

```
1. /index.html
2. <!DOCTYPE html>
3. <html lang="en">
4.   <head>
5.     <meta charset="UTF-8" />
6.     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7.     <title>书城首页</title>
8.     <link rel="stylesheet" href="/static/css/minireset.css" />
9.     <link rel="stylesheet" href="/static/css/common.css" />
10.    <link rel="stylesheet" href="/static/css/iconfont.css" />
11.    <link rel="stylesheet" href="/static/css/index.css" />
12.    <link rel="stylesheet" href="/static/css/swiper.min.css" />
13.  </head>
14.  <body>
15.    <div id="app">
16.      <div class="topbar">
17.        <div class="w">
18.          <div class="topbar-left">
19.            <i>送至:</i>
20.            <i>北京</i>
21.            <i class="iconfont icon-ai-arrow-down"></i>
22.          </div>
23.          <div class="topbar-right">
24.            <a href="/pages/user/login.html" class="login">登录</a>
25.            <a href="/pages/user/regist.html" class="register">注册</a>
26.            <a
27.              href="/pages/cart/cart.html"
28.              class="cart iconfont icon-gouwuche"
29.            >
30.              购物车
31.            <div class="cart-num">3</div>
32.          </a>
33.          <a href="/pages/manager/book_manager.html" class="admin">后台管理</a>
34.        </div>
35.      </div>
36.    </div>
37.    <div class="header w">
38.      <a href="#" class="header-logo"></a>
39.      <div class="header-nav">
40.        <ul>
41.          <li><a href="#">java</a></li>
42.          <li><a href="#">前端</a></li>
43.          <li><a href="#">小说</a></li>
44.          <li><a href="#">文学</a></li>
45.          <li><a href="#">青春文学</a></li>
46.          <li><a href="#">艺术</a></li>
47.          <li><a href="#">管理</a></li>
48.        </ul>
49.      </div>
50.      <div class="header-search">
51.        <input type="text" placeholder="十万个为什么" />
52.        <button class="iconfont icon-search"></button>
53.      </div>
```

```
54.         </div>
55.     <div class="banner w clearfix">
56.         <div class="banner-left">
57.             <ul>
58.                 <li>
59.                     <a href="">
60.                         <span>文学 鉴赏</span>
61.                         <i class="iconfont icon-jiantou"></i>
62.                     ></a>
63.                 </li>
64.                 <li>
65.                     <a href="">
66.                         <span>社科 研究</span>
67.                         <i class="iconfont icon-jiantou"></i>
68.                     ></a>
69.                 </li>
70.                 <li>
71.                     <a href="">
72.                         <span>少儿 培训</span>
73.                         <i class="iconfont icon-jiantou"></i>
74.                     ></a>
75.                 </li>
76.                 <li>
77.                     <a href="">
78.                         <span>艺术 赏析</span>
79.                         <i class="iconfont icon-jiantou"></i>
80.                     ></a>
81.                 </li>
82.                 <li>
83.                     <a href="">
84.                         <span>生活 周边</span>
85.                         <i class="iconfont icon-jiantou"></i>
86.                     ></a>
87.                 </li>
88.                 <li>
89.                     <a href="">
90.                         <span>文教 科技</span>
91.                         <i class="iconfont icon-jiantou"></i>
92.                     ></a>
93.                 </li>
94.                 <li>
95.                     <a href="">
96.                         <span>热销 畅读</span>
97.                         <i class="iconfont icon-jiantou"></i>
98.                     ></a>
99.                 </li>
100.            </ul>
101.        </div>
102.        <div class="banner-right">
103.            <div class="swiper-container">
104.                <ul class="swiper-wrapper">
105.                    <li class="swiper-slide">
106.                        
```

```

107.         <!-- <div class="banner-img"></div> -->
108.     </li>
109.     <li class="swiper-slide">
110.         
111.         <!-- <div class="banner-img"></div> -->
112.     </li>
113.     <li class="swiper-slide">
114.         
115.         <!-- <div class="banner-img"></div> -->
116.     </li>
117. </ul>
118. <div class="swiper-button-prev"></div>
119. <div class="swiper-button-next"></div>
120. <!-- Add Pagination -->
121. <div class="swiper-pagination"></div>
122. </div>
123. </div>
124. </div>
125. <div class="books-list ">
126.     <div class="w">
127.         <div class="list">
128.             <div class="list-header">
129.                 <div class="title">图书列表</div>
130.                 <div class="price-search">
131.                     <span>价格:</span>
132.                     <input type="text">
133.                     <span>-元</span>
134.                     <input type="text">
135.                     <span>元</span>
136.                     <button>查询</button>
137.                 </div>
138.             </div>
139.             <div class="list-content">
140.                 <div class="list-item">
141.                     
142.                     <p>书名:活着</p>
143.                     <p>作者:余华</p>
144.                     <p>价格:¥ 66.6</p>
145.                     <p>销量:230</p>
146.                     <p>库存:1000</p>
147.                     <button>加入购物车</button>
148.                 </div>
149.                 <div href="" class="list-item">
150.                     
151.                     <p>书名:活着</p>
152.                     <p>作者:余华</p>
153.                     <p>价格:¥ 66.6</p>
154.                     <p>销量:230</p>
155.                     <p>库存:1000</p>
156.                     <button>加入购物车</button>
157.                 </div>
158.                 <div href="" class="list-item">
159.                     

```

```
160.         <p>书名:活着</p>
161.         <p>作者:余华</p>
162.         <p>价格:¥ 66.6</p>
163.         <p>销量:230</p>
164.         <p>库存:1000</p>
165.         <button>加入购物车</button>
166.     </div>
167.     <div href="" class="list-item">
168.         
169.         <p>书名:活着</p>
170.         <p>作者:余华</p>
171.         <p>价格:¥ 66.6</p>
172.         <p>销量:230</p>
173.         <p>库存:1000</p>
174.         <button>加入购物车</button>
175.     </div>
176.     <div href="" class="list-item">
177.         
178.         <p>书名:活着</p>
179.         <p>作者:余华</p>
180.         <p>价格:¥ 66.6</p>
181.         <p>销量:230</p>
182.         <p>库存:1000</p>
183.         <button>加入购物车</button>
184.     </div>
185.     <div href="" class="list-item">
186.         
187.         <p>书名:活着</p>
188.         <p>作者:余华</p>
189.         <p>价格:¥ 66.6</p>
190.         <p>销量:230</p>
191.         <p>库存:1000</p>
192.         <button>加入购物车</button>
193.     </div>
194.     <div href="" class="list-item">
195.         
196.         <p>书名:活着</p>
197.         <p>作者:余华</p>
198.         <p>价格:¥ 66.6</p>
199.         <p>销量:230</p>
200.         <p>库存:1000</p>
201.         <button>加入购物车</button>
202.     </div>
203.     <div href="" class="list-item">
204.         
205.         <p>书名:活着</p>
206.         <p>作者:余华</p>
207.         <p>价格:¥ 66.6</p>
208.         <p>销量:230</p>
209.         <p>库存:1000</p>
210.         <button>加入购物车</button>
211.     </div>
212.     <div href="" class="list-item">
```

```

213.         
214.         <p>书名:活着</p>
215.         <p>作者:余华</p>
216.         <p>价格:¥ 66.6</p>
217.         <p>销量:230</p>
218.         <p>库存:1000</p>
219.         <button>加入购物车</button>
220.     </div>
221.     <div href="" class="list-item">
222.         
223.         <p>书名:活着</p>
224.         <p>作者:余华</p>
225.         <p>价格:¥ 66.6</p>
226.         <p>销量:230</p>
227.         <p>库存:1000</p>
228.         <button>加入购物车</button>
229.     </div>
230. </div>
231. <div class="list-footer">
232.     <div>首页</div>
233.     <div>上一页</div>
234.     <ul><li class="active">1</li><li>2</li><li>3</li></ul>
235.     <div>下一页</div>
236.     <div>末页</div>
237.     <span>共 10 页</span>
238.     <span>30 条记录</span>
239.     <span>到第</span>
240.     <input type="text">
241.     <span>页</span>
242.     <button>确定</button>
243. </div>
244. </div>
245. </div>
246. </div>
247. <div class="cate w">
248.     <div class="list">
249.         <a href="" class="list-item">
250.             <i class="iconfont icon-java"></i>
251.             <span>java</span>
252.         </a>
253.         <a href="" class="list-item">
254.             ><i class="iconfont icon-h5"></i>h5</a>
255.         >
256.         <a href="" class="list-item">
257.             <i class="iconfont icon-python"></i>python
258.         </a>
259.         <a href="" class="list-item">
260.             ><i class="iconfont icon-tianchongxing"></i>pm</a>
261.         >
262.         <a href="" class="list-item">
263.             ><i class="iconfont icon-php_elephant"></i>php</a>
264.         >
265.         <a href="" class="list-item">

