About me

* Co-organizo React Medellín y ReactConf Colombia 2020.
* Frontend @Zapier
* Open source
* Elixir / ReasonML
* @alejandronanez

Vamos a aprender un nuevo lenguaje

Tipado

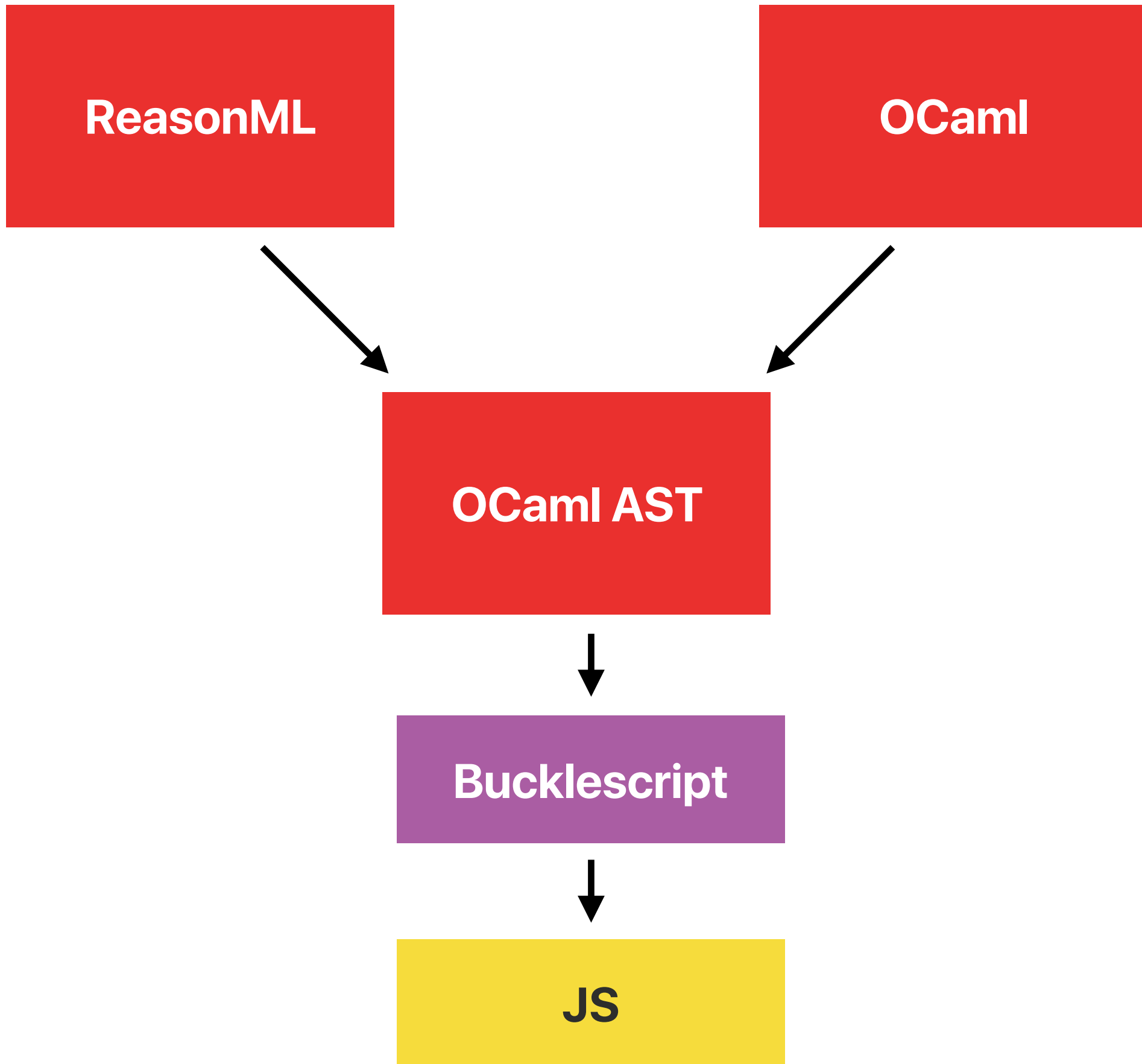# Tooling

Compilador
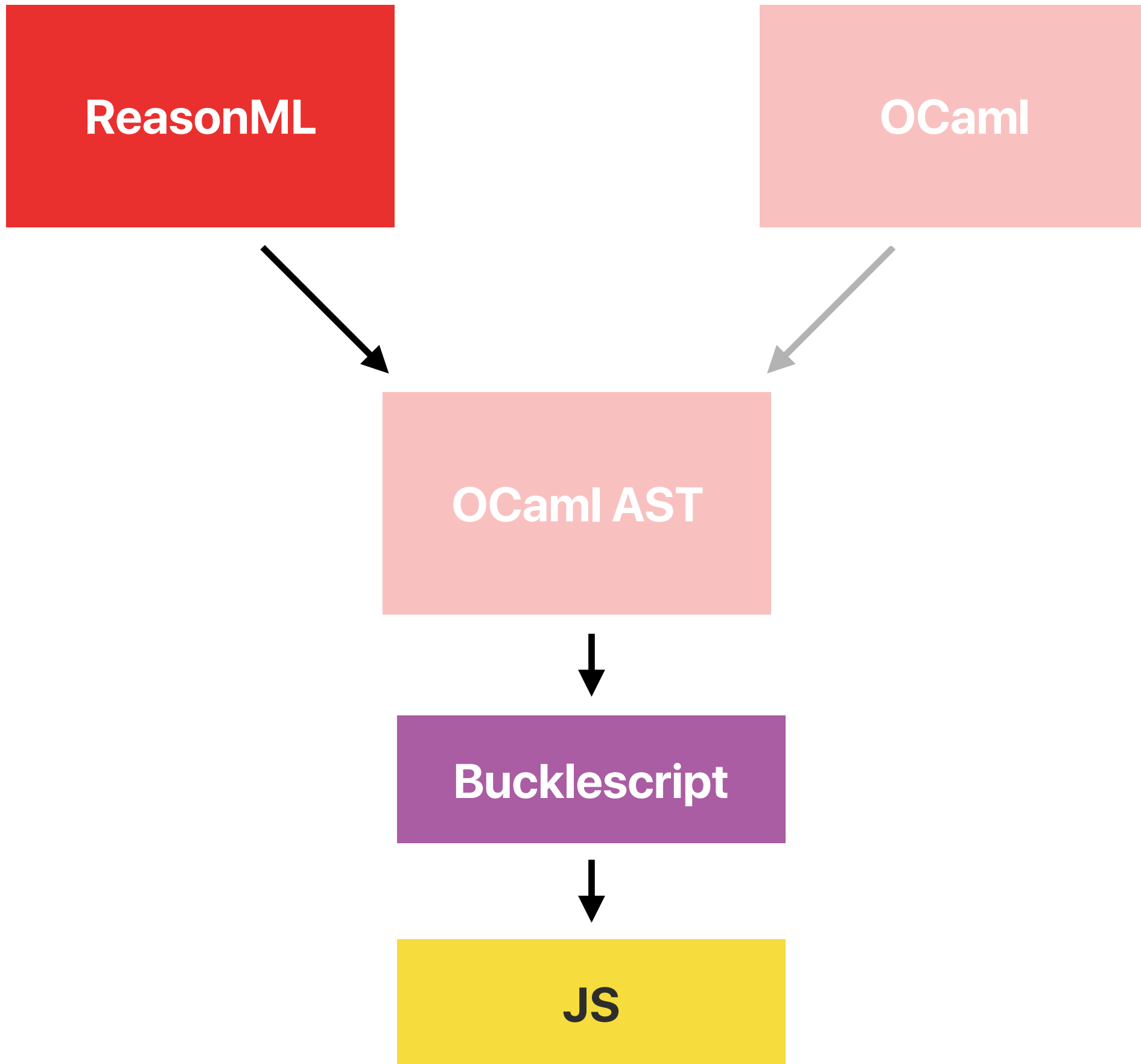
Sintaxis MUY parecida a Javascript

```
let edad = 32;
```

```
let getName = (name) => "Welcome " ++ name;

getName("React MDE");
```

https://sketch.sh/s/1pkUiYltBIm3g1PZrYeH3F/

```
let coworkingsList = [
  "Factoria",
  "Seedspace",
  "WeWork"
];
```

https://sketch.sh/s/1pkUiYltBIm3g1PZrYeH3F/

```
let coworkingsArray= [|
   "Factoria",
   "Seedspace",
   "WeWork"
|];
```

https://sketch.sh/s/1pkUiYltBIm3g1PZrYeH3F/

```
if (true) {
  Js.log("Hola!");
};
```

https://sketch.sh/s/1pkUiYltBIm3g1PZrYeH3F/

@alejandronanez

ReasonML

OCaml

OCaml AST

Bucklescript

JS

@alejandronanez

Records

```
let meetup = {nombre: "React Medellin", edad: 2};
```

```
35   let meetup = {nombre: "React Medellin", edad: 2};
```

