Introdução a SPA com Vue.js

Édimo Sousa Silva

Summary

1) Introduction
2) Getting started
3) Components
4) Data binding and Event binding
5) Using classes
6) Http request
7) Vue Router
8) Using props

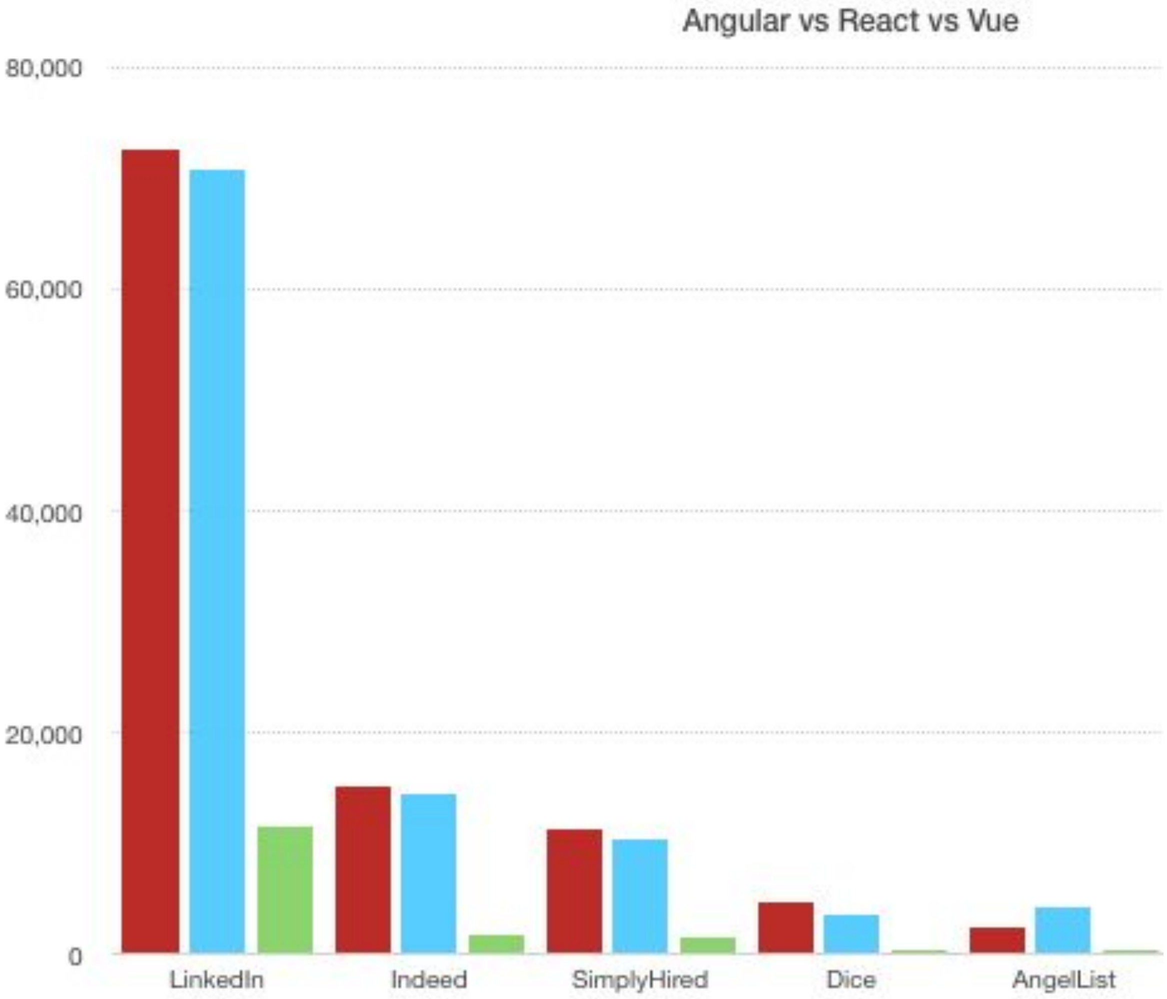
Introduction - Js history

- The beginning JavaScript frameworks.
    - Libraries: jQuery and Mootools
- First wave of modern JavaScript frameworks.
    - Backbone, Ember, Knockout, and AngularJS.
- Second wave of modern JavaScript frameworks.
    - React, Angular(Typescript), and “Vue”