```

```
266.         ><i class="iconfont icon-go"></i>go</a>
267.     >
268. </div>
269. <a href="" class="img">
270.     
271. </a>
272. <a href="" class="img">
273.     
274. </a>
275. <a href="" class="img">
276.     
277. </a>
278. </div>
279. <div class="books">
280.     <div class="w">
281.         <div class="seckill">
282.             <div class="seckill-header">
283.                 <div class="title">
284.                     图书秒杀
285.                 </div>
286.                 <!-- <i class="iconfont icon-huanyipi"></i> -->
287.             </div>
288.             <div class="seckill-content">
289.                 <a href="" class="tip">
290.                     <h5>距离结束还有</h5>
291.                     <i class="iconfont icon-shandian"></i>
292.                     <div class="downcount">
293.                         <span class="time">00</span>
294.                         <span class="token">:</span>
295.                         <span class="time">00</span>
296.                         <span class="token">:</span>
297.                         <span class="time">00</span>
298.                     </div>
299.                 </a>
300.                 <a href="" class="books-sec">
301.                     
302.                     <p>从晚晴到民国</p>
303.                     <div>
304.                         <span class="cur-price">¥ 28.9</span>
305.                         <span class="pre-price">¥ 36.5</span>
306.                     </div>
307.                     <button>立即购买</button>
308.                 </a>
309.                 <a href="" class="books-sec">
310.                     
311.                     <p>c 语言入门经典</p>
312.                     <div>
313.                         <span class="cur-price">¥ 55.9</span>
314.                         <span class="pre-price">¥ 68.5</span>
315.                     </div>
316.                     <button>立即购买</button>
317.                 </a>
318.                 <a href="" class="books-sec">
```

```

319.         
320.         <p>扶桑</p>
321.         <div>
322.             <span class="cur-price">¥ 30.9</span>
323.             <span class="pre-price">¥ 47.5</span>
324.         </div>
325.         <button>立即购买</button>
326.     </a>
327.     <a href="" class="books-sec">
328.         
329.         <p>给孩子的诗</p>
330.         <div>
331.             <span class="cur-price">¥ 18.9</span>
332.             <span class="pre-price">¥ 25.5</span>
333.         </div>
334.         <button>立即购买</button>
335.     </a>
336. </ul>
337. </div>
338. </div>
339. </div>
340. </div>
341. <div class="bottom">
342.     <div class="w">
343.         <div class="top">
344.             <ul>
345.                 <li>
346.                     <a href="">
347.                         
348.                         <span>大咖级讲师亲自授课</span>
349.                     </a>
350.                 </li>
351.                 <li>
352.                     <a href="">
353.                         
354.                         <span>课程为学员成长持续赋能</span>
355.                     </a>
356.                 </li>
357.                 <li>
358.                     <a href="">
359.                         
360.                         <span>学员真是情况大公开</span>
361.                     </a>
362.                 </li>
363.             </ul>
364.         </div>
365.         <div class="content">
366.             <dl>
367.                 <dt>关于</dt>
368.                 <dd>教育理念</dd>
369.                 <!-- <dd>名师团队</dd>
370.                 <dd>学员心声</dd> -->
371.             </dl>

```

```
372.         <dl>
373.             <dt>资源下载</dt>
374.             <dd>视频下载</dd>
375.             <!-- <dd>资料下载</dd>
376.             <dd>工具下载</dd> -->
377.         </dl>
378.         <dl>
379.             <dt>加入我们</dt>
380.             <dd>招聘岗位</dd>
381.             <!-- <dd>岗位介绍</dd>
382.             <dd>招贤纳士</dd> -->
383.         </dl>
384.     </div>
385. </div>
386. </div>
387. <script src="/static/script/swiper.min.js"></script>
388. <script>
389.     var swiper = new Swiper('.swiper-container', {
390.         autoplay: true,
391.         pagination: {
392.             el: '.swiper-pagination',
393.             dynamicBullets: true
394.         },
395.         navigation: {
396.             nextEl: '.swiper-button-next',
397.             prevEl: '.swiper-button-prev'
398.         }
399.     })
400. </script>
401. </body>
402. </html>
403. / cart.html
404. <!DOCTYPE html>
405. <html lang="en">
406.     <head>
407.         <meta charset="UTF-8" />
408.         <meta name="viewport" content="width=device-width, initial-scale=1.0" />
409.         <title>Document</title>
410.         <link rel="stylesheet" href="/static/css/minireset.css" />
411.         <link rel="stylesheet" href="/static/css/common.css" />
412.         <link rel="stylesheet" href="/static/css/cart.css" />
413.     </head>
414.     <body>
415.         <div class="header">
416.             <div class="w">
417.                 <div class="header-left">
418.                     <a href="/index.html">
419.                         </a>
421.                     <h1>我的购物车</h1>
422.                 </div>
423.                 <div class="header-right">
424.                     <h3>欢迎<span></span>光临书城</h3>
```



```

425.         <div class="order"><a href="../order/order.html">我的订单</a></div>
426.         <div class="destory"><a href="../index.html">注销</a></div>
427.         <div class="gohome">
428.             <a href="../index.html">返回</a>
429.         </div>
430.     </div>
431. </div>
432. </div>
433. <div class="list">
434.     <div class="w">
435.         <table>
436.             <thead>
437.                 <tr>
438.                     <th>图片</th>
439.                     <th>商品名称</th>
440.                     <th>数量</th>
441.                     <th>单价</th>
442.                     <th>金额</th>
443.                     <th>操作</th>
444.                 </tr>
445.             </thead>
446.             <tbody>
447.                 <tr>
448.                     <td>
449.                         
450.                     </td>
451.                     <td>活着</td>
452.                     <td>
453.                         <span class="count">-</span>
454.                         <input class="count-num" type="text" value="1" />
455.                         <span class="count">+</span>
456.                     </td>
457.                     <td>36.8</td>
458.                     <td>36.8</td>
459.                     <td><a href="">删除</a></td>
460.                 </tr>
461.                 <tr>
462.                     <td>
463.                         
464.                     </td>
465.                     <td>看见</td>
466.                     <td>
467.                         <span class="count">-</span>
468.                         <input class="count-num" type="text" value="1" />
469.                         <span class="count">+</span>
470.                     </td>
471.                     <td>40.1</td>
472.                     <td>40.1</td>
473.                     <td><a href="">删除</a></td>
474.                 </tr>
475.                 <tr>
476.                     <td>
477.                         

```

```

478.         </td>
479.         <td>
480.             看见
481.         </td>
482.         <td>
483.             <span class="count">-</span>
484.             <input class="count-num" type="text" value="1" />
485.             <span class="count">+</span>
486.         </td>
487.         <td>40.1</td>
488.         <td>40.1</td>
489.         <td><a href="">删除</a></td>
490.     </tr>
491.     <tr>
492.         <td>
493.             
494.         </td>
495.         <td>
496.             看见
497.         </td>
498.         <td>
499.             <!-- <div> -->
500.             <span class="count">-</span>
501.             <input class="count-num" type="text" value="1" />
502.             <span class="count">+</span>
503.             <!-- </div> -->
504.         </td>
505.         <td>40.1</td>
506.         <td>40.1</td>
507.         <td><a href="">删除</a></td>
508.     </tr>
509.     <tr>
510.         <td>
511.             
512.         </td>
513.         <td>
514.             看见
515.         </td>
516.         <td>
517.             <span class="count">-</span>
518.             <input class="count-num" type="text" value="1" />
519.             <span class="count">+</span>
520.         </td>
521.         <td>40.1</td>
522.         <td>40.1</td>
523.         <td><a href="">删除</a></td>
524.     </tr>
525. </tbody>
526. </table>
527. <div class="footer">
528.     <div class="footer-left">
529.         <a href="#" class="clear-cart">清空购物车</a>
530.         <a href="#">继续购物</a>

```

```

531.         </div>
532.         <div class="footer-right">
533.             <div>共<span>5</span>件商品</div>
534.             <div class="total-price">总金额<span>197.2</span>元</div>
535.             <a class="pay" href="checkout.html">去结账</a>
536.         </div>
537.     </div>
538. </div>
539. </div>
540. <div class="bottom">
541.     <div class="w">
542.         <div class="top">
543.             <ul>
544.                 <li>
545.                     <a href="">
546.                         
547.                         <span>大咖级讲师亲自授课</span>
548.                     </a>
549.                 </li>
550.                 <li>
551.                     <a href="">
552.                         
553.                         <span>课程为学员成长持续赋能</span>
554.                     </a>
555.                 </li>
556.                 <li>
557.                     <a href="">
558.                         
559.                         <span>学员真是情况大公开</span>
560.                     </a>
561.                 </li>
562.             </ul>
563.         </div>
564.         <div class="content">
565.             <dl>
566.                 <dt>关于我们</dt>
567.                 <dd>教育理念</dd>
568.                 <!-- <dd>名师团队</dd>
569.                 <dd>学员心声</dd> -->
570.             </dl>
571.             <dl>
572.                 <dt>资源下载</dt>
573.                 <dd>视频下载</dd>
574.                 <!-- <dd>资料下载</dd>
575.                 <dd>工具下载</dd> -->
576.             </dl>
577.             <dl>
578.                 <dt>加入我们</dt>
579.                 <dd>招聘岗位</dd>
580.                 <!-- <dd>岗位介绍</dd>
581.                 <dd>招贤纳士</dd> -->
582.             </dl>
583.         </div>

```



```

637.             </div>
638.         </div>
639.     </div>
640. </div>
641. <div id="main">
642.     <h1>你的订单已结算，订单号为:<span class="oid">546845626455846</span></h1>
643. </div>
644. </body>
645. </html>
646. / /book_add.html
647. <!DOCTYPE html>
648. <html>
649.     <head>
650.         <meta charset="UTF-8" />
651.         <meta name="viewport" content="width=device-width, initial-scale=1.0" />
652.         <title>Document</title>
653.         <link rel="stylesheet" href="../../static/css/minireset.css" />
654.         <link rel="stylesheet" href="../../static/css/common.css" />
655.         <link rel="stylesheet" href="../../static/css/style.css" />
656.         <link rel="stylesheet" href="../../static/css/cart.css" />
657.         <link rel="stylesheet" href="../../static/css/bookManger.css" />
658.         <link rel="stylesheet" href="../../static/css/register.css" />
659.         <link rel="stylesheet" href="../../static/css/book_edit.css" />
660.     </head>
661.     <body>
662.         <div class="header">
663.             <div class="w">
664.                 <div class="header-left">
665.                     <a href="../../index.html">
666.                         </a>
668.                     <h1>添加图书</h1>
669.                 </div>
670.                 <div class="header-right">
671.                     <a href="/book_manager.html" class="order">图书管理</a>
672.                     <a href="/order_manager.html" class="destory">订单管理</a>
673.                     <a href="../../index.html" class="gohome">返回商城</a>
674.                 </div>
675.             </div>
676.         </div>
677.         <div class="login_banner">
678.             <div class="register_form">
679.                 <form action="">
680.                     <div class="form-item">
681.                         <div>
682.                             <label>名称:</label>
683.                             <input type="text" placeholder="请输入名称" />
684.                         </div>
685.                         <span class="errMess" style="visibility: visible;"
686.                             >请输入正确的名称</span>
687.                     </div>
688.                 </div>
689.             <div class="form-item">

