

Vue-cli

DEVinHouse

Parcerias para desenvolver a sua carreira

SENAI

<LAB365>

AGENDA

- O que é vue-cli
- Como inicializar um projeto com vue-cli
- Estrutura do projeto
- Como renderizar um componente
- Mais diretivas
- Prática

É uma ferramenta que providencia um rápido desenvolvimento de aplicações em vue.

Procedimento para instalação



A instalação do vue-cli requer que esteja instalado na máquina o node a partir da versão 8.5!!!

Com o node instalado na máquina, agora é só executar o seguinte comando:

```
npm install -g @vue/cli
```

Para iniciar uma aplicação com o vue-cli é necessário executar o seguinte comando:

```
vue create <nome-do-projeto>
```

O será iniciado o setup de configuração do projeto. Nele podemos escolher a versão do vue que será utilizada, o tipo de processador css, dentre outras configurações

VUE-CLI

```
Vue CLI v4.5.15
```

```
New version available 4.5.15 → 5.0.4  
Run npm i -g @vue/cli to update!
```

```
? Please pick a preset:
```

```
Default ([Vue 2] babel, eslint)
```

```
> Default (Vue 3) ([Vue 3] babel, eslint)
```

```
Manually select features
```

To address issues that do not require attention, run:

```
npm audit fix
```

To address all issues (including breaking changes), run:

```
npm audit fix --force
```

Run ``npm audit`` for details.

```
🔍 Running completion hooks...
```

```
🔍 Generating README.md...
```

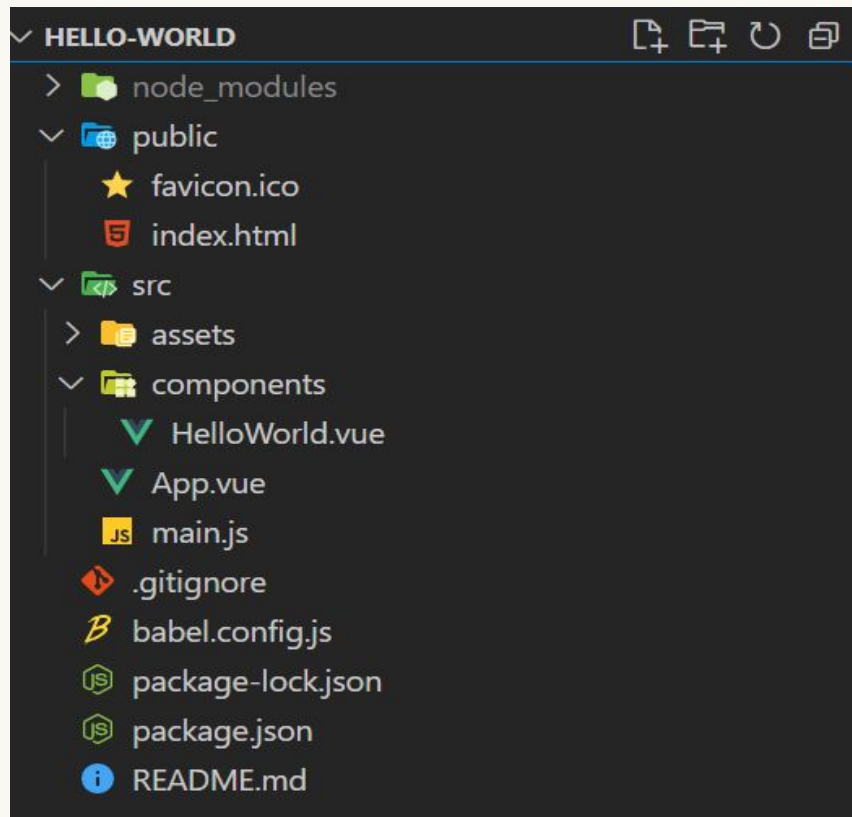
```
🔍 Successfully created project hello-world.
```

```
🔍 Get started with the following commands:
```

```
$ cd hello-world
```

```
$ npm run serve
```

