



REACT

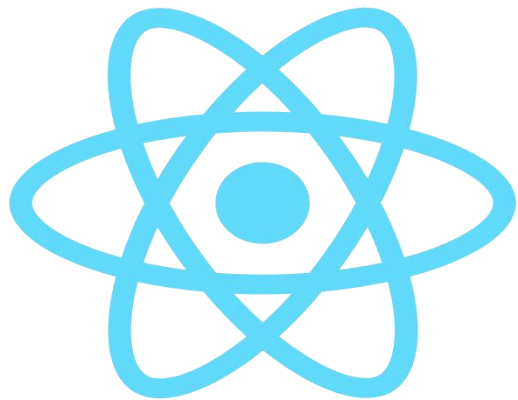
Una libreria para gobernarlos a todos...

Ing.Alexis Eduardo Matamoros

SPA

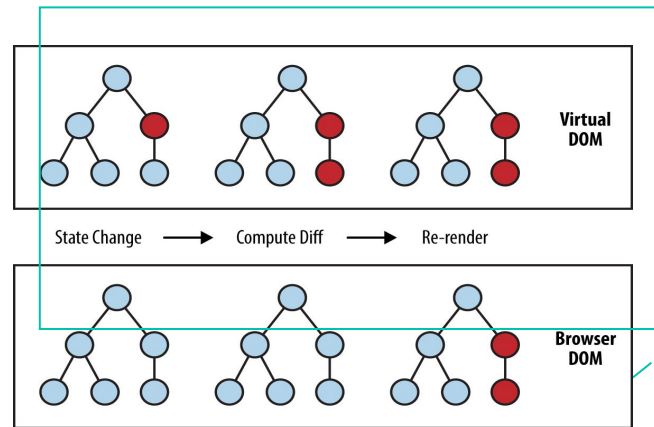


MPA



REACT

ReactJS es una de las librerías más populares de JavaScript para el desarrollo de aplicaciones móviles y web.



Ventajas

Multiplataforma

DOM Virtual

Facil de Aprender

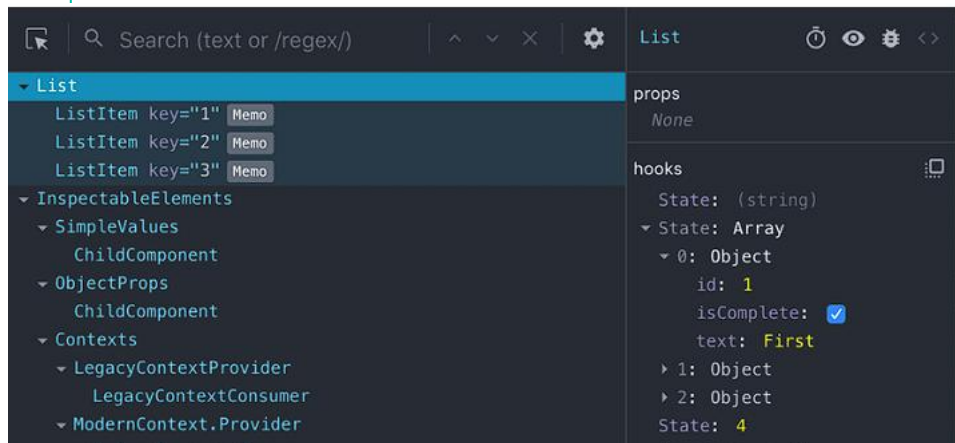
Herramientas

Que necesitamos para iniciar una app con

REACT...

Crear una app sin Framework...

ReactDevTools



JSX

Se llama JSX, y es una extensión de la sintaxis de JavaScript. Recomendamos usarlo con React para describir cómo debería ser la interfaz de usuario. JSX puede recordarte a un lenguaje de plantillas, pero viene con todo el poder de JavaScript.

```
const name = 'Josh Perez';  
const element = <h1>Hello, {name}</h1>;
```

```
function getGreeting(user) {  
  if (user) {  
    return <h1>Hello, {formatName(user)}!</h1>;  
  }  
  return <h1>Hello, Stranger.</h1>;  
}
```