```

```

690.         <div>
691.             <label>价格:</label>
692.             <input type="number" placeholder="请输入价格" />
693.         </div>
694.         <span class="errMess">请输入正确数字</span>
695.     </div>
696.     <div class="form-item">
697.         <div>
698.             <label>作者:</label>
699.             <input type="text" placeholder="请输入作者" />
700.         </div>
701.         <span class="errMess">请输入正确作者</span>
702.     </div>
703.     <div class="form-item">
704.         <div>
705.             <label>销量:</label>
706.             <input type="number" placeholder="请输入销量" />
707.         </div>
708.         <span class="errMess">请输入正确销量</span>
709.     </div>
710.     <div class="form-item">
711.         <div>
712.             <label>库存:</label>
713.             <input type="number" placeholder="请输入库存" />
714.         </div>
715.         <span class="errMess">请输入正确库存</span>
716.     </div>
717.     <button class="btn">提交</button>
718. </form>
719. </div>
720. </div>
721. </body>
722. </html>
723. / /book_deit.html
724. <!DOCTYPE html>
725. <html>
726.     <head>
727.         <meta charset="UTF-8" />
728.         <meta name="viewport" content="width=device-width, initial-scale=1.0" />
729.         <title>Document</title>
730.         <link rel="stylesheet" href="../../static/css/minireset.css" />
731.         <link rel="stylesheet" href="../../static/css/common.css" />
732.         <link rel="stylesheet" href="../../static/css/style.css" />
733.         <link rel="stylesheet" href="../../static/css/cart.css" />
734.         <link rel="stylesheet" href="../../static/css/bookManger.css" />
735.         <link rel="stylesheet" href="../../static/css/register.css" />
736.         <link rel="stylesheet" href="../../static/css/book_edit.css" />
737.     </head>
738.     <body>
739.         <div class="header">
740.             <div class="w">
741.                 <div class="header-left">
742.                     <a href="../../index.html">

```

```
743.         </a>
745.     <h1>编辑图书</h1>
746. </div>
747. <div class="header-right">
748.     <a href="/book_manager.html" class="order">图书管理</a>
749.     <a href="/order_manager.html" class="destory">订单管理</a>
750.     <a href="../../index.html" class="gohome">返回商城</a>
751. </div>
752. </div>
753. </div>
754. <div class="login_banner">
755.     <div class="register_form">
756.         <form action="">
757.             <div class="form-item">
758.                 <div>
759.                     <label>名称:</label>
760.                     <input type="text" placeholder="请输入名称" />
761.                 </div>
762.                 <span class="errMess" style="visibility: visible;"
763.                     >请输入正确的名称</span>
764.                 >
765.             </div>
766.             <div class="form-item">
767.                 <div>
768.                     <label>价格:</label>
769.                     <input type="number" placeholder="请输入价格" />
770.                 </div>
771.                 <span class="errMess">请输入正确数字</span>
772.             </div>
773.             <div class="form-item">
774.                 <div>
775.                     <label>作者:</label>
776.                     <input type="text" placeholder="请输入作者" />
777.                 </div>
778.                 <span class="errMess">请输入正确作者</span>
779.             </div>
780.             <div class="form-item">
781.                 <div>
782.                     <label>销量:</label>
783.                     <input type="number" placeholder="请输入销量" />
784.                 </div>
785.                 <span class="errMess">请输入正确销量</span>
786.             </div>
787.             <div class="form-item">
788.                 <div>
789.                     <label>库存:</label>
790.                     <input type="number" placeholder="请输入库存" />
791.                 </div>
792.                 <span class="errMess">请输入正确库存</span>
793.             </div>
794.             <button class="btn">提交</button>
795.         </form>
```

```

796.     </div>
797.     </div>
798. </body>
799. </html>
800. //book_manager.html
801. <!DOCTYPE html>
802. <html lang="en">
803.     <head>
804.         <meta charset="UTF-8" />
805.         <meta name="viewport" content="width=device-width, initial-scale=1.0" />
806.         <title>Document</title>
807.         <link rel="stylesheet" href="../../static/css/minireset.css" />
808.         <link rel="stylesheet" href="../../static/css/common.css" />
809.         <link rel="stylesheet" href="../../static/css/cart.css" />
810.         <link rel="stylesheet" href="../../static/css/bookManger.css" />
811.     </head>
812.     <body>
813.         <div class="header">
814.             <div class="w">
815.                 <div class="header-left">
816.                     <a href="../../index.html">
817.                         </a>
819.                     <h1>图书管理系统</h1>
820.                 </div>
821.                 <div class="header-right">
822.                     <a href="#" class="order">图书管理</a>
823.                     <a href="/order_manager.html" class="destory">订单管理</a>
824.                     <a href="../../index.html" class="gohome">返回商城</a>
825.                 </div>
826.             </div>
827.         </div>
828.         <div class="list">
829.             <div class="w">
830.                 <div class="add">
831.                     <a href="book_edit.html">添加图书</a>
832.                 </div>
833.                 <table>
834.                     <thead>
835.                         <tr>
836.                             <th>图片</th>
837.                             <th>商品名称</th>
838.                             <th>价格</th>
839.                             <th>作者</th>
840.                             <th>销量</th>
841.                             <th>库存</th>
842.                             <th>操作</th>
843.                         </tr>
844.                     </thead>
845.                     <tbody>
846.                         <tr>
847.                             <td>
848.                                 

```



```

849.         </td>
850.         <td>活着</td>
851.         <td>
852.             100.00
853.         </td>
854.         <td>余华</td>
855.         <td>200</td>
856.         <td>400</td>
857.         <td>
858.             <a href="book_edit.html">修改</a><a href="" class="del">删除</a>
859.         </td>
860.     </tr>
861. <tr>
862.     <td>
863.         
864.     </td>
865.     <td>活着</td>
866.     <td>
867.         100.00
868.     </td>
869.     <td>余华</td>
870.     <td>200</td>
871.     <td>400</td>
872.     <td>
873.         <a href="book_edit.html">修改</a><a href="" class="del">删除</a>
874.     </td>
875. </tr>
876. </tbody>
877. </table>
878. <div class="footer">
879.     <div class="footer-right">
880.         <div>首页</div>
881.         <div>上一页</div>
882.         <ul>
883.             <li class="active">1</li>
884.             <li>2</li>
885.             <li>3</li>
886.         </ul>
887.         <div>下一页</div>
888.         <div>末页</div>
889.         <span>共 10 页</span>
890.         <span>30 条记录</span>
891.         <span>到第</span>
892.         <input type="text" />
893.         <span>页</span>
894.         <button>确定</button>
895.     </div>
896. </div>
897. </div>
898. </div>
899. <div class="bottom">
900.     <div class="w">
901.         <div class="top">

