

使用组件拆分登录注册页面

目录

使用组件拆分登录注册页面	1
目标任务	2
相关知识点	2
思路分析	2
编码步骤	2
● 效果图	2
● 实现代码	3
源码参考	8
回马枪总结	8

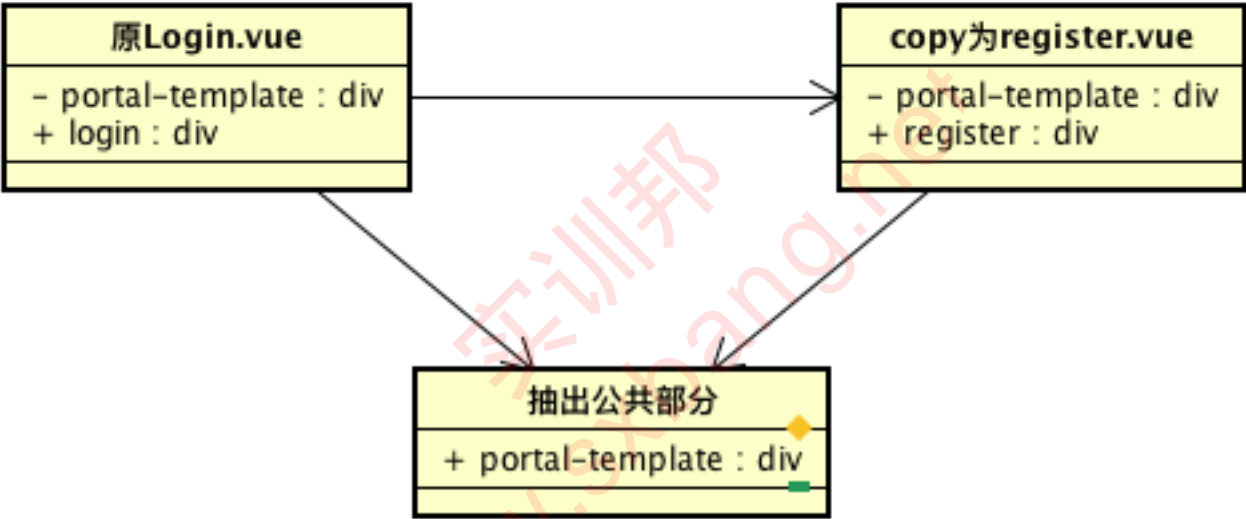
目标任务

- 使用组件拆分登录注册页面

相关知识点

- 使用组件的 slot

思路分析



编码步骤

- 效果图



* 用户名

请输入用户名!

* 密码

请输入密码!

登录

[点现在注册](#)



* 用户名

请输入用户名!

* 密码

请输入密码!