VUE-CLI



VUE-CLI

EXPLORER

HELLO-WORLD

> node_modules

> public

★ favicon.ico

index.html

> src

> assets

> components

✓ HelloWorld.vue

✓ App.vue

main.js

.gitignore

babel.config.js

package-lock.json

package.json

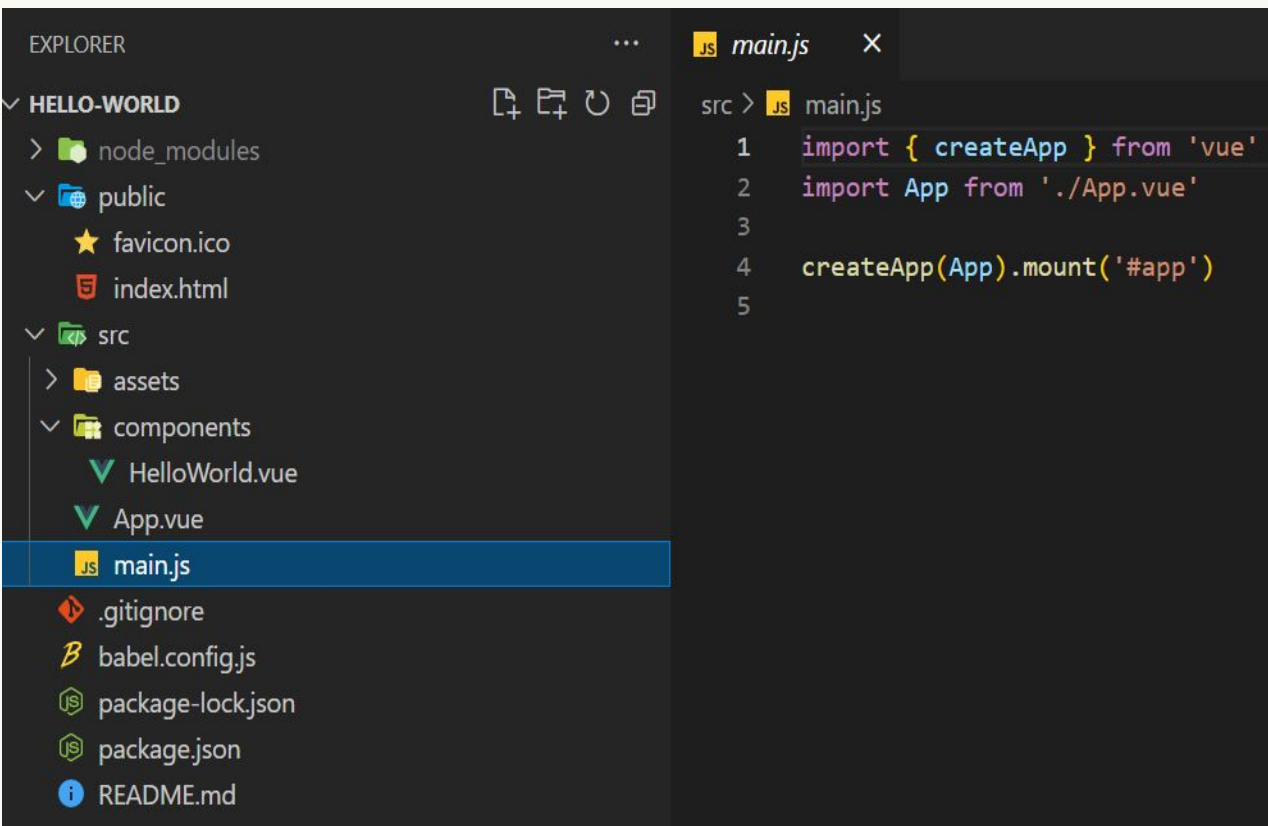
README.md

index.html

public > index.html > ...

```
1 <!DOCTYPE html>
2 <html lang="">
3   <head>
4     <meta charset="utf-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="viewport" content="width=device-width,initial-scale=1.0">
7     <link rel="icon" href="<%= BASE_URL %>favicon.ico">
8     <title><%= htmlWebpackPlugin.options.title %></title>
9   </head>
10  <body>
11    <noscript>
12      <strong>We're sorry but <%= htmlWebpackPlugin.options.title %> doesn't work properly wit
13    </noscript>
14    <div id="app"></div>
15    <!-- built files will be auto injected -->
16  </body>
17 </html>
18
```