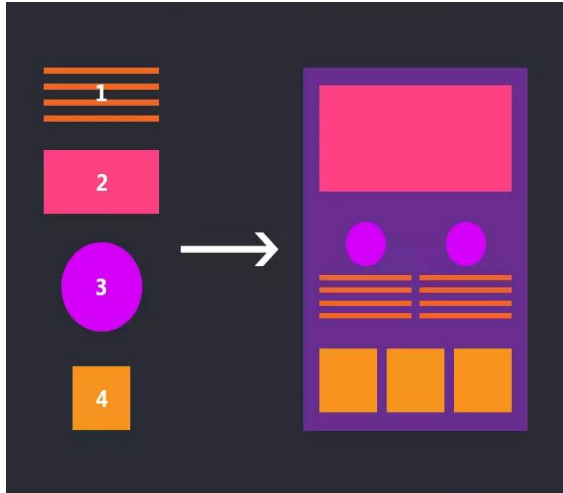
```
const element = (  
  <div>  
    <h1>Hello!</h1>  
    <h2>Good to see you here.</h2>  
  </div>  
)
```

```
import React from 'react'
import ReactDOM from 'react-dom/client'
import App from './App.jsx'
import './index.css'

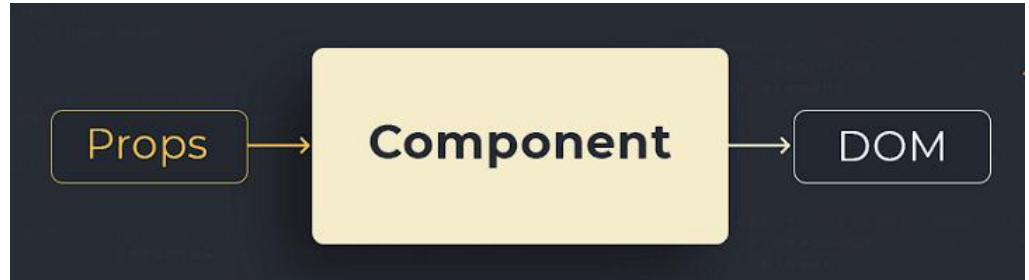
ReactDOM.createRoot(document.getElementById('root')).render(
  <React.StrictMode>
    <App />
  </React.StrictMode>,
)
```

ROOT

Crea tu primera aplicacion



Componentes



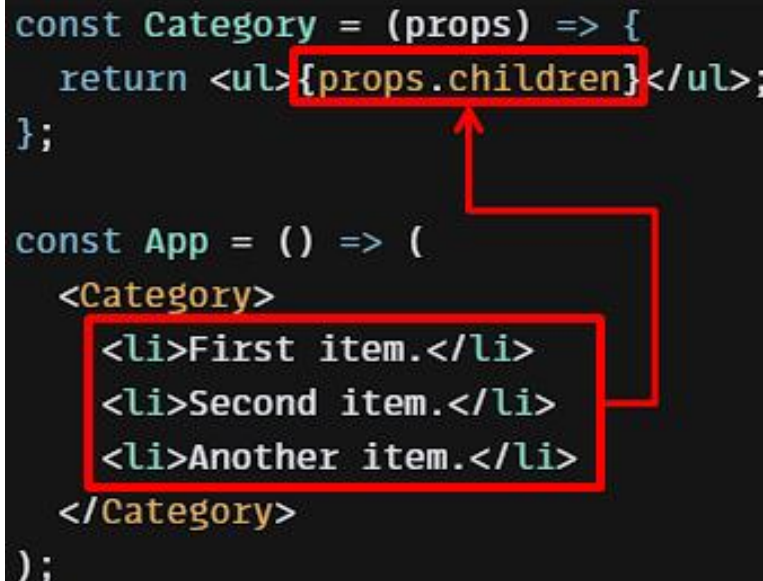
Props

Props

```
const Greet = ({firstName, lastName}) => {  
  return (<div>{`Hi ${firstName} ${lastName}`}!</div>)  
}  
  
Greet.defaultProps = {  
  firstName: 'John',  
  lastName: 'Doe'  
};  
  
Greet.propTypes = {  
  firstName: PropTypes.string,  
  lastName: PropTypes.string  
};
```


Children

```
const Category = (props) => {  
  return <ul>{props.children}</ul>;  
};  
  
const App = () => (  
  <Category>  
    <li>First item.</li>  
    <li>Second item.</li>  
    <li>Another item.</li>  
  </Category>  
)
```



The diagram illustrates the flow of children from the `App` component to the `Category` component. A red box highlights the `First item.`, `Second item.`, and `Another item.` elements within the `<Category>` component. A red arrow points from this box to the `{props.children}` expression in the `Category` component's return statement, indicating that these elements are passed as children to the `Category` component.

Renderizado Condicional

```
function Item({ name, isPacked }) {  
  return (  
    <li className="item">  
      {isPacked ? (  
        <del>  
          {name + ' ✓'}  
        </del>  
      ) : (  
        name  
      )}  
    </li>  
  );  
}
```

```
function Item({ name, isPacked }) {  
  if (isPacked) {  
    return <li className="item">{name} ✓</li>;  
  }  
  return <li className="item">{name}</li>;  
}
```

```
function Item({ name, isPacked }) {  
  return (  
    <li className="item">  
      {name} {isPacked && '✓'}  
    </li>  
  );  
}
```

Renderizando un listado Condicional

```
const listItems = people.map(person => <li>{person}</li>);
```



Gracias

Preguntas???