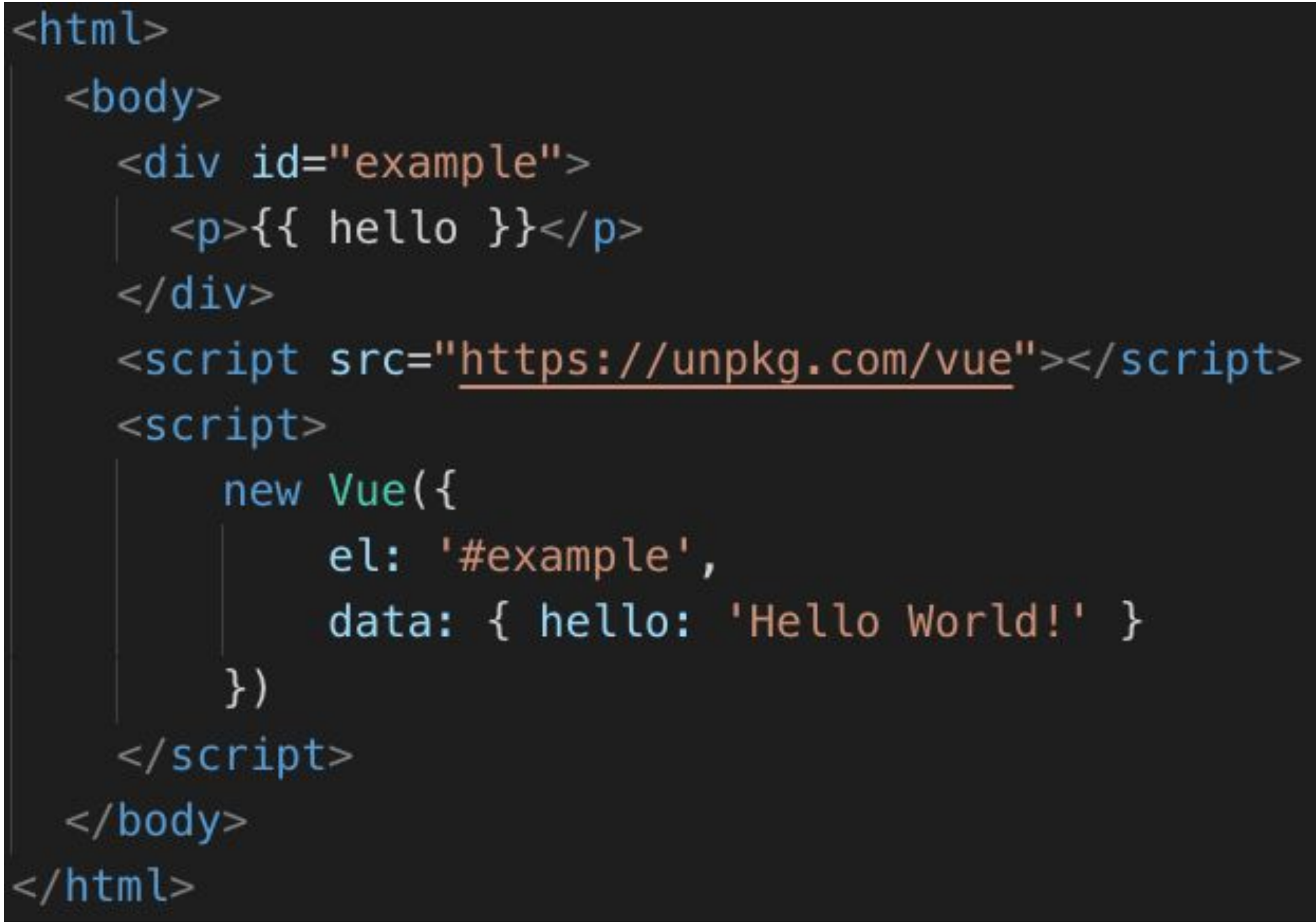
Introduction - Popularity of Vue.js

- Vue
  - 7,600 stars on GitHub in 2016
  - 36,700 stars on GitHub in 2017
  - 127,203 stars on GitHub in 2019
- React
  - 121,668 stars on GitHub in 2019
- Angular
  - 45,035 stars on GitHub in 2019

Introduction
Job Demand



Introduction
Hello World



```
<html>
  <body>
    <div id="example">
      <p>{{ hello }}</p>
    </div>
    <script src="https://unpkg.com/vue"></script>
    <script>
        new Vue({
            el: '#example',
            data: { hello: 'Hello World!' }
        })
    </script>
  </body>
</html>
```

Getting Started - Installation

1) Required
   - Node.js

2) Install vue
   - (sudo) npm install -g @vue/cli
   - (sudo) npm install -g @vue/cli-init

Getting Started - First project

1) Create new app
    - vue init webpack-simple vuestore

2) Prepare and run project
    - cd vuestore
    - npm install
    - npm run dev

Getting Started - Understanding the structure

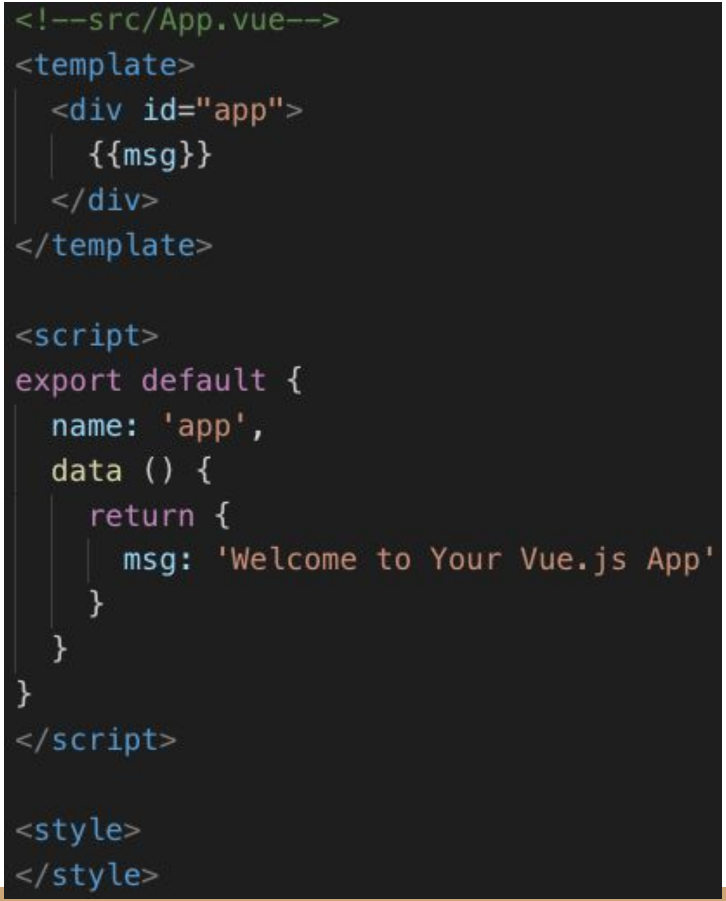
1) Three files
   - src/App.vue
   - src/main.js
   - index.html

