**今天学习的内容**

**1:复习昨天重点**

**练习: 9:20--9:30**

**(1)创建二个组件 main(父组件)/son(子组件)**

**(2)main 显示一个按钮**

**(3)son 显示 一个p标签**

**(4)main 定义一个变量count初始化为0**

**将count传递给son组件,在son子组件接收数据并且在p标签显示**

**(5)main 点击按钮时，实现count自增操作**

**(6)要求 son组件显示count对应变化**

**(7)son在接到count之后计算,在count大于10时候,将main按钮禁用**

**提示:**

**<button v-bind:disabled="isValid">clickMe</button>**

**总结:**

**1:传递到子组件的数据，父组件中数据发生变化子组件一并更新**

**2:子组件中updated方法，会随着数据改变执行**

**3:修改父元素数据 this.$parent**

**2:今天的目标**

**2.1: Vue Router 路由**

**2.2 SPA(single page application)**

**SPA(single page application)单页面应用程序，在一个完成的应用或**

**者站点中，只有一个完整的HTML页面，这个页面有一个容器，**

**可以把需要加载的代码插入到该容器中.**

**2.3SPA的工作原理**

**http://127.0.0.1/index.html#/start**

**(1)根据地址栏 中url解析完的页面index.html 加载index.html**

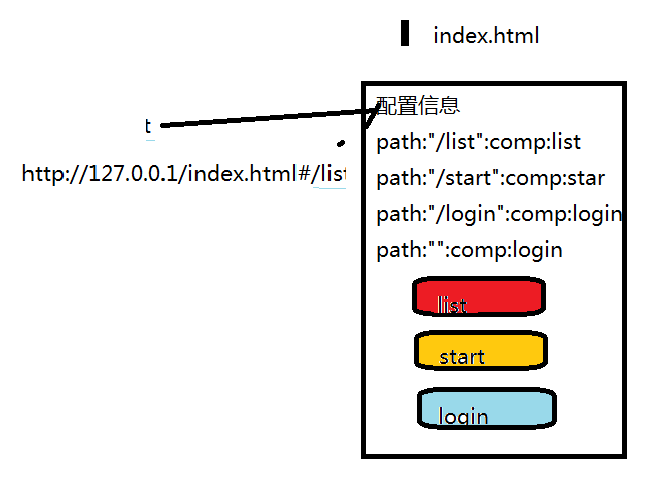
**(2)根据地址栏 解件#后的路由地址:start**

**根据路由地下，去当前应用的配置中找路由地址配置对象去查**

**找路由地址，所对应的模块页面地址**

**发起异步请求加载该页面地址**

**(3)把请求来的数据加载到指定容器中**

****

**2.4 SPA的实现步骤----(重点&难点)**

**(1)引入对应 vue-router.js 文件**

**(2)显示组件 前:<div id="app"><parent></parent></div>**

**显示组件 现: <div id="app">**

**<router-view></router-view> 渲染不同组件**

**</div>**

**(3)创建各个组件**

**(4)配置信息(路由词典)**

**每一个路由地址配置对象(组件)**

**const myRoutes=[**

**{path:"/start",component:myStart},**

**{path:"/list",component:myList},**

**];**

**(5)创建路由表对象(语法标签)**

**const myRouter = new VueRouter({**

**routes: myRoutes**

**});**

**(6)在Vue对象添加现在应用路由表对象**

**new Vue({**

**router: myRouter**

**});**

**(7)测试**

**在地址中输入对应的不同路由地址中，显示不同组件**

**练习:11:25--11:35**

**创建一个SPA,这个SPA有3个组件，分别对**

**path /collect -->收藏页组件**

**path /detail -->详情页组件**

**path /order -->订单页组件**

**2.5 SPA 通过VueRouter来实现组件之间跳转**

**提供了3种方式实现跳转**

**(1)直接修改地址栏中路由地址**

**(2)通过router-link 实现跳转**

**<router-link to="/order">订单</router-link>**

**(3)通过js的编程方式来实现**

**jumpToLogin:function(){**

**this.$router.push("/login");**

**}**

**2.6 SPA 通过VueRouter来实现组件之间跳转(参数传递)**

**login--->用户名--->main**

**(1)明确发送方和接收方**

**(2)配置接收方法路由地址**

**原生**

**{path:"/main",component:MainComponent}**

**{path:"/main/:uname/:upwd", component:MainComponent }**

**(3)接收方法获取传递的数据**

**this.$route.params //params 数据里面多个参数对象**

**this.$route.params.uname**

**this.$route.params.upwd**

**(4)跳转时,发送参数**

**this.$router.push("/main/tom");**

**<router-link to="/main/jerry">跳转</router-link>**

**练习：有两个组件**

**product-list: 构造一个有5条数据组，显示在列表中,**

**每一个列表项添加一个超链接**

**product\_detail:pid 显示字符串 pid 产品详细编号**

**功能:点击product-list列表项,跳转product-detail,**

**并且将数组下标发送product-detail,product-detail接收pid**

**并且显示在p标签中.**

**(1)明确发送 接收 15:20--15:30**

**发送: product-list**

**接收:product-detail**

**(2)配置接收方路由地址**

**{path:"/detail/:pid"}**

**(3)接收方获取参数值**

**this.$route.params.pid**

**(4)发送参数**

**<router-link :to="/detail/"+pid></router-link>**

**this.$router.push("/detail/20");**

**2.6 SPA 路由设置高级用法**

**alias 别名(访问路径)**

**{path:"/list",component:MyList,alias:"/lists"}**

**redirect 重定向(重新地址栏地址转换/productList-->/list)**

**{path:"/productList",redirect:"/list"}**

**path:"\*" 异常处理**

**{path:"\*", component:NotFound}**

**2.7: 路由嵌套**

**示例:**

**163邮箱为例**

**login/main(根据需求 加载inbox或者outbox)**

**main**

**--inbox**

**--outbox**

**路由嵌套的SPA实现的步骤**

**A(/a)组件需要嵌套B组件(/c /d)**

**(1)准备嵌套其它组件的父组件指定一个容器**

**在B组件指定容器:模板**

**<router-view></router-view>**

**(2)在B组件的路由配置对象中指定 children属性**

**{**

**path:"/b",component:B**

**children:[**

**{path:"/c",component:C},**

**{path:"/d",component:D}**

**]**

**}**

**2.8: vue-resource vue http协议插件(实现ajax)**

**(1)先加载vue.js,再加载vue-resource.js**

**(2)使用**

**this.$http.get(url).then(function(data){**

**},function(err){**

**});**

**2.9: vue cli 脚手架(创建vue项目工具)**

**(1)当代码修改，直接预览效果 hot reload**

**(2)框架己搭建好，快速进入开**

**(3)组织代码的方式**

**将组件通过 .vue 文件单独组件，放在src目录 compoents**

**注意事项:**

**(1) 保证node.js 正确(版本高一些 6.00 >)**

**(2) 保证网络通 (安装组件)**

**远程中心：开网**

**作业1:resource 综合示例 todolist**

**(1)完成组件创建和使用**

**(2)显示用户名列表(ajax)-->删除**