

Vue

Vue2 几种常见开局方式

Vue2 加了 `reader` 选项后, 再加上几种构建方式, 开局方式真是各种五花八门, 这里列几种常见的, 说说注意点

我们先建立一个 `app.vue` 来当入口组件, 即所有页面都会以这个组件为模板 (下面代码中无特别说明, `App` 即指下面这个组件)

```
<template>
  <div id="app">
    
    <hello></hello>
    <!-- 这里还可以写点其他组件, 或者路由也可以 <router-view></router-view> -->
  </div>
</template>
<script>
import Hello from './components/Hello'
export default {
  name: 'app',
  components: {
    Hello
  }
}
</script>
<style>
</style>
```

方式 1

模板文件:

```
<div id="app"></div>
```

注意: vue2 已经不支持直接绑定在 `body` 和 `html` 元素上, 所以我们需要在 `body` 里写个挂载元素

main.js 入口文件:

```
import App from './app.vue'
new Vue({
  el: '#app',
  render: createElement => createElement(App)
})
```

这里的采用 es6 的写法, 转出 es5 就是

```
render: function(createElement) {
  return createElement(App)
}
```

`createElement` 的第一个参数可以是 `String`(HTML 标签名称) | `Object`(组件对象) |

`Function`(函数), 这里传的就是个组件对象

注意: 这种情况下, `App` 组件并不是根组件

方式 2

模板文件:

```
<div id="app"></div>
```

main.js 入口文件:

```
import App from './app.vue'
new Vue({
  el: '#app',
  render: createElement => createElement('App'),
  components: {
    App
  }
})
```

这个写法是不是和第一种很像? 只不过这里传的就是个 `App` 标签, 通过 `render` 渲染一个 `<App></App>` 元素, 然后把 `App` 当组件来用

注意: 这种情况下, `App` 组件并不是根组件

方式 3

模板文件:

```
<div id="app"></div>
```

main.js 入口文件:

```
import App from './app.vue'
new Vue({
  el: '#app',
  ...App
})
```

这种方式和方式1 基本一样, 区别就在于 `render: createElement =>`

`createElement(App)` 和 `...App`

`render: createElement => createElement(App)` 是把 `App` 当成一个组件对象, 给 `render` 解析,

而 `...App` 是将 `App` 这个组件对象和 `{el: '#app'}` 这个对象直接合并, 变成Vue的参数

注意: 这种情况下, `this.$root` 是 `App` 组件

注意: 这种写法, 需要在 `.babelrc` 里添加 `stage-3` 以上的 `presets`, 如:

```
{
  "presets": ["es2015", "stage-2"]
}
```

方式 4

模板文件:

```
<div id="app">
  <App></App>
</div>
```

main.js 入口文件:

```
import App from './app.vue'
new Vue({
  el: '#app',
  components: {
    App
  }
})
```

这种写法就是完全把 `App` 当成一个组件使用, 所以我们需要在模板里添加 `<App></App>`

注意: 上面这种写法需要在 `webpack` 配置别名, 这种情况下, `App` 组件并不是根组件

```
module.exports = {
  resolve: {
    alias: {
      'vue$': 'vue/dist/vue.common.js'
    }
  }
}
```

方式 5

模板文件:

```
<div id="app"></div>
```

main.js 入口文件:

```
import App from './app.vue'
new Vue({
  el: '#app',
  template: '<App/>',
  components: {
    App
  }
})
```

这种写法也就是完全把 `App` 当成一个组件使用, 不过模板直接写在了 `template` 选项里

注意: 上面这种写法需要在 `webpack` 配置别名, 这种情况下, `App` 组件并不是根组件

```
module.exports = {
  resolve: {
    alias: {
      'vue$': 'vue/dist/vue.common.js'
    }
  }
}
```

👍 5 赞

💬 6 评论

👁 1572 浏览

🔗 分享



发表评论



citizen

涨知识了

2017-05-07 15:31:06 回复



%E6%9D%9C%E5%85%B8

中文乱码?

2017-04-08 14:45:06 回复



hello

nice

2017-02-24 13:02:43 回复



林岑影

回复 @josephine: 只是字幕mmf而已, 这个字体比较特别...

2016-12-09 21:12:27 回复



josephine

你这个logo怎么画的? 手动微笑

2016-12-09 15:12:36 回复

加载更多



React

3 篇文章



Vue

23 篇文章



ES6

5 篇文章



NodeJS

3 篇文章



JavaScript

18 篇文章



HTML+CSS

7 篇文章



Webpack

6 篇文章



前端

12 篇文章



生活

0 篇文章



其他

5 篇文章



热门文章

1

Vue 群中, 一些常见问题及解答

👍 0 💬 1

2

Vue2 dist 目录下各个文件的区别

👍 6 💬 1

3

Vue2 几种常见开局方式

👍 5 💬 6

4

Vue2 SSR渲染, 如何根据不同页面修改 meta?

👍 6 💬 6

5

vue2 如何让同名路由切换时, transition 起作用?

👍 3 💬 2