2) node_modules

3) package.json

Getting Started - Editing the project

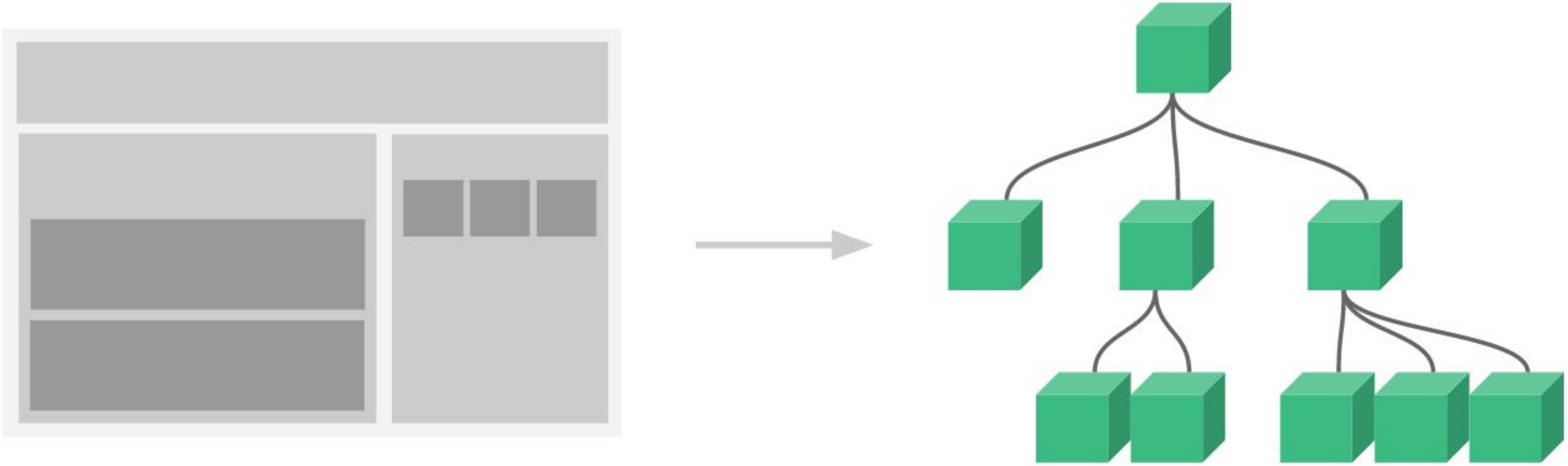
1) Open src/App.vue and edit



```
<!--src/App.vue-->
<template>
  <div id="app">
    {{msg}}
  </div>
</template>

<script>
export default {
  name: 'app',
  data () {
    return {
      msg: 'Welcome to Your Vue.js App'
    }
  }
}
</script>

<style>
</style>
```

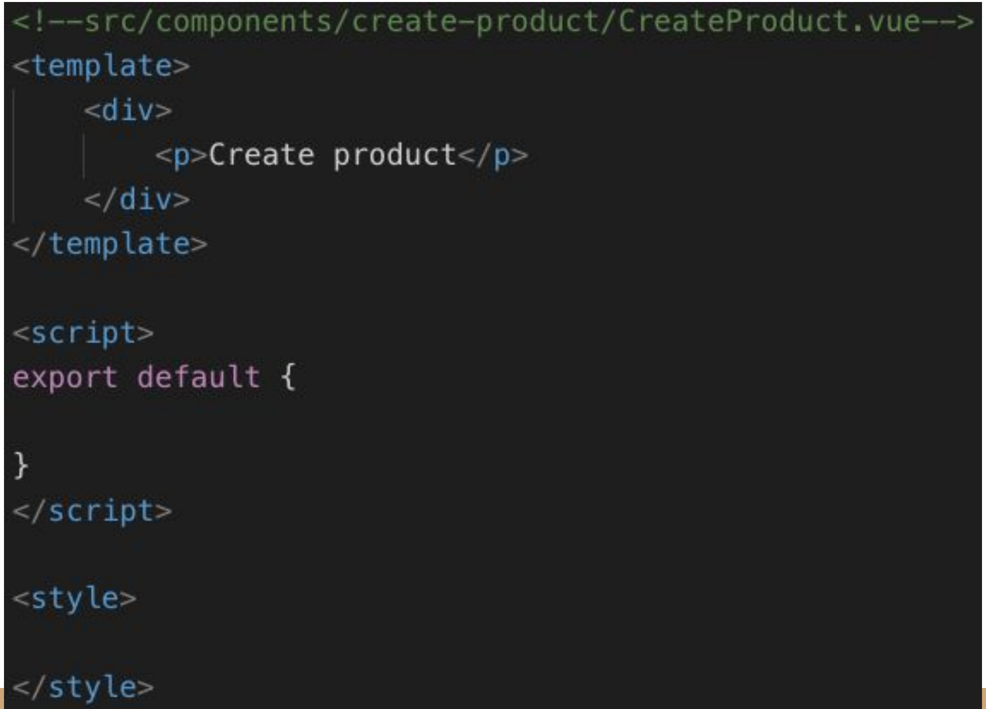
Components

1) “Components are single, independent units of an interface. They can have their own state, markup and style.“ [5]



