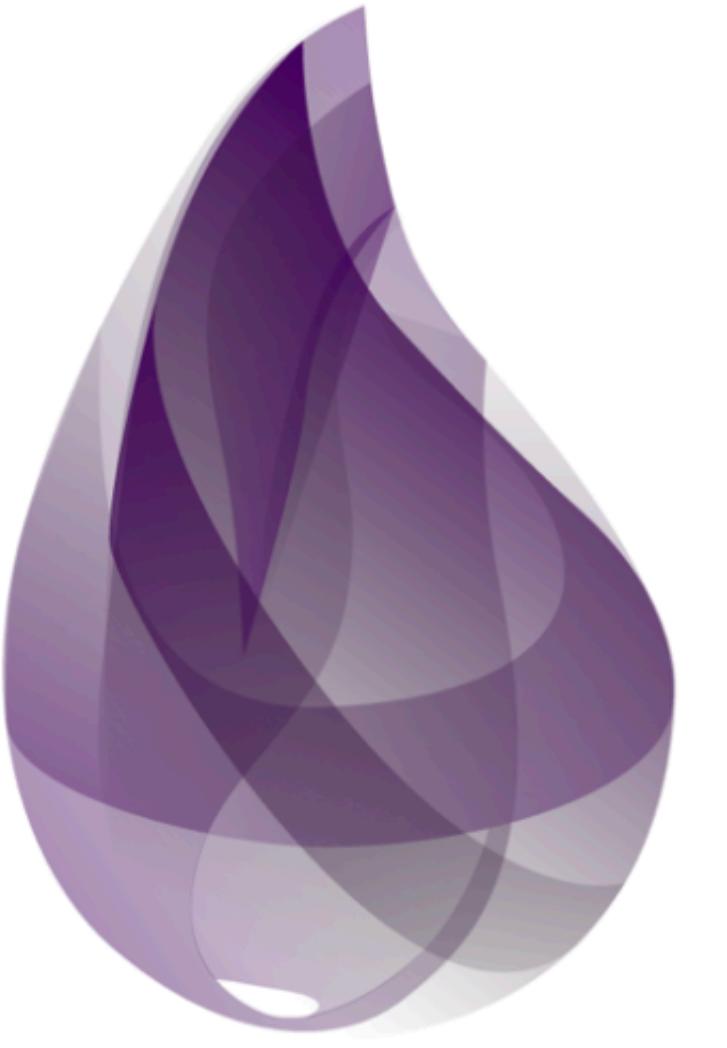
```
iex> Fibonacci.fibonacci(7)  
13
```

Functions

fibonacci(n)

Examples

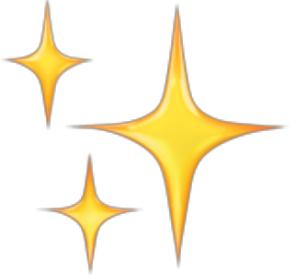
```
iex> Fibonacci.fibonacci(7)  
13  
  
iex> Fibonacci.fibonacci(2)  
1
```

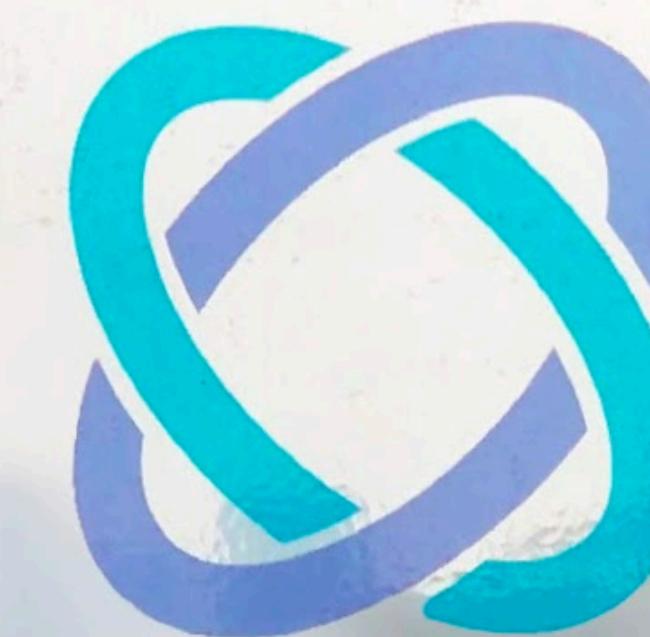


All-in-one Ecosystem

Work	Tools	Built-in?
Package Manager	Mix	✓
Testing	ExUnit	✓
Worker	OTP	✓
Concurrency	OTP	✓
Caching	ETS	✓
Database	Ecto	
Deployment	Distrillary	
Web Framework	Phoenix	
WebSocket	Phoenix Channel	
Documentation	Hex	

Learn to solve the real
world problems better





flipay

is hiring!



@turboza