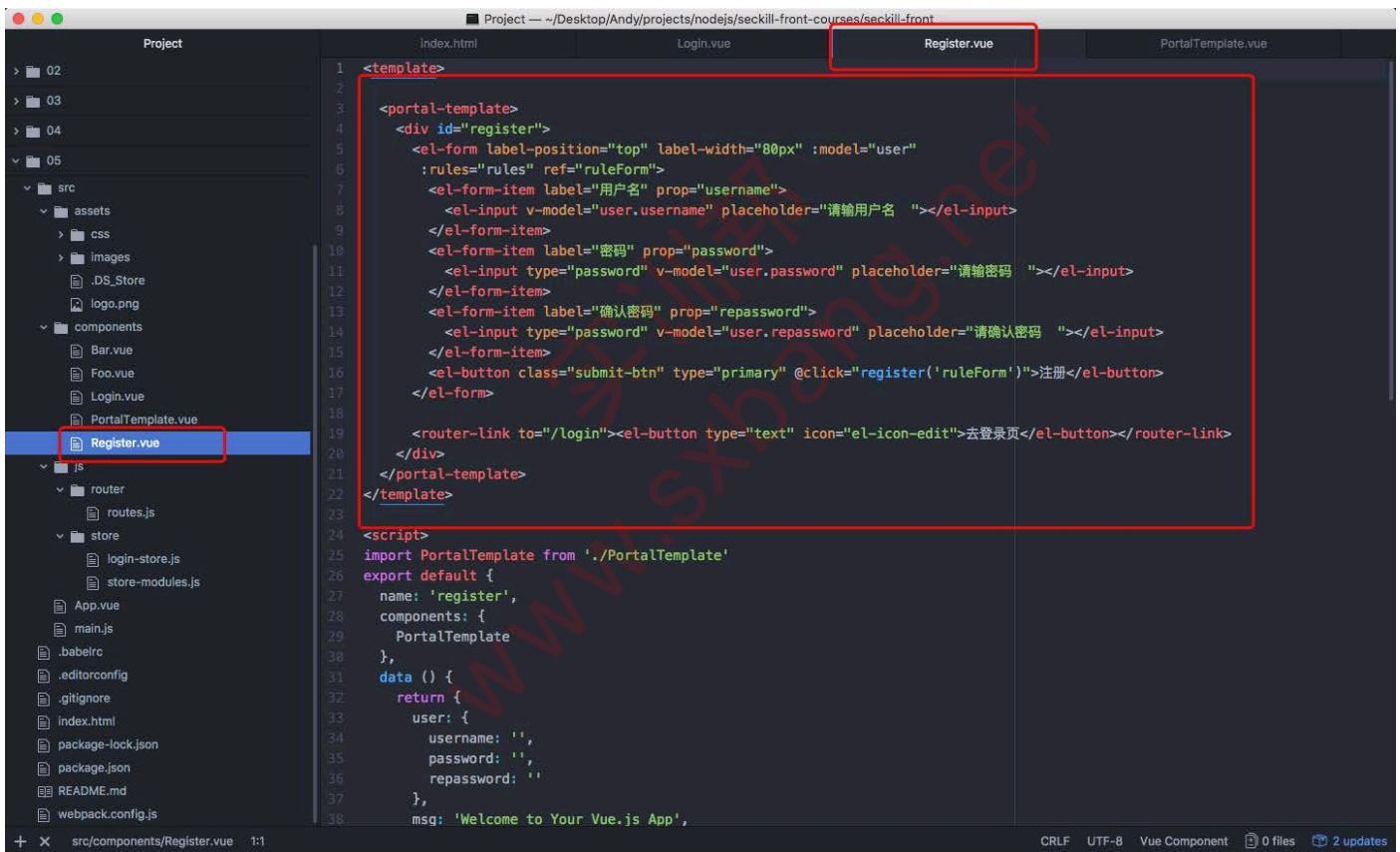
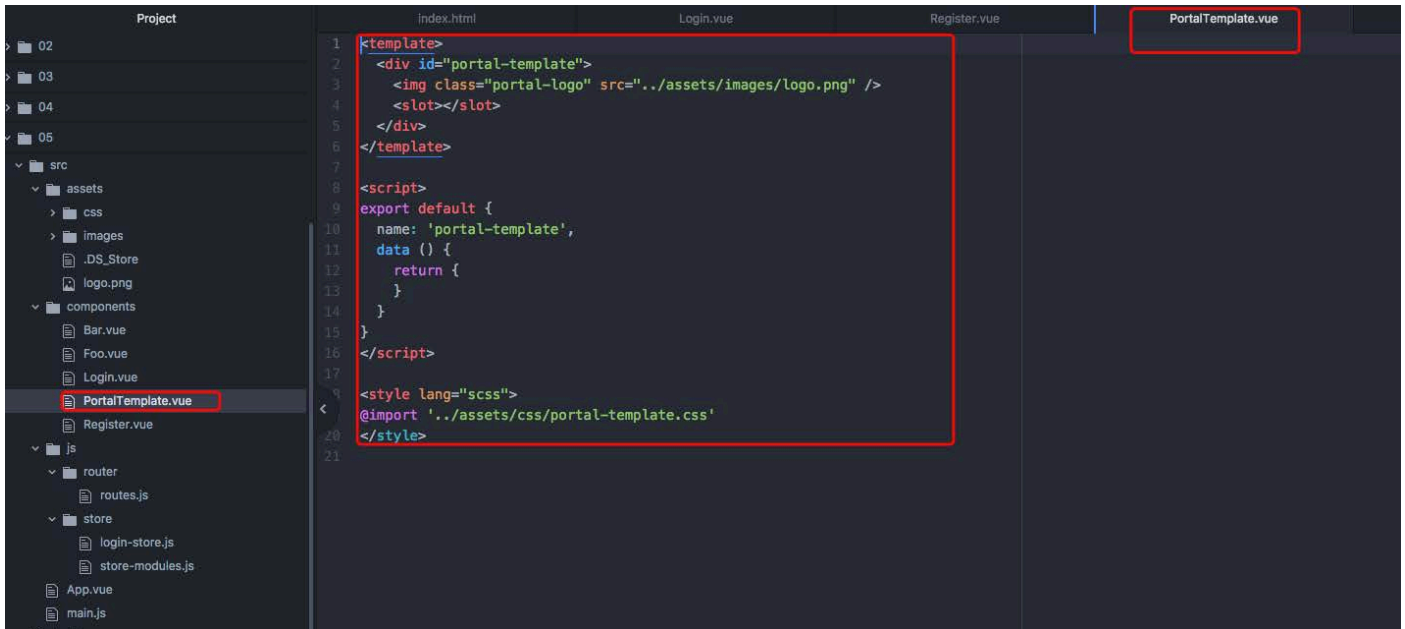
* 确认密码

请输入密码!