Components - first component - part 1

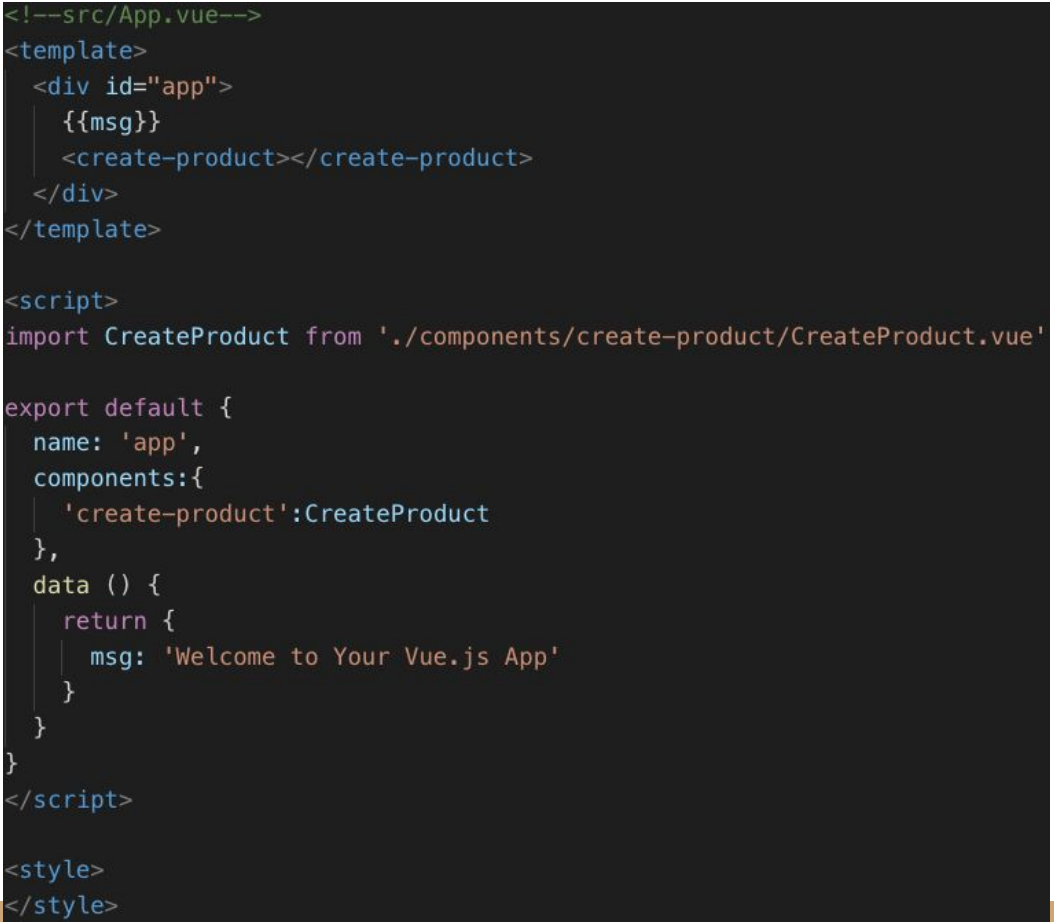
1) Create the folder “src/components/create-product”
2) Create the file “src/components/create-product/CreateProduct.vue”
3) Code the file “src/components/create-product/CreateProduct.vue”



```
<!--src/components/create-product/CreateProduct.vue-->
<template>
    <div>
        <p>Create product</p>
    </div>
</template>

<script>
export default {

}
</script>

<style>

</style>
```

Components - first component - part 2

- Code the file “src/App.vue”
  a. import CreateProduct
  b. use as component
  c. use as directive



```
<!--src/App.vue-->
<template>
  <div id="app">
    {{msg}}
    <create-product></create-product>
  </div>
</template>

<script>
import CreateProduct from './components/create-product/CreateProduct.vue'

export default {
  name: 'app',
  components:{
    'create-product':CreateProduct
  },
  data () {
    return {
      msg: 'Welcome to Your Vue.js App'
    }
  }
}
</script>

<style>
</style>
```