```

```
902.         <ul>
903.             <li>
904.                 <a href="">
905.                     
906.                     <span>大咖级讲师亲自授课</span>
907.                 </a>
908.             </li>
909.             <li>
910.                 <a href="">
911.                     
912.                     <span>课程为学员成长持续赋能</span>
913.                 </a>
914.             </li>
915.             <li>
916.                 <a href="">
917.                     
918.                     <span>学员真是情况大公开</span>
919.                 </a>
920.             </li>
921.         </ul>
922.     </div>
923.     <div class="content">
924.         <dl>
925.             <dt>关于</dt>
926.             <dd>教育理念</dd>
927.             <!-- <dd>名师团队</dd>
928.             <dd>学员心声</dd> -->
929.         </dl>
930.         <dl>
931.             <dt>资源下载</dt>
932.             <dd>视频下载</dd>
933.             <!-- <dd>资料下载</dd>
934.             <dd>工具下载</dd> -->
935.         </dl>
936.         <dl>
937.             <dt>加入我们</dt>
938.             <dd>招聘岗位</dd>
939.             <!-- <dd>岗位介绍</dd>
940.             <dd>招贤纳士</dd> -->
941.         </dl>
942.     </div>
943. </div>
944. </div>
945. </div>
946. </body>
947. </html>
948. //manager.html
949. <!DOCTYPE html>
950. <html>
951. <head>
952. <meta charset="UTF-8">
953. <title>后台管理</title>
954. <link type="text/css" rel="stylesheet" href="../../static/css/style.css" >
```

```
955. <style type="text/css">
956.     h1 {
957.         text-align: center;
958.         margin-top: 200px;
959.     }
960. </style>
961. </head>
962. <body>
963.     <div id="header">
964.         
965.         <span class="wel_word">后台管理系统</span>
966.     <div>
967.         <a href="book_manager.html">图书管理</a>
968.         <a href="order_manager.html">订单管理</a>
969.         <a href="../../index.html">返回商城</a>
970.     </div>
971. </div>
972. <div id="main">
973.     <h1>欢迎管理员进入后台管理系统</h1>
974. </div>
975. </body>
976. </html>
977. //order_manager.html
978. <!DOCTYPE html>
979. <html lang="en">
980. <head>
981.     <meta charset="UTF-8" />
982.     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
983.     <title>Document</title>
984.     <link rel="stylesheet" href="../../static/css/minireset.css" />
985.     <link rel="stylesheet" href="../../static/css/common.css" />
986.     <link rel="stylesheet" href="../../static/css/cart.css" />
987.     <link rel="stylesheet" href="../../static/css/bookManger.css" />
988.     <link rel="stylesheet" href="../../static/css/orderManger.css" />
989. </head>
990. <body>
991.     <div class="header">
992.         <div class="w">
993.             <div class="header-left">
994.                 <a href="../../index.html">
995.                     </a>
997.                 <h1>订单管理系统</h1>
998.             </div>
999.             <div class="header-right">
1000.                 <a href="/book_manager.html" class="order">图书管理</a>
1001.                 <a href="#" class="destory">订单管理</a>
1002.                 <a href="../../index.html" class="gohome">返回商城</a>
1003.             </div>
1004.         </div>
1005.     </div>
1006.     <div class="list">
1007.         <div class="w">
```

```

1008.      <table>
1009.          <thead>
1010.              <tr>
1011.                  <th>订单号</th>
1012.                  <th>订单日期</th>
1013.                  <th>订单金额</th>
1014.                  <th>订单数量</th>
1015.                  <th>订单详情</th>
1016.                  <th>发货状态</th>
1017.              </tr>
1018.          </thead>
1019.          <tbody>
1020.              <tr>
1021.                  <td>12354456895</td>
1022.                  <td>
1023.                      2015.04.23
1024.                  </td>
1025.                  <td>90.00</td>
1026.                  <td>88</td>
1027.                  <td><a href="">查看详情</a></td>
1028.                  <td><a href="" class="send">等待发货</a></td>
1029.              </tr>
1030.              <tr>
1031.                  <td>12354456895</td>
1032.                  <td>
1033.                      2015.04.23
1034.                  </td>
1035.                  <td>90.00</td>
1036.                  <td>88</td>
1037.                  <td><a href="">查看详情</a></td>
1038.                  <td><a href="" class="send">已发货</a></td>
1039.              </tr>
1040.              <tr>
1041.                  <td>12354456895</td>
1042.                  <td>
1043.                      2015.04.23
1044.                  </td>
1045.                  <td>90.00</td>
1046.                  <td>88</td>
1047.                  <td><a href="">查看详情</a></td>
1048.                  <td><a href="">点击发货</a></td>
1049.              </tr>
1050.          </tbody>
1051.      </table>
1052.      <div class="footer">
1053.          <div class="footer-right">
1054.              <div>首页</div>
1055.              <div>上一页</div>
1056.              <ul>
1057.                  <li class="active">1</li>
1058.                  <li>2</li>
1059.                  <li>3</li>
1060.              </ul>

```

```

1061.      <div>下一页</div>
1062.      <div>末页</div>
1063.      <span>共 10 页</span>
1064.      <span>30 条记录</span>
1065.      <span>到第</span>
1066.      <input type="text" />
1067.      <span>页</span>
1068.      <button>确定</button>
1069.  </div>
1070. </div>
1071. </div>
1072. </div>
1073. <div class="bottom">
1074.   <div class="w">
1075.    <div class="top">
1076.     <ul>
1077.      <li>
1078.       <a href="">
1079.        
1080.        <span>大咖级讲师亲自授课</span>
1081.      </a>
1082.    </li>
1083.    <li>
1084.     <a href="">
1085.      
1086.      <span>课程为学员成长持续赋能</span>
1087.    </a>
1088.   </li>
1089.   <li>
1090.    <a href="">
1091.     
1092.     <span>学员真是情况大公开</span>
1093.   </a>
1094.  </li>
1095. </ul>
1096. </div>
1097. <div class="content">
1098.   <dl>
1099.    <dt>关于</dt>
1100.    <dd>教育理念</dd>
1101.    <!-- <dd>名师团队</dd>
1102.    <dd>学员心声</dd> -->
1103.  </dl>
1104.  <dl>
1105.   <dt>资源下载</dt>
1106.   <dd>视频下载</dd>
1107.   <!-- <dd>资料下载</dd>
1108.   <dd>工具下载</dd> -->
1109.  </dl>
1110.  <dl>
1111.   <dt>加入我们</dt>
1112.   <dd>招聘岗位</dd>
1113.   <!-- <dd>岗位介绍</dd>

```

```
1114.         <dd>招贤纳士</dd> -->
1115.     </dl>
1116. </div>
1117. </div>
1118. </div>
1119. </body>
1120. </html>
1121. //order.html
1122. <!DOCTYPE html>
1123. <html lang="en">
1124.     <head>
1125.         <meta charset="UTF-8" />
1126.         <meta name="viewport" content="width=device-width, initial-scale=1.0" />
1127.         <title>我的订单</title>
1128.         <link rel="stylesheet" href="../../static/css/minireset.css" />
1129.         <link rel="stylesheet" href="../../static/css/common.css" />
1130.         <link rel="stylesheet" href="../../static/css/cart.css" />
1131.         <link rel="stylesheet" href="../../static/css/bookManger.css" />
1132.         <link rel="stylesheet" href="../../static/css/orderManger.css" />
1133.     </head>
1134.     <body>
1135.         <div class="header">
1136.             <div class="w">
1137.                 <div class="header-left">
1138.                     <a href="../../index.html">
1139.                         </a>
1141.                     <h1>我的订单</h1>
1142.                 </div>
1143.                 <div class="header-right">
1144.                     <h3>欢迎<span>张总</span>光临书城</h3>
1145.                     <div class="order"><a href="../../order/order.html">我的订单</a></div>
1146.                     <div class="destory"><a href="../../index.html">注销</a></div>
1147.                     <div class="gohome">
1148.                         <a href="../../index.html">返回</a>
1149.                     </div>
1150.                 </div>
1151.             </div>
1152.         </div>
1153.         <div class="list">
1154.             <div class="w">
1155.                 <table>
1156.                     <thead>
1157.                         <tr>
1158.                             <th>订单号</th>
1159.                             <th>订单日期</th>
1160.                             <th>订单金额</th>
1161.                             <th>订单数量</th>
1162.                             <th>订单状态</th>
1163.                             <th>订单详情</th>
1164.                         </tr>
1165.                     </thead>
1166.                     <tbody>
```

```

1167.      <tr>
1168.          <td>12354456895</td>
1169.          <td>
1170.              2015.04.23
1171.          </td>
1172.          <td>90.00</td>
1173.          <td>88</td>
1174.          <td><a href="" class="send">等待发货</a></td>
1175.          <td><a href="">查看详情</a></td>
1176.      </tr>
1177.      <tr>
1178.          <td>12354456895</td>
1179.          <td>
1180.              2015.04.23
1181.          </td>
1182.          <td>90.00</td>
1183.          <td>88</td>
1184.          <td><a href="" class="send">未发货</a></td>
1185.          <td><a href="">查看详情</a></td>
1186.      </tr>
1187.      <tr>
1188.          <td>12354456895</td>
1189.          <td>
1190.              2015.04.23
1191.          </td>
1192.          <td>90.00</td>
1193.          <td>88</td>
1194.          <td><a href="" class="send">已发货</a></td>
1195.          <td><a href="">查看详情</a></td>
1196.      </tr>
1197.  </tbody>
1198. </table>
1199. <div class="footer">
1200.     <div class="footer-right">
1201.         <div>首页</div>
1202.         <div>上一页</div>
1203.         <ul>
1204.             <li class="active">1</li>
1205.             <li>2</li>
1206.             <li>3</li>
1207.         </ul>
1208.         <div>下一页</div>
1209.         <div>末页</div>
1210.         <span>共 10 页</span>
1211.         <span>30 条记录</span>
1212.         <span>到第</span>
1213.         <input type="text" />
1214.         <span>页</span>
1215.         <button>确定</button>
1216.     </div>
1217. </div>
1218. </div>
1219. </div>

```

```

1220.     <div class="bottom">
1221.         <div class="w">
1222.             <div class="top">
1223.                 <ul>
1224.                     <li>
1225.                         <a href="">
1226.                             
1227.                             <span>大咖级讲师亲自授课</span>
1228.                         </a>
1229.                     </li>
1230.                     <li>
1231.                         <a href="">
1232.                             
1233.                             <span>课程为学员成长持续赋能</span>
1234.                         </a>
1235.                     </li>
1236.                     <li>
1237.                         <a href="">
1238.                             
1239.                             <span>学员真是情况大公开</span>
1240.                         </a>
1241.                     </li>
1242.                 </ul>
1243.             </div>
1244.             <div class="content">
1245.                 <dl>
1246.                     <dt>关于</dt>
1247.                     <dd>教育理念</dd>
1248.                     <!-- <dd>名师团队</dd>
1249.                     <dd>学员心声</dd> -->
1250.                 </dl>
1251.                 <dl>
1252.                     <dt>资源下载</dt>
1253.                     <dd>视频下载</dd>
1254.                     <!-- <dd>资料下载</dd>
1255.                     <dd>工具下载</dd> -->
1256.                 </dl>
1257.                 <dl>
1258.                     <dt>加入我们</dt>
1259.                     <dd>招聘岗位</dd>
1260.                     <!-- <dd>岗位介绍</dd>
1261.                     <dd>招贤纳士</dd> -->
1262.                 </dl>
1263.             </div>
1264.         </body>
1265. </html>
1266. /login.html
1267. <!DOCTYPE html>
1268. <html>
1269.     <head>
1270.         <meta charset="UTF-8" />
1271.         <title>会员登录页面</title>
1272.         <link type="text/css" rel="stylesheet" href="../../static/css/style.css" />

