

Mint UI - Unit05

1.作业

1.1 函数的封装

在 `Index.vue` 中 `mounted`、无限滚动及切换顶部选项卡时，都需要从服务器的 `articles` 接口中获取数据，而且业务基本上是相同的，所以建议该过程封装为函数。

```

160 methods: {
161   // 加载数据的方法
162   loadData(id, page) {
163     // 修改状态
164     this.busy = true;
165     // 显示加载提示框
166     this.$indicator.open({
167       text: "加载中...",
168       spinnerType: "double-bounce"
169     });
170     this.axios
171       .get("/articles", {
172         params: {
173           id: id,
174           page: page,
175         },
176       })
177       .then((res) => {
178         let articles = res.data.results;
179         articles.forEach(article => {
180           if (article.image != null) {
181             article.image = require('../assets/images/articles/' +
              article.image);
182           }
183           this.articles.push(article);
184         });
185         // 修改状态
186         this.busy = false;
187         // 关闭加载提示框
188         this.$indicator.close();
189       });
190   },
191   // 向下滚动时调用的方法
192   loadMore() {
193     // 页码累加
194     this.page++;
195     // 调用loadData()方法
196     this.loadData(this.active, this.page);
197   },
198 },
199 watch: {
200   active(value) {
201     // 清空之前的数据
202     this.articles = [];
203     // 将页码重置1
204     this.page = 1;
205     // 此时的value代表的切换后的顶部选项卡的ID
206     // 既然id(分类ID)已经知道了,那么此时就需要发送请求以获取服务器的相关数据
207     // 调用loadData()方法
208     this.loadData(value, this.page);
209   },
210 },
211 },
212 mounted() {
213   // 发送HTTP请求以获取服务器所有分类的数据
214   this.axios.get("/category").then((res) => {
215     // 获取服务器返回的结果
216     let results = res.data.results;
217     // 将服务器返回结果存储到category变量中
218     this.category = results;
219   });
220 }

```

```
220 * * //////////////////////////////////////
221 * * // 发送HTTP请求以获取服务器中默认分类下包含的文章数据
222 * * // 调用loadData()方法
223 * * this.loadData(this.active,1);
224 * * },
225 * * };
226 </script>
```