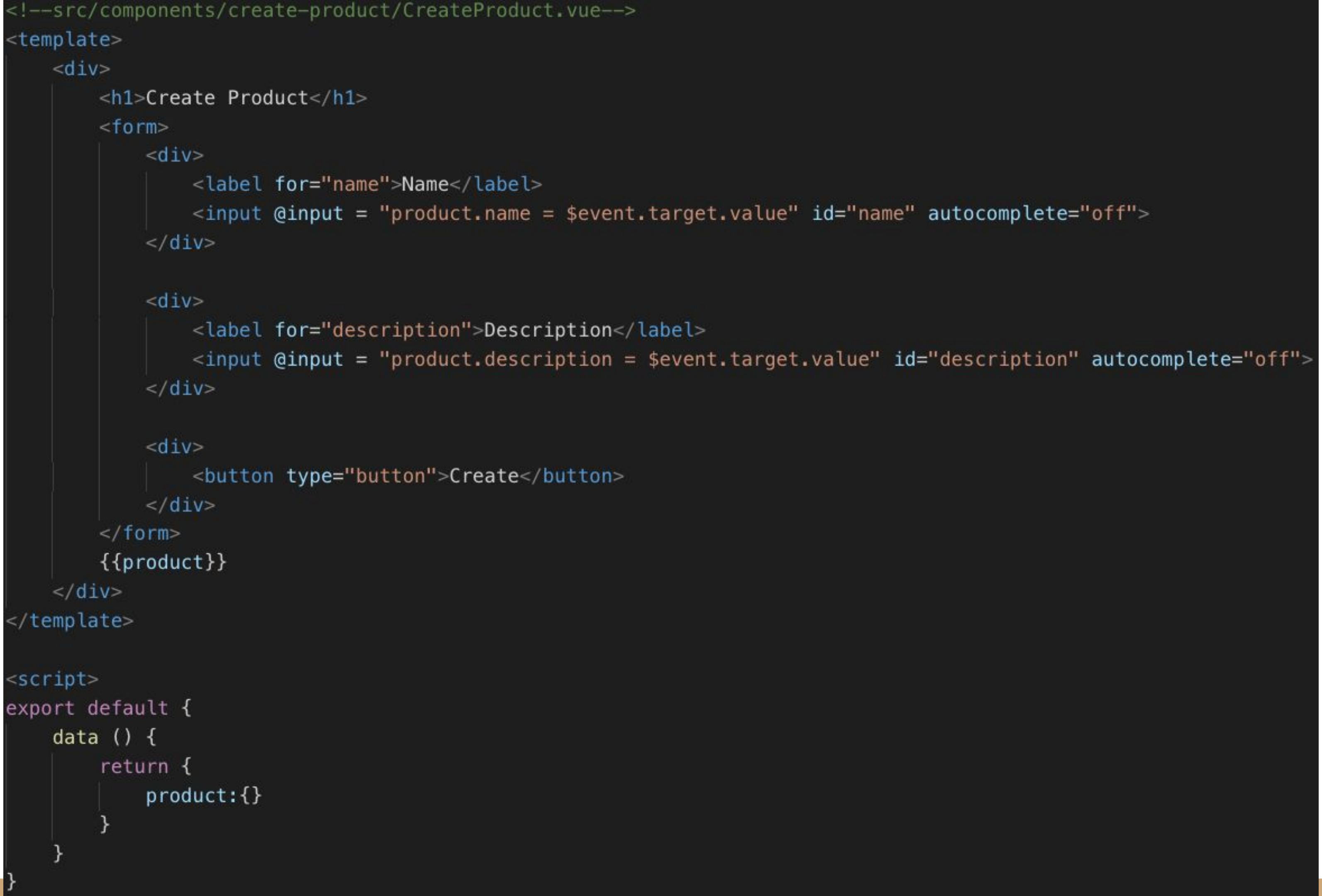
Data binding

- One way data-binding
    a. From template to script -> using @input
    b. From script to template -> interpolation using “{{ }}”

- Two way data-binding
    a. From template to script and from script to template -> using v-model

Data binding - From template to script

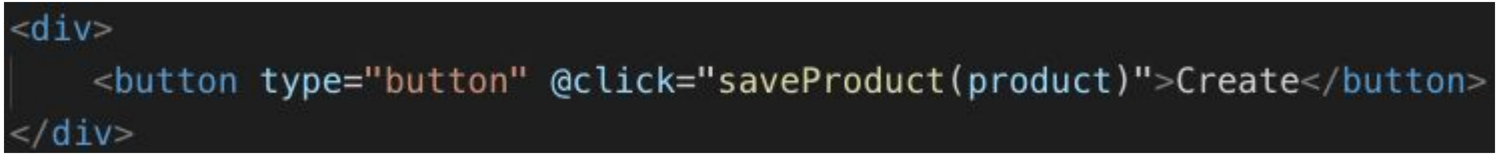
- Code the file “src/components/create-product/CreateProduct.vue”
- https://paste.ofcode.org/Rc97axbNEKedT8mnASQm6N



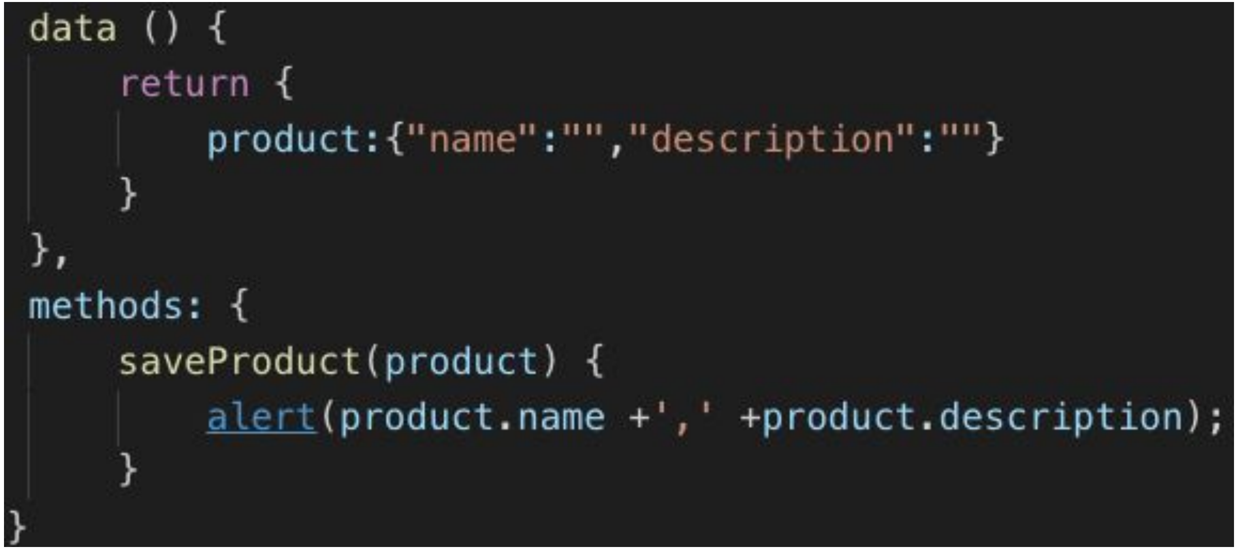
```
<!--src/components/create-product/CreateProduct.vue-->
<template>
    <div>
        <h1>Create Product</h1>
        <form>
            <div>
                <label for="name">Name</label>
                <input @input = "product.name = $event.target.value" id="name" autocomplete="off">
            </div>

            <div>
                <label for="description">Description</label>
                <input @input = "product.description = $event.target.value" id="description" autocomplete="off">
            </div>

            <div>
                <button type="button">Create</button>
            </div>
        </form>
        {{product}}
    </div>
</template>

<script>
export default {
    data () {
        return {
            product:{}
        }
    }
}
```