```



```
1273. </head>
1274. <body>
1275.   <div id="login_header">
1276.     <a href="../../index.html">
1277.       
1278.     </a>
1279.   </div>
1280.   <div class="login_banner">
1281.     <div id="l_content">
1282.       <span class="login_word">欢迎登录</span>
1283.     </div>
1284.     <div id="content">
1285.       <div class="login_form">
1286.         <div class="login_box">
1287.           <div class="tit">
1288.             <h1>会员</h1>
1289.           </div>
1290.           <div class="msg_cont">
1291.             <b></b>
1292.             <span class="errorMsg">请输入用户名和密码</span>
1293.           </div>
1294.           <div class="form">
1295.             <form action="login_success.html">
1296.               <label>用户名称: </label>
1297.               <input
1298.                 class="itxt"
1299.                 type="text"
1300.                 placeholder="请输入用户名"
1301.                 autocomplete="off"
1302.                 tabindex="1"
1303.                 name="username"
1304.                 id="username"
1305.               />
1306.               <br />
1307.               <br />
1308.               <label>用户密码: </label>
1309.               <input
1310.                 class="itxt"
1311.                 type="password"
1312.                 placeholder="请输入密码"
1313.                 autocomplete="off"
1314.                 tabindex="1"
1315.                 name="password"
1316.                 id="password"
1317.               />
1318.               <br />
1319.               <br />
1320.               <input type="submit" value="登录" id="sub_btn" />
1321.             </form>
1322.             <div class="tit">
1323.               <a href="regist.html">立即注册</a>
1324.             </div>
1325.           </div>
```



```
1379.         margin-top: 25px;
1380.     }
1381. </style>
1382. </head>
1383. <body>
1384.     <div id="login_header">
1385.         <a href="../../index.html">
1386.             
1387.         </a>
1388.     </div>
1389.     <div class="login_banner">
1390.         <div class="register_form">
1391.             <h1>注册会员</h1>
1392.             <form action="regist_success.html">
1393.                 <div class="form-item">
1394.                     <div>
1395.                         <label>用户名称:</label>
1396.                         <input type="text" placeholder="请输入用户名" />
1397.                     </div>
1398.                     <span class="errMsg">用户名应为 6~16 位数组和字母组成</span>
1399.                 </div>
1400.                 <div class="form-item">
1401.                     <div>
1402.                         <label>用户密码:</label>
1403.                         <input type="password" placeholder="请输入密码" />
1404.                     </div>
1405.                     <span class="errMsg">密码的长度至少为 8 位</span>
1406.                 </div>
1407.                 <div class="form-item">
1408.                     <div>
1409.                         <label>确认密码:</label>
1410.                         <input type="password" placeholder="请输入确认密码" />
1411.                     </div>
1412.                     <span class="errMsg">密码两次输入不一致</span>
1413.                 </div>
1414.                 <div class="form-item">
1415.                     <div>
1416.                         <label>用户邮箱:</label>
1417.                         <input type="text" placeholder="请输入邮箱" />
1418.                     </div>
1419.                     <span class="errMsg">请输入正确的邮箱格式</span>
1420.                 </div>
1421.                 <div class="form-item">
1422.                     <div>
1423.                         <label>验证码:</label>
1424.                         <div class="verify">
1425.                             <input type="text" placeholder="" />
1426.                             
1427.                         </div>
1428.                     </div>
1429.                     <span class="errMsg">请输入正确的验证码</span>
1430.                 </div>
1431.             </form>
1432.         </div>
1433.     </div>
1434. </body>
1435. </html>
```



```
1485. margin: 0;
1486. }
1487. .login_banner .register_form {
1488. padding: 30px 0;
1489. }
1490. .login_banner .register_form .form-item {
1491. margin-top: 10px !important;
1492. }
1493. .bottom {
1494. margin: 0;
1495. background-color: #fff;
1496. color: #333;
1497. margin-top: 10px;
1498. }
1499. .header-right a {
1500. font-size: 20px;
1501. color: #39987c;
1502. }
1503. .list input {
1504. text-align: center;
1505. }
1506. .list .add {
1507. padding: 10px 0;
1508. }
1509. .list .add a {
1510. width: 80px;
1511. height: 30px;
1512. background-color: #39987c;
1513. color: #fff;
1514. border: none;
1515. outline: none;
1516. text-align: center;
1517. line-height: 30px;
1518. }
1519. .list table th {
1520. height: 40px !important;
1521. }
1522. .list table td {
1523. padding: 5px 0 !important;
1524. }
1525. .list table td img {
1526. width: 60px !important;
1527. height: 60px !important;
1528. }
1529. .list table td a {
1530. display: inline;
1531. color: #39987c;
1532. }
1533. .list table td .del {
1534. margin-left: 10px;
1535. color: red;
1536. }
1537. .list .footer {
```

```
1538. margin-top: 3px !important;
1539. justify-content: flex-end !important;
1540. }
1541. .list .footer-right {
1542. height: 50px;
1543. display: flex;
1544. justify-content: flex-end;
1545. align-items: center;
1546. color: #666;
1547. font-size: 20px;
1548. }
1549. .list .footer-right div {
1550. margin: 0 5px;
1551. cursor: pointer;
1552. text-decoration: underline;
1553. }
1554. .list .footer-right span {
1555. margin: 0 5px !important;
1556. font-size: 20px !important;
1557. color: #666 !important;
1558. }
1559. .list .footer-right ul li {
1560. float: left;
1561. cursor: pointer;
1562. text-decoration: underline;
1563. }
1564. .list .footer-right ul li:nth-child(n + 2) {
1565. margin-left: 10px;
1566. }
1567. .list .footer-right ul li.active {
1568. font-weight: 700;
1569. color: #39987c;
1570. font-size: 24px;
1571. }
1572. .list .footer-right input {
1573. width: 40px;
1574. height: 30px;
1575. }
1576. .list .footer-right button {
1577. width: 60px;
1578. height: 30px;
1579. border: none;
1580. outline: none;
1581. background-color: #39987c;
1582. color: #fff;
1583. margin-left: 10px;
1584. cursor: pointer;
1585. }
1586. .header {
1587. height: 100px;
1588. border-bottom: 2px solid #39987c;
1589. }
1589. .header .w {
```

```

7600.         }
7601.     })
7602.     that.sysRoleId = that.localdata[0].value
7879.     <uni-pagination :current="that.pageNumber" :pageSize="that.pageSize" :total="that.total"
@change="changePage"/>
7880.     <uni-popup ref="inputDialog" type="dialog">
7881.         <uni-popup-dialog ref="inputClose" mode="input" :title="用户详情" @confirm="dialogInputConfirm">
7882.             <uni-forms :model="that.userDetail" label-width="84px">
7883.                 <uni-forms-item label="账号: ">
7884.                     <uni-easyinput v-model="that.userDetail.userAccount" type="text" placeholder="请输入">
7885.                         </uni-easyinput>
7886.                     </uni-forms-item>
7887.                     <uni-forms-item label="昵称: ">
7888.                         <uni-easyinput v-model="that.userDetail.nickname" type="text" placeholder="请输入">
7889.                             </uni-easyinput>
7890.                         </uni-forms-item>
7891.                         <uni-forms-item label="真实姓名: ">
7892.                             <uni-easyinput v-model="that.userDetail.realName" type="text" placeholder="请输入">
7893.                                 </uni-easyinput>
7894.                             </uni-forms-item>
7895.                             <uni-forms-item label="手机号: ">
7896.                                 <uni-easyinput v-model="that.userDetail.mobilePhone" type="text" placeholder="请输入">
7897.                                     </uni-easyinput>
7898.                                 </uni-forms-item>
7899.                                 <uni-forms-item label="身份证号: ">
7900.                                     <uni-easyinput v-model="that.userDetail.idCardNumber" type="text" placeholder="请输入">
7901.                                         </uni-easyinput>
7902.                                     </uni-forms-item>
7903.                                     <uni-forms-item label="所属位置: ">
7904.                                         <uni-easyinput v-model="that.userDetail.address" type="text" :disabled="true" :clearable="true"
7905.                                             placeholder="请选择" prefixIcon="map-pin-ellipse"
@iconClick="getLocation()">
7906.                                             </uni-easyinput>
7907.                                         </uni-forms-item>
7908.                                         <uni-forms-item label="用户身份: ">
7909.                                             <uni-data-select v-model="that.userDetail.sysRoleId"
7910.                                                 :localdata="localdata.sysRole"
7911.                                                 placeholder="请选择"/>
7912.                                             </uni-forms-item>
7913.                                             <uni-forms-item v-if="that.userDetail.sysRoleId===4||that.userDetail.sysRoleId===5" label="可操作杆:
">
7914.                                                 <uni-data-checkbox v-model="that.accessSysRoleIdArray" multiple
7915.                                                     :localdata="localdata.sysRod"/>
7916.                                                 </uni-forms-item>

