**Para saber mais: componentes funcionais**

Durante o curso, nós vamos trabalhar apenas com componentes funcionais. Essa é a forma mais simples de criar um componente React.

Mas ela não é a única forma possível. Podemos criar componentes utilizando [Classes do ES6](https://cursos.alura.com.br/course/javascript-orientacao-a-objetos).

Vamos entender um pouco mais as diferenças?

Aqui vai um componente funcional:

**function** **BoasVindas**(props) {

**return** <h1>Olá, {props.nome}</h1>;

}

O mesmo componente, baseado em classes, seria:

**class** **BoasVindas** **extends** **React.Component** {

**render**() {

**return** <h1>Olá, {this.props.nome}</h1>;

}

}

Além da sintaxe, existem diversas diferenças. Se você quiser ir mais fundo no conceito de componentes baseado em classes, a [documentação](https://pt-br.react.dev/reference/react/Component#defining-a-class-component) é a nossa melhor fonte.

# Para saber mais: Node.JS

Sabemos que mergulhos mais fundos vem com o tempo, mas nesse momento é muito importante entendermos o que é o Node.JS.

Tanto o Node quanto o NPM vão fazer parte do nosso dia a dia enquanto pessoas desenvolvedoras frontend.

Para compreender melhor o que é cada coisa, eu separei pra você [esse vídeo com o Paulo e o Mário Souto](https://www.hipsters.tech/o-que-e-node-js/) que trata justamente desse tema.

Lembrando que em caso de qualquer dúvida e/ou dificuldade você pode sempre recorrer ao nosso fórum e receber aquela ajudinha camarada dos nossos experts e de outros estudantes.

Bons estudos!