注册

[点去登录](#)

- 实现代码



```
24 <script>
25 import PortalTemplate from './PortalTemplate'
26 export default {
27   name: 'register',
28   components: {
29     PortalTemplate
30   },
31   data () {
32     return {
33       user: {
34         username: '',
35         password: '',
36         repassword: ''
37       },
38       msg: 'Welcome to Your Vue.js App',
39       rules: {
40         username: [
41           { required: true, message: '请输入用户名', trigger: 'blur' }
42         ],
43         password: [
44           { required: true, message: '请输入密码 ', trigger: 'blur' }
45         ],
46         repassword: [
47           { required: true, message: '请确认密码 ', trigger: 'blur' },
48           { validator: (rule, value, callback) => {
49             if (value === '') {
50               callback(new Error('请再次输入密码'));
51             } else if (value !== this.user.password) {
52               callback(new Error('两次输入密码不一致!'));
53             } else {
54               callback();
55             }
56           }, trigger: 'blur' }
57         ]
58       }
59     }
60   },
```

```
{ required: true, message: '请确认密码 ', trigger: 'blur' },
{ validator: (rule, value, callback) => {
  if (value === '') {
    callback(new Error('请再次输入密码'));
  } else if (value !== this.user.password) {
    callback(new Error('两次输入密码不一致!'));
  } else {
    callback();
  }
}, trigger: 'blur' }
]
}
},
methods: {
  register(formName) {
    var self = this;
    self.$refs[formName].validate((valid) => {
      if (valid) {
        alert('可以注册了');
        // self.$store.dispatch('register', {username: self.user.username, password: s
        // .then((response) => {
        //   self.$message.success(response.data.message)
        //   self.$router.push('/foo');
        // })
        // .catch((response) => {
        //   self.$message.error(response.data.message)
        // })
      }
    });
  }
}
}
}
</script>

<style lang="scss">
</style>
```

```
<template>
  <portal-template>
    <div id="login">
      <el-form label-position="top" label-width="80px" :model="user"
        :rules="rules" ref="ruleForm">
        <el-form-item label="用户名" prop="username">
          <el-input v-model="user.username" placeholder="请输入用户名 "></el-input>
        </el-form-item>
        <el-form-item label="密码" prop="password">
          <el-input type="password" v-model="user.password" placeholder="请输入密码 "></el-input>
        </el-form-item>
        <el-button class="submit-btn" type="primary" @click="login('ruleForm')">登录</el-button>
      </el-form>
      <router-link to="/register"><el-button type="text" icon="el-icon-edit">去注册页</el-button></router-link>
    </div>
  </portal-template>
</template>

<script>
import PortalTemplate from './PortalTemplate'
export default {
  name: 'login',
  components: {
    PortalTemplate
  },
  data () {
    return {
      user: {
        username: '',
        password: ''
      },
      msg: 'Welcome to Your Vue.js App',
      rules: {
        username: [
          { required: true, message: '请输入用户名', trigger: 'blur' }
        ],
        password: [
          { required: true, message: '请输入密码 ', trigger: 'blur' }
        ]
      }
    }
  }
}
```



```
index.html | Login.vue | Register.vue | PortalTemplate.vue
username: '',
password: ''
},
msg: 'Welcome to Your Vue.js App',
rules: {
  username: [
    { required: true, message: '请输入用户名', trigger: 'blur' }
  ],
  password: [
    { required: true, message: '请输入密码 ', trigger: 'blur' }
  ]
}
},
methods: {
  login(formName) {
    var self = this;
    self.$refs[formName].validate((valid) => {
      if (valid) {

        self.$store.dispatch('login', {username: self.user.username, password: self.user.password})
        .then((response) => {
          self.$message.success(response.data.message)
          self.$router.push('/foo');
        })
        .catch((response) => {
          self.$message.error(response.data.message)
        })
      }
    });
  }
}
}
</script>

<style lang="scss">
</style>
```

-

源码参考

N/A

回马枪总结

- 尽量考虑拆分公共部分，以便重用
- 可以使用 slot