```

```
7917.             <uni-forms-item label="状态: ">
7918.                 <uni-data-checkbox v-model="that.userDetail.status"
7919.                                     :localdata="localdata.status"/>
7920.             </uni-forms-item>
7921.         </uni-forms>
7922.     </uni-popup-dialog>
7923. </uni-popup>
7924. </template>
7925. <script setup>
7926. import {onMounted, reactive, ref} from 'vue'
7927. import {delete_userSysUserId, get_user, put_userSysUserId} from "@/static/js/sysUserApi";
7928. import {get_role} from "@/static/js/sysRoleApi";
7929. import {get_rod} from "@/static/js/sysRodApi";
7930. import {delete_userRod, get_userRod, post_userRod} from "@/static/js/sysUserRodApi";
7931. const inputDialog = ref()
7932. const that = reactive({
7933.     pageNumber: 1,
7934.     pageSize: 5,
7935.     total: 0,
7936.     tableData: [],
7937.     userDetail: {},
7938.     accessSysRodIdArray: [],
7939. })
7940. const localdata = reactive({
7941.     sysRole: [],
7942.     sysRod: [],
7943.     status: [{
7944.         text: '禁用',
7945.         value: 0
7946.     }, {
7947.         text: '启用',
7948.         value: 1
7949.     }],
7950. })
7951. function changePage(e) {
7952.     console.log(e.type, e.current)
7953.     that.pageNumber = e.current
7954.     pagingQueryFile()
7955. }
7956. function getLocation() {
7957.     uni.chooseLocation({
7958.         success: function (res) {
7959.             console.log('位置名称: ' + res.name);
7960.             console.log('详细地址: ' + res.address);
```



```
7961.         console.log('纬度: ' + res.latitude);
7962.         console.log('经度: ' + res.longitude);
7963.         that.userDetail.address = res.address
7964.     }
7965. });
7966. }
7967. function pagingQueryFile() {
7968.     get_user().then(data => {
7969.         // that.total = data.total
7970.         that.tableData = data
7971.     })
7972. }
7973. function handleEdit(row) {
7974.     // 获取权限列表
7975.     get_role().then(data => {
7976.         localdata.sysRole = data.map(item => {
7977.             return {
7978.                 value: item.id,
7979.                 text: item.name
7980.             }
7981.         })
7982.     })
7983.     // 查询所有杆子
7984.     get_rod(true).then(data => {
7985.         localdata.sysRod = data.map(item => {
7986.             return {
7987.                 value: item.id,
7988.                 text: item.projectName
7989.             }
7990.         })
7991.     })
7992.     console.log(row)
7993.     that.userDetail = row
7994.     // 获取可用杆子权限
7995.     get_userRod(that.userDetail.id).then(data => {
7996.         console.log("get_userRod", data)
7997.         that.accessSysRodIdArray = data.map(item => item.sysRodId)
7998.     })
7999.     inputDialog.value.open()
8000. }
8001. function dialogInputConfirm() {
8002.     console.log('点击 dialogInputConfirm()', that.userDetail)
8003.     uni.showModal({
8004.         title: '系统提示',
```

```
8005.         content: '确认更新用户【' + that.userDetail.nickname + '】的信息吗? ! ',
8006.         success: function (res) {
8007.             if (res.confirm) {
8008.                 console.log('用户点击确定');
8009.                 that.userDetail.userPassword = null
8010.                 const sysUserId = that.userDetail.id;
8011.                 put_userSysUserId(sysUserId, that.userDetail).then(data => {
8012.                     // TODO 批量更新用户杆子
8013.                     batch_update_userRod(sysUserId);
8014.                     // 刷新页面
8015.                     pagingQueryFile()
8016.                 })
8017.             } else if (res.cancel) {
8018.                 console.log('用户点击取消');
8019.             }
8020.         }
8021.     });
8022. }
8023. /**
8024.  * TODO 批量更新用户杆子
8025.  */
8026. function batch_update_userRod(sysUserId) {
8027.     get_userRod(sysUserId).then(data => {
8028.         data.forEach(data => {
8029.             delete_userRod({sysUserId: sysUserId, sysRodId: data.sysRodId})
8030.         })
8031.     })
8032.     setTimeout(() => {
8033.         that.accessSysRodIdArray.forEach(sysRodId => {
8034.             post_userRod({sysUserId: sysUserId, sysRodId: sysRodId})
8035.         })
8036.     }, 500)
8037. }
8038. function handleDelete(row) {
8039.     console.log(row)
8040.     uni.showModal({
8041.         title: '系统提示',
8042.         content: '确认删除用户【' + row.nickname + '】吗? ! ',
8043.         success: function (res) {
8044.             if (res.confirm) {
8045.                 console.log('用户点击确定');
8046.                 delete_userSysUserId(row.id).then(data => {
8047.                     pagingQueryFile()
8048.                 })
```

```

8049.         } else if (res.cancel) {
8050.             console.log('用户点击取消');
8051.         }
8052.     }
8053. });
8054. }
8055. onMounted(() => {
8056.     console.log('mounted...');
8057.     // 加载表格数据
8058.     pagingQueryFile()
8059. })
8060. </script>
8061. <template>
8062.     <camera-panel :sysRodId="that.sysRodId" @changePresetId="onChangePresetId">
8063.         <uni-section title="配置" type="line">
8064.             <view class="example">
8065.                 <uni-forms :model="form" label-width="120px" label-position="top">
8066.                     <uni-forms-item label="文件名">
8067.                         <uni-easyinput v-model="form.filename" type="text" placeholder="默认为当前时
间"/>
8068.                     </uni-forms-item>
8069.                     <uni-forms-item label="日期区间: " required>
8070.                         <uni-datetime-picker v-model="form.dateRange" type="daterange" placeholder="
请选择"
8071.                                     :clear-icon="false"/>
8072.                     </uni-forms-item>
8073.                     <uni-forms-item label="开始时间: " required>
8074.                         <view class="uni-list-cell-db">
8075.                             <picker mode="time" :value="form.startTime"
8076.                                     @change="bindFromChange">
8077.                                 <uni-easyinput v-model="form.startTime" placeholder="请选择"/>
8078.                             </picker>
8079.                         </view>
8080.                     </uni-forms-item>
8081.                     <uni-forms-item label="结束时间: " required>
8082.                         <view class="uni-list-cell-db">
8083.                             <picker mode="time" :value="form.endTime"
8084.                                     @change="bindToChange">
8085.                                 <uni-easyinput v-model="form.endTime" placeholder="请选择"/>
8086.                             </picker>
8087.                         </view>
8088.                     </uni-forms-item>
8089.                     <!-- <uni-forms-item label="片段: ">-->
8090.                     <!-- <uni-easyinput v-model="form.sliceDuration"

```

```

type="number">-->
8091.                                     <!--                                <template #suffixIcon>分</template>-->
8092.                                     <!--                                </uni-easyinput>-->
8093.                                     <!--                                </uni-forms-item>-->
8094.                                     <uni-forms-item label="码流： ">
8095.                                         <uni-data-select v-model="form.streamType" :localdata="that.streamTypeOption"
8096.                                             placeholder="请选择"/>
8097.                                     </uni-forms-item>
8098.                                     <uni-forms-item label="声音： ">
8099.                                         <uni-data-select
v-model="form.voiceSwitch" :localdata="that.voiceSwitchOption"
8100.                                             placeholder="请选择"/>
8101.                                     </uni-forms-item>
8102.                                     <button type="primary" @click="onSubmit()">提交</button>
8103.                                 </uni-forms>
8104.                             </view>
8105.                         </uni-section>
8106.                    </camera-panel>
8107. </template>
8108. <script setup>
8109. import {
8110.     reactive
8111. } from 'vue'
8112. import {
8113.     axios
8114. } from '@static/js/api.js'
8115. import {
8116.     onLoad
8117. } from '@dcloudio/uni-app';
8118. import UniForms from "../uni_modules/uni-forms/components/uni-forms/uni-forms.vue";
8119. import UniSection from "../uni_modules/uni-section/components/uni-section/uni-section.vue";
8120. import UniDatetimePicker
8121.     from "../uni_modules/uni-datetime-picker/components/uni-datetime-picker/uni-datetime-picker.vue";
8122. import CameraPanel from "../online_preview/camera/CameraPanel.vue";
8123. import UniFormsItem from "../uni_modules/uni-forms/components/uni-forms-item/uni-forms-item.vue";
8124. const props = defineProps({
8125.     sysRodId: Number
8126. })
8127. const that = reactive({
8128.     sysRodId: Number,
8129.     streamTypeOption: [{
8130.         value: 1,
8131.         text: "高清"
8132.     },