Event binding - click

- Code the file “src/components/create-product/CreateProduct.vue”
- https://paste.ofcode.org/GK3K2Zm9D8TqFeyRCzPfv4



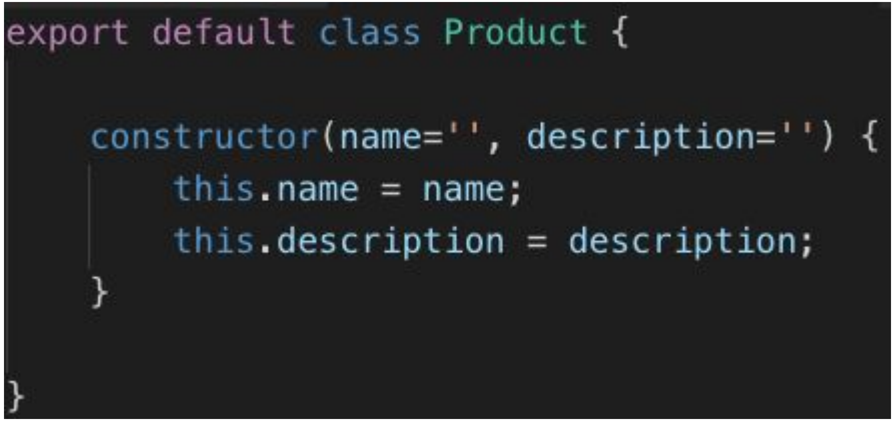
```
<div>
    <button type="button" @click="saveProduct(product)">Create</button>
</div>
```



```
data () {
    return {
        product:{"name":"","description":""}
    }
},
methods: {
    saveProduct(product) {
        alert(product.name +',' +product.description);
    }
}
```

Using classes - create class Product

1) Create the folder “src/domain”
2) Create the folder “src/domain/product”
3) Code the file “src/domain/product/Product.js”

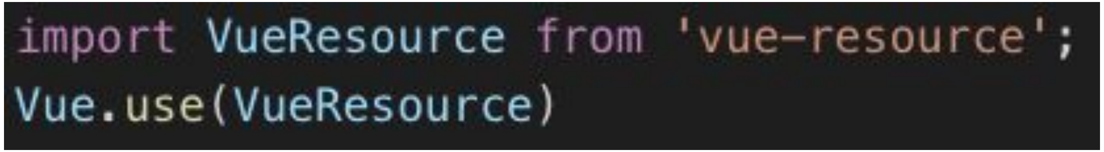


```
export default class Product {

    constructor(name='', description='') {
        this.name = name;
        this.description = description;
    }

}
```

Http request - Configure

1) Install VueResource
    - ○ npm install vue-resource

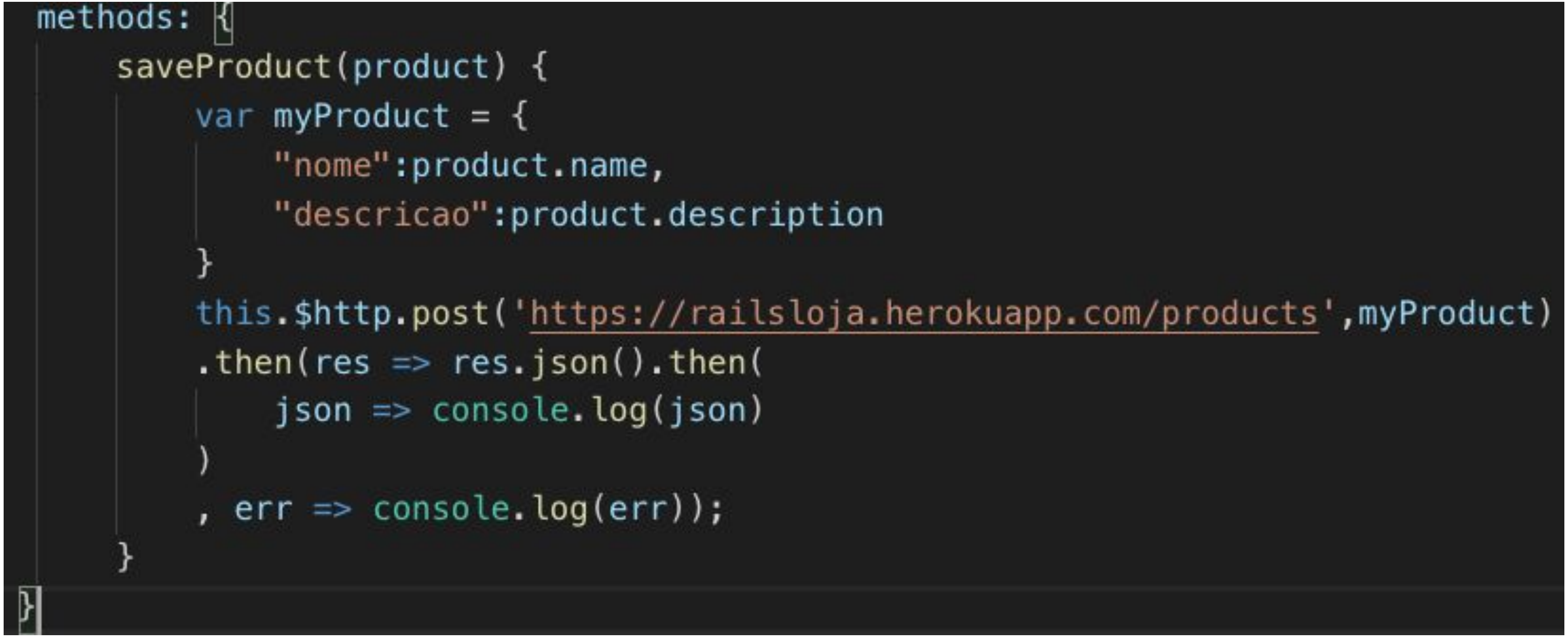
2) Edit “src/main.js”



```
import VueResource from 'vue-resource';
Vue.use(VueResource)
```