❌ Index.re 1 of 1 problem

Error: Unbound record field nombre

https://sketch.sh/s/1pkUiYltBIm3g1PZrYeH3F/

```
let meetup = {nombre: "React Medellin", edad: 2};
```

```
type meetup = {
  nombre: string,
  edad: int,
};


let meetup = {nombre: "React Medellin", edad: 2};
```

https://sketch.sh/s/1pkUiYltBIm3g1PZrYeH3F/

Optional

null y undefined **no existen** en Reason

🚨

undefined is not a function
cannot access foo **of** **undefined**

@alejandronanez

```
type option('a) = None | Some('a)
```

```
let licenseNumber = Some("123456");
```

https://sketch.sh/s/1pkUiYltBIm3g1PZrYeH3F/

```
let licenseNumber =
  if (licenseNumber) {
    Some(5);
  } else {
    None;
  };
```

https://sketch.sh/s/1pkUiYltBIm3g1PZrYeH3F/

```
let licenseNumber = Some("123456");

let result = switch (licenseNumber) {
| None =>"No tengo licencia"
| Some(number) =>
  "Tengo licencia # " ++ number
};
```

https://sketch.sh/s/1pkUiYltBIm3g1PZrYeH3F/

Variants

```javascript
// Javascript
const ABIERTO = "ABIERTO";
const CERRADO = "CERRADO";

const coworking = ABIERTO;
```

```
type estado =
    | Abierto
    | Cerrado;

let coworking = Abierto;
```

Javascript

```javascript
const state = {
  loading: true,
  error: false,
  data: {}
};
```

Javascript

```
state.loading // true
state.error   // true
state.data    // {}


state.loading // false
state.error   // true
state.data    // { ... }


state.loading // true
state.error   // false
state.data    // { ... }
```

```
type data = {coworkings: list(string)};

type state =
    | Loading
    | Error
    | Success(data);
```

https://sketch.sh/s/1pkUiYltBIm3g1PZrYeH3F/

Pattern Matching

```
switch (true) {
| true => "verdadero"
| false => "falso"
| _ => "No se :("
};
```

https://sketch.sh/s/1pkUiYItBIm3g1PZrYeH3F/

```
let condition = Success({coworkings: ["workia"]});

let resultado =
  switch (condition) {
  | Loading => "Esta cargando"
  | Error => "Erro :("
  | Success(data) =>
    "Hay " ++ string_of_int(Belt.List.length(data.coworkings)) ++ " datos"
  };
```

https://sketch.sh/s/1pkUiYItBIm3g1PZrYeH3F/

Functions

```
let getName = (name) => "Welcome " ++ name;

getName("React MDE");
```

```
let name = (~firstName, ~lastName) => firstName ++ " " ++ lastName;

name(~firstName="React", ~lastName="Medellin")

name(~lastName="Medellin", ~firstName="React")
```

@alejandronanez

Pipe & Fast Pipe

```
let fullName = (firstName, middleName, lastName) =>
  firstName ++ " " ++ middleName ++ " " ++ lastName;

/** Alejandro Nanez Ortiz */
let normal = fullName("Alejandro", "Nanez", "Ortiz");

/** Alejandro Nanez Ortiz */
let fastPipe = "Alejandro"->fullName("Nanez", "Ortiz");

/** Alejandro Nanez Ortiz */
let pipe = "Ortiz" |> fullName("Alejandro", "Nanez");
```

@alejandronanez

Módulos

Todos los archivos son módulos

```
/** Greeting.re */
[@react.component]
let make = () =>
  <div>{React.string("Hey")}</div>;

let nombre = "ReactMDE";
```

```
/** OtroArchivo.re */
<Greeting />

let valor = Greeting.name;
```

Puedes crear módulos dentro de los archivos

```
/** Greeting.re */
module ModuloA = {
  let name = "ReactMDE";
};
```

`/** OtroArchivo.re */`
`Greeting.ModuloA.name`

ReasonReact

```
<div>
    React.string("ReactMDE")
</div>
```

```reason
/** App.re */
[@react.component]
let make = () => <div> {React.string("Hola")} </div>;
```

@alejandronanez

```
/** Index.re */
ReactDOMRe.renderToElementWithId(<App />, "app");
```

Prop

```reason
/** App.re */
[@react.component]
let make = (~mensaje) =>
  <div> {React.string(mensaje)} </div>;
```

@alejandronanez

```jsx
<App mensaje="ReactMDE" />
```

Prop opcional

```reason
/** App.re */
[@react.component]
let make = (~mensaje="default") =>
  <div> {React.string(mensaje)} </div>;
```

@alejandronanez

<App />

Gracias :)

@alejandronanez