Main.js文件：

// The Vue build version to load with the `import` command

// (runtime-only or standalone) has been set in webpack.base.conf with an alias.

import Vue from 'vue'

import App from './App'

import VueRouter from 'vue-router'

import Test1 from './components/Test1.vue'

import Users from './components/Users.vue'

Vue.config.productionTip = false

Vue.use(VueRouter)

//设置路由

const router = new VueRouter({

    mode: 'history',

    routes: [{

            path: "/user",

            component: Users

        },

        {

            path: "/test",

            component: Test1

        }

    ]

})

/\* eslint-disable no-new \*/

new Vue({

    router,

    template: '<div id="app"><router-link to="/user">Users</router-link><hr/><router-link to="/test">test</router-link><hr/><router-view></router-view></div>'

}).$mount("#app")

对应的vue组件：

Users.vue:

<template>

<div class="users">

     <form class="form-inline" v-on:submit="addUser">

     <div class="form-group">

     <label for="exampleInputName2">姓名</label>

     <input type="text" class="form-control" id="exampleInputName2" placeholder="Jane Doe" v-model="newUser.name">

     </div>

     <div class="form-group">

     <label for="exampleInputEmail2">Email</label>

     <input type="email" class="form-control" id="exampleInputEmail2" placeholder="jane.doe@example.com" v-model="newUser.emial">

     </div>

     <button type="submit" class="btn btn-default" >提交</button>

    </form>

     <!--用于展示用户信息-->

<table class="table table-bordered table-hover ">

        <thead>

            <tr>

                <th >

                    <input type="checkbox" class="toogle" />

                </th>

                <th>姓名</th>

                <th>年龄</th>

                <th>地址</th>

                <th>email</th>

                <th>操作</th>

            </tr>

        </thead>

        <tbody>

            <tr v-for="user in users">

            <td>

                <input type="checkbox" class="toogle" v-model="user.contacted"/>

            </td>

            <td>{{user.name}}</td>

            <td>{{user.age}}</td>

            <td>{{user.addrress}}</td>

            <td>{{user.email}}</td>

            <td>

                <button type="button" class="btn btn-primary"><span class="glyphicon glyphicon-edit" aria-hidden="true"></span>编辑</button>

                <button type="button" class="btn btn-danger" v-on:click="deleteUser(user)"><span class="glyphicon glyphicon-trash" aria-hidden="true"></span>删除</button>

            </td>

            </tr>

        </tbody>

</table>

</div>

</template>

<script>

//引入jquery

import $ from "jquery";

import "../assets/css/bootstrap.min.css";

import "../assets/css/bootstrap-theme.min.css";

import "../assets/js/jquery-3.3.1.min.js";

import "../assets/js/bootstrap.min.js";

export default {

name: "users",

data(){

        return {

            newUser:{

            },

            users:[

                {

                    name:'zx',

                    age:18,

                    addrress:'江苏南京',

                    email:'1773203101@qq.com',

                    contacted:false,

                },

                {

                    name:'zhiyi',

                    age:19,

                    addrress:'中国北京',

                    email:'1773203101@qq.com',

                    contacted:false,

                },

                {

                    name:'zhuxu',

                    age:20,

                    addrress:'中国上海',

                    email:'1773203101@qq.com',

                    contacted:false,

                },

            ]

        }

    },

    methods:{

        addUser:function(e){

            this.users.push({

                name:this.newUser.name,

                email:this.newUser.emial,

                age:20,

                addrress:'江苏苏州',

                contacted:false,

            })

            e.preventDefault();

        },

        deleteUser:function(user){

            this.users.splice(this.users.indexOf(user),1)

        }

    }

};

</script>

<!-- Add "scoped" attribute to limit CSS to this component only -->

<!--scope只会影响到当前组件的样式-->

<style scoped>

    .contacted{

        text-decoration: line-through;

    }

</style>

Tset1.vue:

<template>

<div class="test">

<h2>{{title}}</h2>

<h3 v-if="showName">姓名：{{user.name}}</h3>

<h3 v-else="showName">地址：{{user.address}}</h3>

<h3>年龄：{{user.age}}</h3>

<ul>

    <li v-for="item in items">{{item.title}}</li>

    </ul>

<button v-on:click="greet()" class="btn btn-primary">Say Greeting</button>

<h4>监听键盘事件</h4>

<input type="text" v-on:keyup="pressKeybord" id="input"/>

<h4>监听键盘指定单个按钮事件——输入Enter进行监听</h4>

<input type="text" v-on:keyup.enter="pressKeybord\_enter" id="input\_enter"/>

<h4>监听键盘指定多个按钮事件——输入Alt+Enter进行监听</h4>

<input type="text" v-on:keyup.alt.enter="pressKeybord\_alt\_enter" id="input\_alt\_enter"/>

<h3>bootstrap的使用</h3>

<div class="row">

<div class="col-md-1">.col-md-1</div>

<div class="col-md-1">.col-md-1</div>

<div class="col-md-1">.col-md-1</div>

<div class="col-md-1">.col-md-1</div>

<div class="col-md-1">.col-md-1</div>

<div class="col-md-1">.col-md-1</div>

<div class="col-md-1">.col-md-1</div>

<div class="col-md-1">.col-md-1</div>

<div class="col-md-1">.col-md-1</div>

<div class="col-md-1">.col-md-1</div>

<div class="col-md-1">.col-md-1</div>

<div class="col-md-1">.col-md-1</div>

</div>

<h3>Computed计算属性</h3>

<form>

<div class="form-group">

<label for="exampleInputEmail1">姓名：</label>

<input type="text" class="form-control" id="exampleInputEmail1" v-model="user.name">

</div>

<div class="form-group">

<label for="exampleInputPassword1">地址：</label>

<input type="text" class="form-control" id="exampleInputPassword1" v-model="user.address">

</div>

<div class="form-group">

<label for="exampleInputPassword1">姓名&地址：</label>

<input type="text" class="form-control" id="exampleInputPassword1" v-model="nameAndaddress">

</div>

</form>

<h3>props属性</h3>

<h4>{{msg}}</h4>

</div>

</template>

<script>

//引入jquery

import $ from "jquery";

import "../assets/css/bootstrap.min.css";

import "../assets/css/bootstrap-theme.min.css";

import "../assets/js/jquery-3.3.1.min.js";

import "../assets/js/bootstrap.min.js";

export default {

name: "test",

data() {

return {

user: {

name: "zhuxu",

age: 12,

address: "江苏南京"

},

title: "Hello Word Vue.js!",

showName: true,

items: [{ title: "item1" }, { title: "item2" }, { title: "item3" }]

};

},

methods: {

greet: function() {

alert("Hello Word");

},

pressKeybord: function() {

var content = $("#input").val();

console.log(content);

},

pressKeybord\_enter: function() {

var content = $("#input\_enter").val();

console.log("input\_enter=" + content);

},

pressKeybord\_alt\_enter: function() {

var content = $("#input\_alt\_enter").val();

console.log("input\_alt\_enter=" + content);

}

},

computed: {

nameAndaddress: function() {

return this.user.name + " " + this.user.address;

}

},

props: {

msg: {

type: String,

default: "默认就是当前内容"

}

}

};

</script>

<!-- Add "scoped" attribute to limit CSS to this component only -->

<!--scope只会影响到当前组件的样式-->

<style scoped>

</style>