```

```
8133.      {
8134.          value: 2,
8135.          text: "标清"
8136.      },
8137.  ],
8138.  voiceSwitchOption: [{
8139.      value: 0,
8140.      text: "关"
8141.  },
8142.  {
8143.      value: 1,
8144.      text: "开"
8145.  },
8146.  {
8147.      value: 2,
8148.      text: "自动"
8149.  },
8150.  ],
8151. })
8152. onLoad((option) => {
8153.    that.sysRodId = Number(option.sysRodId)
8154.    form.sysRodId = Number(option.sysRodId)
8155.    console.log(form)
8156. })
8157. function onChangePresetId(payload) {
8158.    form.presetId = payload
8159.    console.log(form)
8160. }
8161. const form = reactive({
8162.    dateRange: [],
8163.    from: null,
8164.    to: null,
8165.    sysRodId: Number,
8166.    presetId: Number,
8167.    startTime: null,
8168.    endTime: null,
8169.    sliceDuration: 30,
8170.    aiBox: 0,
8171.    streamType: 1,
8172.    voiceSwitch: 2,
8173. })
8174. function bindFromChange(e) {
8175.    form.startTime = e.detail.value
8176. }
```

```
8177. function bindToChange(e) {
8178.     form.endTime = e.detail.value
8179. }
8180. function onSubmit() {
8181.     console.log('submit!')
8182.     form.from = form.dateRange[0]
8183.     form.to = form.dateRange[1]
8184.     console.log(form)
8185.     uni.showModal({
8186.         title: '警告',
8187.         content: '请再次确认是否提交! ',
8188.         success: function (res) {
8189.             if (res.confirm) {
8190.                 uni.showLoading({
8191.                     'title': "正在操作"
8192.                 })
8193.                 axios({
8194.                     method: 'POST',
8195.                     url: '/cloud/video/schedule/records',
8196.                     params: {
8197.                         sysRodId: form.sysRodId,
8198.                         presetId: form.presetId,
8199.                         from: form.from,
8200.                         to: form.to,
8201.                         startTime: "1998-02-22 " + form.startTime + ":00",
8202.                         endTime: "1998-02-22 " + form.endTime + ":00",
8203.                         sliceDuration: form.sliceDuration,
8204.                         aiBox: form.aiBox,
8205.                         streamType: form.streamType,
8206.                         voiceSwitch: form.voiceSwitch,
8207.                         filename: form.filename,
8208.                     }
8209.                 }).then(({data}) => {
8210.                     uni.hideLoading()
8211.                     if (data.code === 200) {
8212.                         uni.showToast({
8213.                             title: data.data
8214.                         });
8215.                         uni.navigateBack({})
8216.                     }else{
8217.                         uni.showToast({
8218.                             title: data.msg,
8219.                             icon: "error"
8220.                         });
```

```
8221.             }
8222.             }).catch(() => {
8223.                 alert('请求参数有误')
8224.             })
8225.         } else if (res.cancel) {
8226.             console.log('用户点击取消');
8227.         }
8228.     }
8229. });
8230. }
8231. </script>
8232. <style lang="scss">
8233. .example {
8234.     padding: 15px;
8235.     background-color: #fff;
8236. }
8237. .segmented-control {
8238.     margin-bottom: 15px;
8239. }
8240. .button-group {
8241.     margin-top: 15px;
8242.     display: flex;
8243.     justify-content: space-around;
8244. }
8245. .form-item {
8246.     display: flex;
8247.     align-items: center;
8248. }
8249. .button {
8250.     display: flex;
8251.     align-items: center;
8252.     height: 35px;
8253.     margin-left: 10px;
8254. }
8255. </style>
8256. <template>
8257.     <view class="container">
8258.         <CameraDisplay :sysRodId="that.sysRodId"></CameraDisplay>
8259.         <CameraNav :sysRodId="that.sysRodId"/>
8260.         <PresetViewer :sysRodId="that.sysRodId" @changePresetId="onChangePresetId"/>
8261.         <uni-section title="配置" type="line">
8262.             <view class="example">
8263.                 <uni-forms :model="form" label-width="90px">
8264.                     <uni-forms-item label="文件名">
```

```

8265.         <uni-easyinput v-model="form.filename" type="text" placeholder="默认为当前时间"/>
8266.     </uni-forms-item>
8267.     <uni-forms-item label="总时长: ">
8268.         <uni-easyinput v-model="form.minutes" type="number">
8269.             <template #suffixIcon>分钟</template>
8270.         </uni-easyinput>
8271.     </uni-forms-item>
8272.     <!--                 <uni-forms-item label="片段时长: "-->
8273.     <!--                 <uni-easyinput                 v-model="form.sliceDuration"
type="number">-->
8274.     <!--                 <template #suffixIcon>分</template>-->
8275.     <!--                 </uni-easyinput>-->
8276.     <!--                 </uni-forms-item>-->
8277.     <!--         <uni-forms-item label="重试次数: ">
8278.         <slider :value="form.retryNum" :step="1" show-value max="5" />
8279.     </uni-forms-item> -->
8280.     <uni-forms-item label="码流类型: ">
8281.         <uni-data-select v-model="form.streamType" :localdata="that.streamTypeOption"
8282.             placeholder="请选择"/>
8283.     </uni-forms-item>
8284.     <uni-forms-item label="声音: ">
8285.         <uni-data-select v-model="form.voiceSwitch" :localdata="that.voiceSwitchOption"
8286.             placeholder="请选择"/>
8287.     </uni-forms-item>
8288.     <button type="primary" @click="onSubmit()">提交</button>
8289. </uni-forms>
8290. </view>
8291. </uni-section>
8292. </view>
8293. </template>
8294. <script setup>
8295. import {
8296.     reactive,
8297. } from 'vue'
8298. import {
8299.     onLoad,
8300. } from '@dcloudio/uni-app';
8301. import PresetViewer from "../online_preview/preset/PresetViewer.vue";
8302. import UniForms from "../uni_modules/uni-forms/components/uni-forms/uni-forms.vue";
8303. import CameraDisplay from "../online_preview/camera/CameraDisplay.vue";
8304. import CameraNav from "../online_preview/camera/CameraNav.vue";
8305. import UniFormsItem from "../uni_modules/uni-forms/components/uni-forms-item/uni-forms-item.vue";
8306. import UniSection from "../uni_modules/uni-section/components/uni-section/uni-section.vue";
8307. import {axios} from "../static/js/api";

```



```
8308. const emit = defineEmits(["closeView"])
8309. const that = reactive({
8310.     presetId: Number,
8311.     streamTypeOption: [{
8312.         value: 1,
8313.         text: "高清"
8314.     },
8315.     {
8316.         value: 2,
8317.         text: "标清"
8318.     },
8319. ],
8320.     voiceSwitchOption: [{
8321.         value: 0,
8322.         text: "关"
8323.     },
8324.     {
8325.         value: 1,
8326.         text: "开"
8327.     },
8328.     {
8329.         value: 2,
8330.         text: "自动"
8331.     },
8332. ],
8333. })
8334. const form = reactive({
8335.     filename: null,
8336.     minutes: 1,
8337.     sliceDuration: 30,
8338.     retryNum: 0,
8339.     streamType: 1,
8340.     voiceSwitch: 2,
8341.     sysRodId: 1
8342. })
8343. onLoad((options) => {
8344.     that.sysRodId = options.sysRodId
8345. })
8346. function onChangePresetId(payload) {
8347.     that.presetId = payload
8348. }
8349. function onSubmit() {
8350.     console.log('submit!')
8351.     uni.showModal({
```

```
8352.         title: '警告',
8353.         content: '请再次确认是否提交! ',
8354.         success: function (res) {
8355.             if (res.confirm) {
8356.                 console.log('用户点击确定');
8357.                 form.sysRodId = that.sysRodId
8358.                 console.log(form)
8359.                 uni.showLoading({
8360.                     title: "正在操作"
8361.                 })
8362.                 axios({
8363.                     method: 'POST',
8364.                     url: '/cloud/video/instant/record',
8365.                     params: form
8366.                 }).then(((data)) => {
8367.                     uni.hideLoading()
8368.                     if (data.code === 200) {
8369.                         uni.showToast({
8370.                             title: "提交成功"
8371.                         });
8372.                         uni.navigateBack({})
8373.                         emit('closeView')
8374.                     } else {
8375.                         uni.showToast({
8376.                             title: data.msg
8377.                         })
8378.                     }
8379.                 })
8380.             } else if (res.cancel) {
8381.                 console.log('用户点击取消');
8382.             }
8383.         }
8384.     }
8385. )
8386. ;
8387. }
8388. </script>
8389. <style lang="scss">
8390. .example {
8391.     padding: 15px;
8392.     background-color: #fff;
8393. }
8394. .segmented-control {
8395.     margin-bottom: 15px;
```

```
8396. }
8397. .button-group {
8398.   margin-top: 15px;
8399.   display: flex;
8400.   justify-content: space-around;
8401. }
8402. .form-item {
8403.   display: flex;
8404.   align-items: center;
8405. }
8406. .button {
8407.   display: flex;
8408.   align-items: center;
8409.   height: 35px;
8410.   margin-left: 10px;
8411. }
8412. </style>
8413. <template>
8414.   <view class="example-body">
8415.     <button v-show="that.tableType==='selection'" class="uni-button top-button" size="mini" type="default"
8416.       @click="delTable()">批量删除
8417.     </button>
8418.     <button class="mini-btn top-button" type="default" size="mini" @click="openDrawer">筛选条件</button>
8419.     <button class="mini-btn top-button" type="default" size="mini" @click="batchDownload">批量下载</button>
8420.   </view>
8421.   <uni-drawer ref="refDrawer" :width="320" mode="right" :mask-click="true">
8422.     <uni-section title="筛选条件" type="line">
8423.       <template v-slot:right>
8424.         <uni-tag type="info" size="30" text="重置条件" @click="resetFilter"/>
8425.       </template>
8426.       <view class="example">
8427.         <uni-forms label-position="top">
8428.           <uni-forms-item label="日期">
8429.             <uni-datetime-picker :border="false" v-model="that.daterange" type="daterange"
8430.               rangeSeparator="-"
8431.               @change="changeDaterange"/>
8432.           </uni-forms-item>
8433.           <uni-forms-item label="文件名">
8434.             <uni-easyinput v-model="that.filename" type="text" placeholder="输入筛选文件名"/>
8435.           </uni-forms-item>
8436.           <button type="primary" @click="queryAndClose">筛选</button>
8437.         </uni-forms>
8438.       </view>
8439.     </uni-section>
```

```

8440.     </uni-drawer>
8441. <!-- <view class="uni-group">
8442.     <label></label>
8443.     <input class="uni-search" type="text" v-model="searchVal" @confirm="search" :placeholder="ss" />
8444.     <button class="uni-button" type="default" size="mini" @click="search">search</button>
8445. </view> -->
8446.     <view class="uni-container">
8447.         <uni-table ref="tableRef" :loading="that.tableLoading" :type="that.tableType" stripe
8448.             @selection-change="selectionChange" style="width: 100%">
8449.             <uni-tr>
8450.                 <!--             <uni-th prop="index" align="center">序号</uni-th>-->
8451.                 <uni-th prop="index"
8452.                     width="200"
8453.                     align="center"
8454.                     filter-type="search"
8455.                     @filter-change="filterChange">
8456.                     文件名
8457.                 </uni-th>
8458.                 <uni-th prop="createTime"
8459.                     width="150"
8460.                     align="center"
8461.                     filter-type="date"
8462.                     @filter-change="filterChange">
8463.                     创建时间
8464.                 </uni-th>
8465.                 <uni-th width="30" v-if="props.fileTypeEnum==='VIDEO'" prop="humidity" align="center">时长
</uni-th>
8466.                 <uni-th align="center" width="30">操作</uni-th>
8467.                 <!--             <uni-th align="center"></uni-th>-->
8468.             </uni-tr>
8469.             <uni-tr v-for="(item, index) in that.tableData" :key="index">
8470.                 <!--             <uni-td align="center">{{ index + 1 }}</uni-td>-->
8471.                 <uni-td align="center">{{ item.filename || "[未命名]" }}</uni-td>
8472.                 <uni-td align="center">{{ item.createTime }}</uni-td>
8473.                 <uni-td v-if="props.fileTypeEnum==='VIDEO'" align="center">
8474.                     {{ item.duration == null ? 0 : item.duration }}s
8475.                 </uni-td>
8476.                 <uni-td align="center">
8477.                     <uni-icons v-if="item.fileTypeEnum === 'VIDEO'" type="paperclip" class="uni-icons"
8478.                         size="20"
8479.                         @click="handleSensor(item)"
8480.                         color="#529b2e">
8481.                     </uni-icons>
8482.                 <uni-icons type="eye-filled" class="uni-icons"

