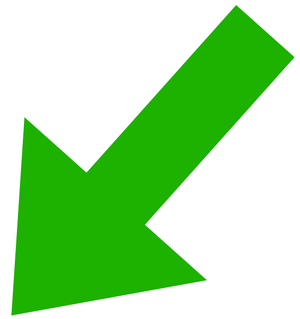

REASON

ReasonReact

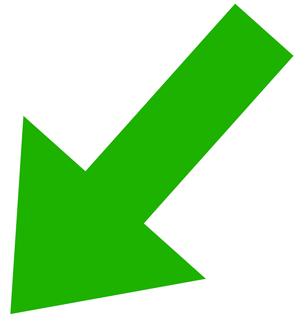
useState

```
[@react.component]
let make = () => {
  let (count, setCount) = React.useState(() => 0);

  <div>
    <p> {React.string("Clicked " ++ string_of_int(count) ++ " times")} </p>
    <button onClick={_event => setCount(count => count + 1)}>
      {React.string("Click me")}
    </button>
  </div>;
};
```

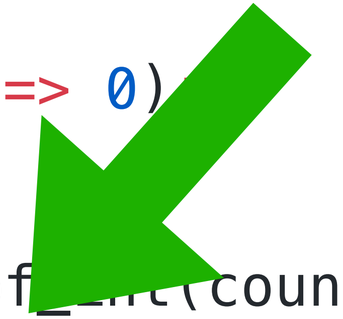


```
[@react.component]
let make = () => {
  let (count, setCount) = React.useState(() => 0);

  <div>
    <p> {React.string("Clicked " ++ string_of_int(count) ++ " times")} </p>
    <button onClick={_event => setCount(count => count + 1)}>
      {React.string("Click me")}
    </button>
  </div>;
};
```

```
[@react.component]
let make = () => {
  let (count, setCount) = React.useState(() => 0);

  <div>
    <p> {React.string("Clicked " ++ string_of_int(count) ++ " times")} </p>
    <button onClick={_event => setCount(count => count + 1)}>
      {React.string("Click me")}
    </button>
  </div>;
};
```

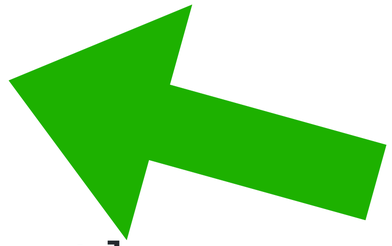


```
[@react.component]
let make = () => {
  let (count, setCount) = React.useState(() => 0);

  <div>
    <p> {React.string("Clicked " ++ string_of_int(count) ++ " times")} </p>
    <button onClick={_event => setCount(count => count + 1)}>
      {React.string("Click me")}
    </button>
  </div>;
};
```

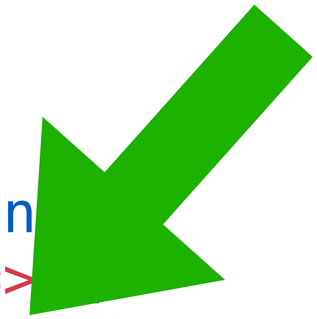


useReducer

```
type action =
  | Increment
  | Reset;

[@react.component]
let make = () => {
  let (state, send) =
    React.useReducer(
      (state, action) =>
        switch (action) {
        | Increment => state + 1
        | Reset => 0
        },
      0,
    );

  <div>
    <p> {React.string("Counter:  " ++ string_of_int(state))} </p>
    <button onClick={_event => send(Increment)}>
      {React.string("Increment")}
    </button>
```



```
type action =
  | Increment
  | Reset;

[@react.component]
let make = () => {
  let (state, send) =
    React.useReducer(
      (state, action) =>
        switch (action) {
        | Increment => state + 1
        | Reset => 0
        },
      0,
    );

  <div>
    <p> {React.string("Counter:  " ++ string_of_int(state))} </p>
    <button onClick={_event => send(Increment)}>
      {React.string("Increment")}
    </button>
```



```
type action =
  | Increment
  | Reset;

[@react.componen
let make = () =>
  let (state, send) =
    React.useReducer(
      (state, action) =>
        switch (action) {
        | Increment => state + 1
        | Reset => 0
        },
      0,
    );

  <div>
    <p> {React.string("Counter:  " ++ string_of_int(state))} </p>
    <button onClick={_event => send(Increment)}>
      {React.string("Increment")}
    </button>
```

```
let make = () => {
  let (state, send) =
    React.useReducer(
      (state, action) =>
        switch (action) {
        | Increment => state + 1
        | Reset => 0
        },
      0,
    );

  <div>
    <p> {React.string("Counter:  " ++ string_of_int(state))} </p>
    <button onClick={_event => send(Increment)}>
      {React.string("Increment")}
    </button>
    <button onClick={_event => send(Reset)}>
      {React.string("Reset")}
    </button>
  </div>;
};
```

The End