

## Você entendeu a estrutura do projeto?

Quando o Vue CLI levanta nosso servidor, automaticamente o navegador padrão do sistema operacional é aberto no endereço `localhost:8080`. Este endereço exibe o conteúdo de `index.html`, única página com a extensão `.html` do projeto gerado.

No entanto, seu conteúdo não condiz com o que é exibido para nós no navegador. A respeito disso, temos as seguintes afirmativas.

**A)** Todos os arquivos da pasta `src` são transformados em um bundle chamado `build.js` e este é carregado pelo `index.html`. Inclusive o script do Vue.js faz parte desse arquivo. Então, o próprio Vue.js se encarregará de exibir o conteúdo em `index.html` interpretando as informações dentro do bundle.

**B)** Em ambiente de desenvolvimento, o endereço `/dist/build.js` aponta para um endereço no servidor que aponta para um arquivo criado em memória. Esse processo agiliza o tempo de alteração do `build.js` a cada alteração, melhorando assim sua experiência de desenvolvimento.

**C)** Durante a geração de `build.js`, Babel é o responsável por empacotar todos os arquivos da pasta `src` do nosso projeto e Webpack atua como transcompilador de ES2015 (ES6) para ES5, garantindo assim maior compatibilidade do nosso código.

**D)** Declaramos componentes em Single File Templates. Esses arquivos nada mais são do que módulos que exportam componentes.

**E)** A estrutura padrão de um componente é composta pelas tags `template`, `script` e `style`.

**A**

B e C são falsas

**B**

C e D são falsas

**C**

E e C são falsas



Apenas C é falsa



Correto!

A única alternativa errada é C. O correto é Babel ser o responsável pela transcompilação e Webpack pelo bundle.

PRÓXIMA ATIVIDADE