VUE-CLI



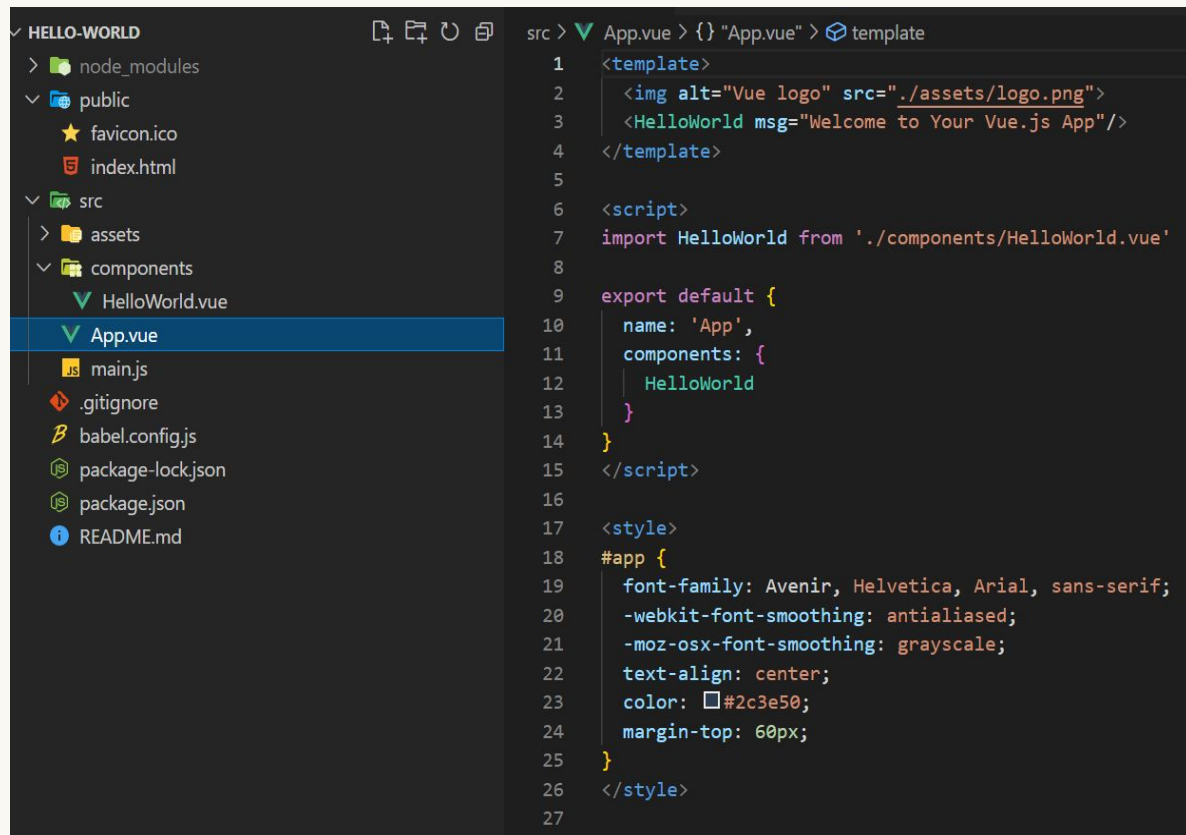
The image shows a Visual Studio Code editor window with a Vue.js project. The Explorer sidebar on the left displays the project structure:

- HELLO-WORLD
 - node_modules
 - public
 - favicon.ico
 - index.html
 - src
 - assets
 - components
 - HelloWorld.vue
 - App.vue
 - main.js** (selected)
 - .gitignore
 - babel.config.js
 - package-lock.json
 - package.json
 - README.md

The main.js file is open in the editor, showing the following code:

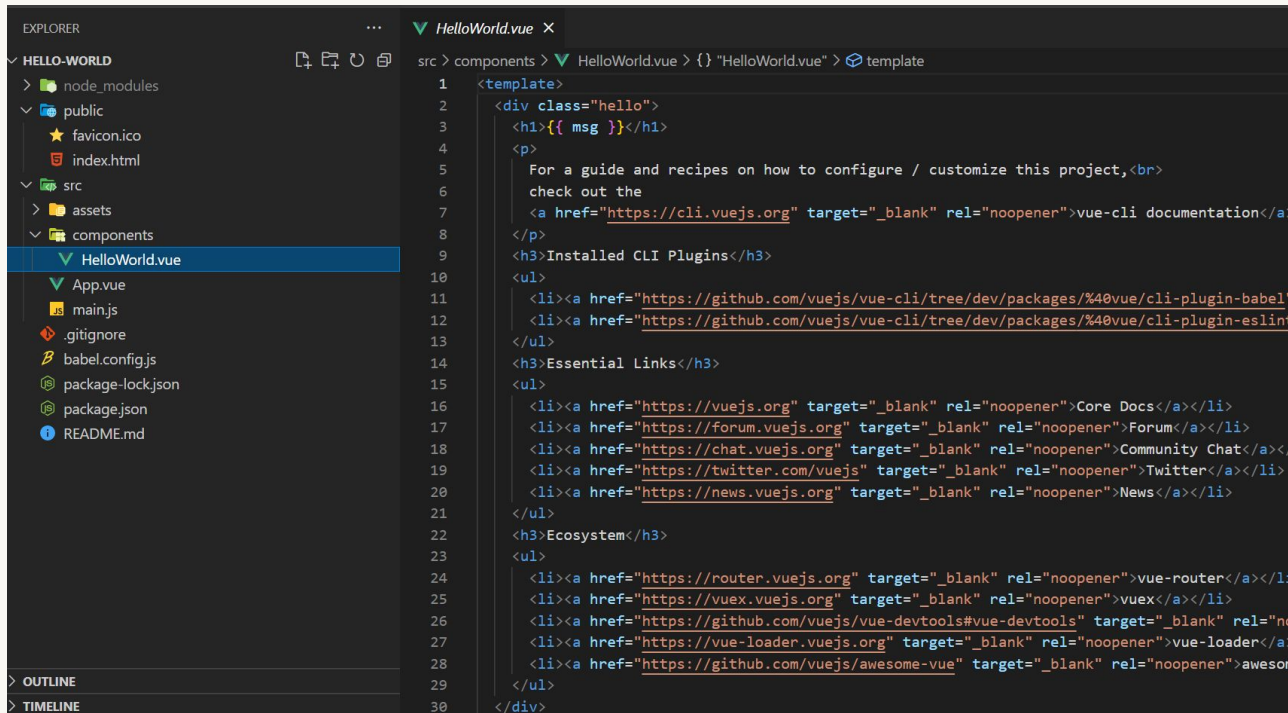
```
src > JS main.js
1  import { createApp } from 'vue'
2  import App from './App.vue'
3
4  createApp(App).mount('#app')
5
```

VUE-CLI



```
src > App.vue > {} "App.vue" > template
1 <template>
2   
3   <HelloWorld msg="Welcome to Your Vue.js App"/>
4 </template>
5
6 <script>
7   import HelloWorld from './components/HelloWorld.vue'
8
9   export default {
10     name: 'App',
11     components: {
12       HelloWorld
13     }
14   }
15 </script>
16
17 <style>
18 #app {
19   font-family: Avenir, Helvetica, Arial, sans-serif;
20   -webkit-font-smoothing: antialiased;
21   -moz-osx-font-smoothing: grayscale;
22   text-align: center;
23   color: #2c3e50;
24   margin-top: 60px;
25 }
26 </style>
27
```