Http request - Create Product

1) Code the file “src/components/create-product/CreateProduct.vue”



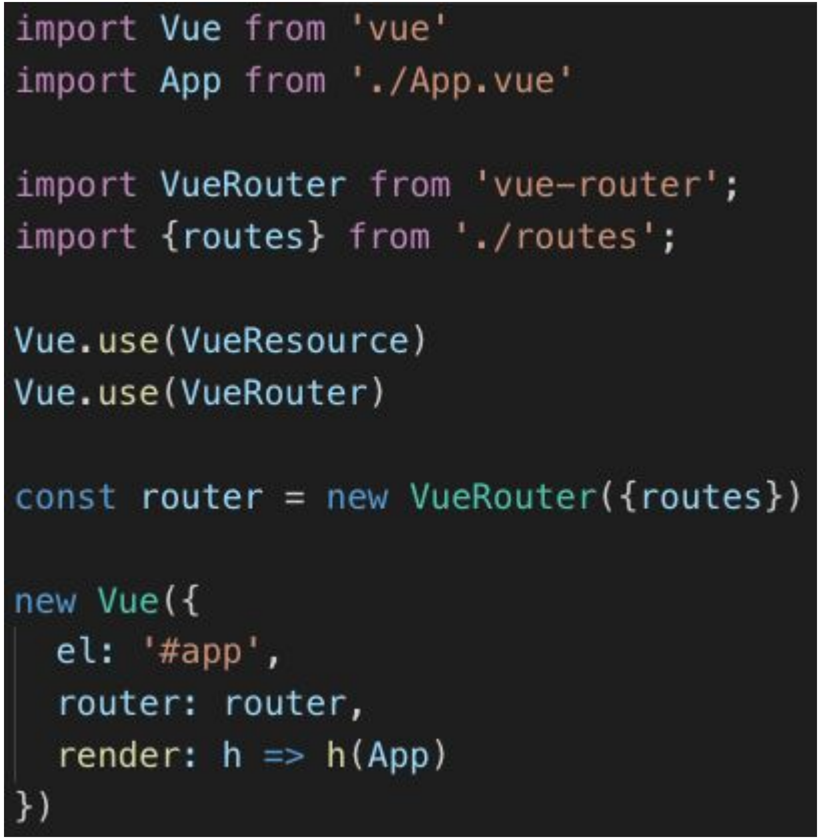
```
methods: {
    saveProduct(product) {
        var myProduct = {
            "nome":product.name,
            "descricao":product.description
        }
        this.$http.post('https://railsloja.herokuapp.com/products',myProduct)
        .then(res => res.json().then(
            json => console.log(json)
        )
        , err => console.log(err));
    }
}
```

Http request - List Products

1) Create the folder “src/components/list-product”
2) Create the file “src/components/list-product/ListProducts.vue”
3) Code: https://paste.ofcode.org/HY4N5rbvwXFHW2hyh2pE5p
4) Inside App.vue and import ListProducts
5) Inside App.vue <script> update the components
6) Inside App.vue <template> use the component

Vue Router - Configure

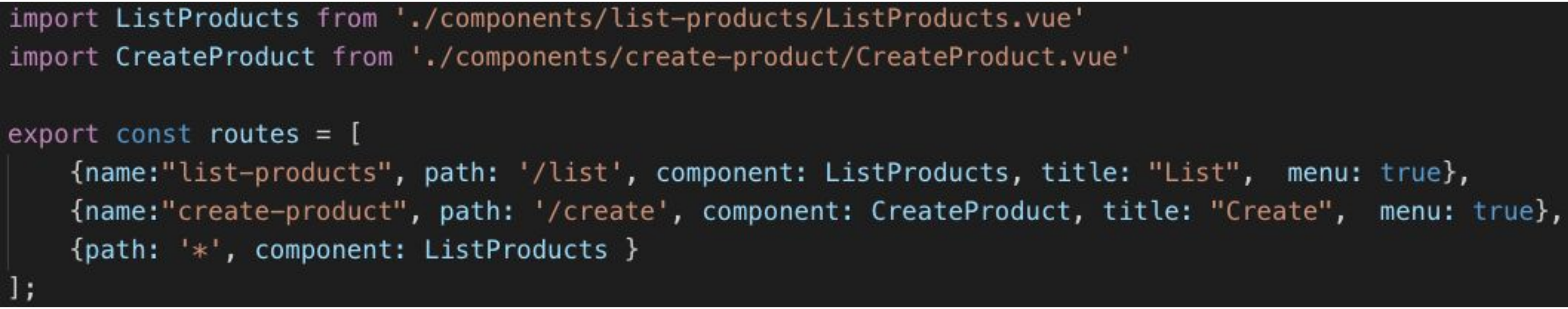
1) Install Vue Router
   - npm install vue-router
2) Edit “src/main.js” =>>>>>>>>>>>>



```
import Vue from 'vue'
import App from './App.vue'

import VueRouter from 'vue-router';
import {routes} from './routes';

Vue.use(VueResource)
Vue.use(VueRouter)

const router = new VueRouter({routes})

new Vue({
  el: '#app',
  router: router,
  render: h => h(App)
})
```