VUE-CLI



The image shows a VS Code editor interface. On the left, the 'EXPLORER' sidebar displays a project structure for 'HELLO-WORLD'. The structure includes a 'public' directory with 'favicon.ico' and 'index.html', and a 'src' directory containing 'assets', 'components', 'App.vue', 'main.js', '.gitignore', 'babel.config.js', 'package-lock.json', 'package.json', and 'README.md'. The 'components' directory is expanded, showing 'HelloWorld.vue' selected. The main editor area shows the content of 'HelloWorld.vue', which is a single-file component. The file path in the breadcrumb is 'src > components > HelloWorld.vue > {} "HelloWorld.vue" > template'. The code defines a template with a 'hello' class, a message placeholder, and a paragraph of text with a link to the Vue CLI documentation. Below the paragraph, there are three sections: 'Installed CLI Plugins' with a list of links to GitHub repositories for 'cli-plugin-babel' and 'cli-plugin-eslint', and 'Essential Links' with a list of links to Vue.js resources like 'Core Docs', 'Forum', 'Community Chat', 'Twitter', 'News', and 'Ecosystem'.

```
1 <template>
2 <div class="hello">
3   <h1>{{ msg }}</h1>
4   <p>
5     For a guide and recipes on how to configure / customize this project,<br>
6     check out the
7     <a href="https://cli.vuejs.org" target="_blank" rel="noopener">vue-cli documentation</a>
8   </p>
9   <h3>Installed CLI Plugins</h3>
10  <ul>
11    <li><a href="https://github.com/vuejs/vue-cli/tree/dev/packages/%40vue/cli-plugin-babel" target="_blank" rel="noopener">cli-plugin-babel</a></li>
12    <li><a href="https://github.com/vuejs/vue-cli/tree/dev/packages/%40vue/cli-plugin-eslint" target="_blank" rel="noopener">cli-plugin-eslint</a></li>
13  </ul>
14  <h3>Essential Links</h3>
15  <ul>
16    <li><a href="https://vuejs.org" target="_blank" rel="noopener">Core Docs</a></li>
17    <li><a href="https://forum.vuejs.org" target="_blank" rel="noopener">Forum</a></li>
18    <li><a href="https://chat.vuejs.org" target="_blank" rel="noopener">Community Chat</a></li>
19    <li><a href="https://twitter.com/vuejs" target="_blank" rel="noopener">Twitter</a></li>
20    <li><a href="https://news.vuejs.org" target="_blank" rel="noopener">News</a></li>
21  </ul>
22  <h3>Ecosystem</h3>
23  <ul>
24    <li><a href="https://router.vuejs.org" target="_blank" rel="noopener">vue-router</a></li>
25    <li><a href="https://vuex.vuejs.org" target="_blank" rel="noopener">vuex</a></li>
26    <li><a href="https://github.com/vuejs/vue-devtools#vue-devtools" target="_blank" rel="noopener">vue-devtools</a></li>
27    <li><a href="https://vue-loader.vuejs.org" target="_blank" rel="noopener">vue-loader</a></li>
28    <li><a href="https://github.com/vuejs/awesome-vue" target="_blank" rel="noopener">awesome-vue</a></li>
29  </ul>
30 </div>
```

Como renderizar um componente

```
✓ App.vue ×
src > ✓ App.vue > {} "App.vue" > 📁 template
1 <template>
2   
3   <HelloWorld msg="Welcome to Your Vue.js App"/>
4 </template>
5
6 <script>
7   import HelloWorld from '../components/HelloWorld.vue'
8
9   export default {
10     name: 'App',
11     components: {
12       HelloWorld
13     }
14   }
15 </script>
16
17 <style>
18 #app {
19   font-family: Avenir, Helvetica, Arial, sans-serif;
20   -webkit-font-smoothing: antialiased;
21   -moz-osx-font-smoothing: grayscale;
22   text-align: center;
23   color: #2c3e50;
24   margin-top: 60px;
25 }
26 </style>
27
```

Mais diretivas...



```
<button v-on:click="imprimirMensagem">Imprimir</button>
```

Utilizado para escutar eventos nos componentes.

Pode ser abreviado utilizando apenas o @

```
<button @click="imprimirMensagem">Imprimir</button>
```

```
<template>
  <div class="hello">
    <p v-text="mensagem"></p>
  </div>
</template>
```



O v-text é utilizado para renderizar no html os valores de texto contigo dentro de uma variável instanciada no script.

```
<script>
export default {
  name: 'HelloWorld',
  data() {
    return {
      mensagem: 'Hello world!'
    }
  }
}
</script>
```

```
<template>
  <div class="hello">
    <div v-for="ferramenta in ferramentas"
:key="ferramenta">
      {{ ferramenta }}
    </div>
  </div>
</template>
```

Útil para renderizar um elemento html ou um componente múltiplas vezes.

```
<script>
export default {
  name: 'HelloWorld',
  data() {
    return {
      ferramentas: ['Chave de fenda', 'Serra',
'Alicate']
```


Dúvidas?





DEVinHouse

Parcerias para desenvolver a sua carreira

OBRIGADO!



<LAB365>