Vue Router - Create routes file

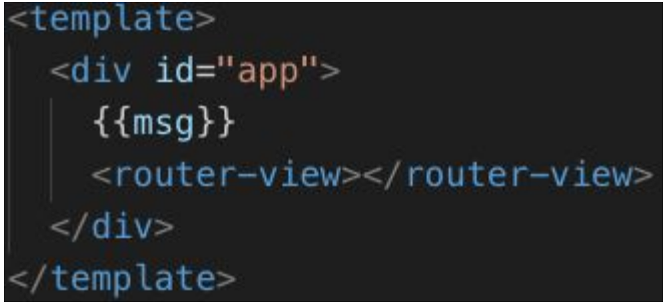
1) Create the file “src/routes.js”
2) Edit the file “src/routes.js”



```
import ListProducts from './components/list-products/ListProducts.vue'
import CreateProduct from './components/create-product/CreateProduct.vue'

export const routes = [
    {name:"list-products", path: '/list', component: ListProducts, title: "List",  menu: true},
    {name:"create-product", path: '/create', component: CreateProduct, title: "Create",  menu: true},
    {path: '*', component: ListProducts }
];
```

Vue Router - Use the ‹router-view›

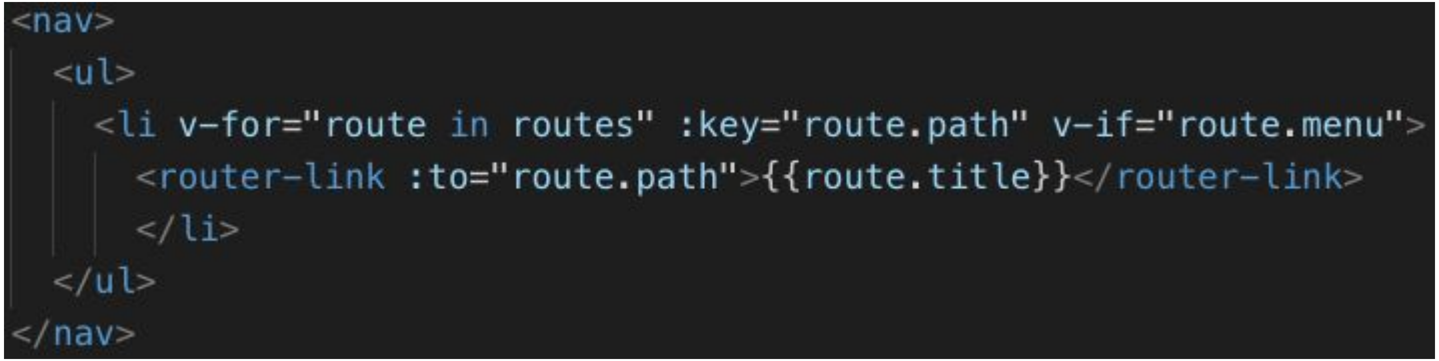
1) Open the file “src/App.vue”
2) Edit the file “src/App.vue”



```
<template>
  <div id="app">
    {{msg}}
    <router-view></router-view>
  </div>
</template>
```

Vue Router - Creating a menu

1) Open the file “src/App.vue”
2) Edit the file “src/App.vue”

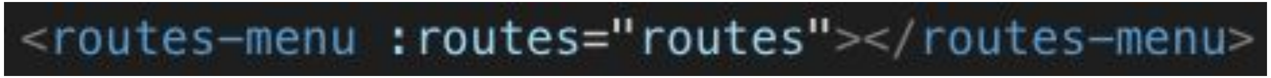


```
<nav>
  <ul>
    <li v-for="route in routes" :key="route.path" v-if="route.menu">
      <router-link :to="route.path">{{route.title}}</router-link>
    </li>
  </ul>
</nav>
```

3) Create the “routes-menu” component and use it on “src/App.vue”

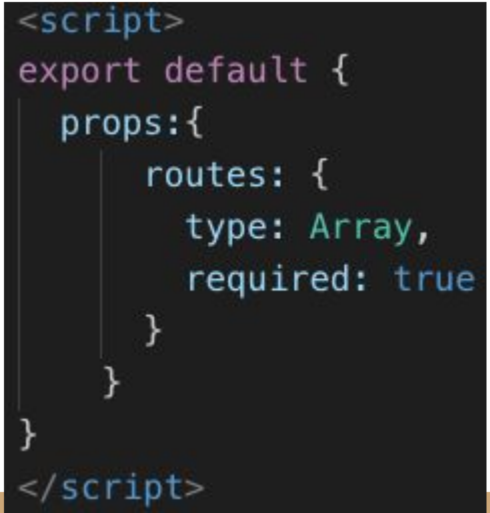
Using props - Using props in RoutesMenu

1) Open the file “src/App.vue”
2) Edit the file “src/App.vue”



```
<routes-menu :routes="routes"></routes-menu>
```

3) Edit the file “src/components/routes-menu/RoutesMenu.vue”



```
<script>
export default {
  props:{
      routes: {
        type: Array,
        required: true
      }
    }
}
</script>
```

References

[1] - Vue.js The Progressive JavaScript Framework, "https://vuejs.org/", 2019
[2] - Vue CLI Standard Tooling for Vue.js Development, "https://cli.vuejs.org/", 2019
[3] - The Vue Handbook: a thorough introduction to Vue.js , "https://medium.freecodecamp.org/the-vue-handbook-a-thorough-introduction-to-vue-js-1e86835d8446", 2019
[4] - Tech Trends Showdown: React vs Angular vs Vue, "https://medium.com/zerotomastery/tech-trends-showdown-react-vs-angular-vs-vue-61ffaf1d8706", 2019
[5] - VUE.JS COMPONENTS, "https://flaviocopes.com/vue-components/", 2019
[6] - Style Guide ,"https://vuejs.org/v2/style-guide/", 2019

Muito Obrigado!