```

```

8483.                size="20" :disabled="item.state!=='DONE'"
8484.                @click="handlePreview(item)"
8485.                color="#529b2e">
8486.            </uni-icons>
8487.            <uni-icons type="cloud-download-filled" class="uni-icons" size="20"
8488.                :disabled="item.state!=='DONE'"
8489.                @click="handleDownload(item)"
8490.                color="#79bbff">
8491.            </uni-icons>
8492.            <uni-icons type="trash" class="uni-icons" size="20"
8493.                :disabled="item.state!=='DONE'"
8494.                color="#f89898"
8495.                @click="handleDelete(item, index)">
8496.            </uni-icons>
8497.        </uni-td>
8498.    </uni-tr>
8499. </uni-table>
8500.    <!--      <uni-pagination :current="that.pageNumber" :pageSize="that.pageSize" :total="that.total"
@change="changePage"/>-->
8501.        <uni-load-more      :status="that.tableDataStatus"      :content-text="that.contentText"
@click-load-more="clickMore"/>
8502.    </view>
8503.    <uni-popup ref="inputDialog" type="dialog">
8504.        <uni-popup-dialog ref="inputClose" mode="input" :title="previewSource.title">
8505.            <video v-if="previewSource.title==='视频'" controls :autoplay="false" poster="/static/uni.png"
8506.                :src="previewSource.src">
8507.            </video>
8508.            <image v-else :src="previewSource.src"/>
8509.        </uni-popup-dialog>
8510.    </uni-popup>
8511. </template>
8512. <script setup>
8513. import {onMounted, reactive, ref, watch,} from 'vue'
8514. import {delete_fileSysFileId, get_downloadBatch, get_file, get_fileSysFileIdPreview} from "@static/js/sysFileApi";
8515. import {uni_downloadFile} from "@static/js/uniApi";
8516. import {timestampToTime} from "@static/js/timeUtil";
8517. const tableRef = ref()
8518. const refDrawer = ref()
8519. const props = defineProps({
8520.     sysRodId: Number,
8521.     fileTypeEnum: String
8522. })
8523. const inputDialog = ref()
8524. const that = reactive({

```

```
8525.     tableDataStatus: 'more',
8526.     pageNumber: 1,
8527.     pageSize: 10,
8528.     total: 0,
8529.     tableLoading: false,
8530.     tableData: [],
8531.     daterange: [],
8532.     tableType: 'selection',
8533.     selectedIndexes: [],
8534.     filename: "",
8535.     contentText: {
8536.         contentdown: "上拉显示更多",
8537.         contentrefresh: "正在加载...",
8538.         contentnomore: "没有更多数据了"
8539.     }
8540. })
8541. const previewSource = reactive({
8542.     src: "",
8543.     cover: null,
8544.     title: '视频',
8545. })
8546. // 设备杆变化, 重新加载
8547. watch(
8548.     () => props.sysRodId,
8549.     (newValue) => {
8550.         console.log("initSensorData()设备杆变化, 重新加载,props.sysRodId:", newValue)
8551.         pagingQueryFile()
8552.     }, {
8553.         deep: true
8554.     }
8555. )
8556. function clickMore() {
8557.     if (that.pageNumber * (that.pageSize + 1) > that.total) {
8558.         that.tableDataStatus = 'noMore'
8559.     } else {
8560.         that.tableDataStatus = 'loading'
8561.         that.pageSize += 5
8562.         that.pageNumber += 1
8563.         const originalTableData = that.tableData
8564.         pagingQueryFile().then(tableData => {
8565.             that.tableData = originalTableData.concat(tableData)
8566.             that.tableDataStatus = 'more'
8567.         })
8568.     }
```

```
8569. }
8570. function batchDownload() {
8571.     console.log(that.selectedIndexs)
8572.     uni.showModal({
8573.         title: '系统提示',
8574.         content: '确认批量下载勾选的文件吗? ! ',
8575.         success: function (res) {
8576.             if (res.confirm) {
8577.                 console.log('用户点击确定');
8578.                 let ids = selectedItems().map(i => {
8579.                     return i.id
8580.                 });
8581.                 console.log('-----', ids)
8582.                 get_downloadBatch(ids)
8583.             } else if (res.cancel) {
8584.                 console.log('用户点击取消');
8585.             }
8586.         }
8587.     });
8588. }
8589. function filterChange(e) {
8590.     console.log(e)
8591.     that.pageNumber = 1
8592.     switch (e.filterType) {
8593.         case 'search':
8594.             that.filename = e.filter
8595.             break
8596.         case 'date':
8597.             that.daterange = e.filter.map(i => timestampToTime(i))
8598.             console.log(that.daterange)
8599.             break
8600.     }
8601.     pagingQueryFile()
8602. }
8603. // 提交筛选条件
8604. function queryAndClose() {
8605.     pagingQueryFile()
8606.     refDrawer.value.close()
8607. }
8608. // 重置筛选条件
8609. function resetFilter() {
8610.     uni.showModal({
8611.         title: '是否重置筛选条件',
8612.         success: function (res) {
```

```
8613.         that.filename = "
8614.         that.daterange = []
8615.         pagingQueryFile()
8616.         refDrawer.value.close()
8617.     }
8618. })
8619. }
8620. // 多选处理
8621. function selectedItems() {
8622.     return that.selectedIndexes.map(i => that.tableData[i])
8623. }
8624. function openDrawer() {
8625.     refDrawer.value.open()
8626. }
8627. // 多选
8628. function selectionChange(e) {
8629.     that.selectedIndexes = e.detail.index
8630. }
8631. //批量删除
8632. function delTable() {
8633.     uni.showModal({
8634.         title: '系统提示',
8635.         content: '确认批量删除勾选的文件吗? ! ',
8636.         success: function (res) {
8637.             if (res.confirm) {
8638.                 console.log('用户点击确定');
8639.                 selectedItems().forEach((item, index) => deleteFile(item, index))
8640.                 // that.selectedIndexes = []
8641.             } else if (res.cancel) {
8642.                 console.log('用户点击取消');
8643.             }
8644.         }
8645.     });
8646. }
8647. // 搜索
8648. function changeDaterange(e) {
8649.     console.log(e)
8650.     that.daterange = e.length > 0 ? [e[0] + " 00:00:00", e[1] + " 23:59:59"] : []
8651.     console.log('daterangee', that.daterange)
8652. }
8653. // 分页
8654. function changePage(e) {
8655.     console.log(e.type, e.current)
8656.     that.pageNumber = e.current
```



```
8657. }
8658. // 获取数据
8659. function pagingQueryFile() {
8660.     if (props.sysRodId === 0) {
8661.         return
8662.     }
8663.     that.tableLoading = true
8664.     let dateRange = that.daterange.length > 0 ? that.daterange : []
8665.     return new Promise((resolve, reject) => {
8666.         get_file(props.sysRodId, props.fileTypeEnum, dateRange[0], dateRange[1], that.filename, that.pageNumber -
1, that.pageSize).then(data => {
8667.             console.log(data.data)
8668.             that.total = data.total
8669.             that.tableData = data.data
8670.             that.tableLoading = false
8671.             // 清空选中状态
8672.             tableRef.value.clearSelection()
8673.             resolve(that.tableData)
8674.         }).catch(err => reject(err))
8675.     })
8676. }
8677. // 预览
8678. function handlePreview(row) {
8679.     console.log(row)
8680.     get_fileSysFileIdPreview(row.id).then(data => {
8681.         if (props.fileTypeEnum === "VIDEO") {
8682.             previewSource.title = '视频'
8683.         } else {
8684.             previewSource.title = '图片'
8685.         }
8686.         previewSource.src = data
8687.         inputDialog.value.open()
8688.     })
8689. }
8690. // 下载视频期间的传感器数据
8691. function handleSensor(item){
8692.     console.log(item)
8693.     uni.showModal({
8694.         title: '系统提示',
8695.         content: '确认下载' + item.filename + '的传感器数据吗?',
8696.         success: function (res) {
8697.             if (res.confirm) {
8698.                 console.log(item)
8699.                 // uni_downloadFile('/file/video/sensor/download/' + item.sysRodTaskId, item, '.xlsx')
```

```

8700.         uni_downloadFile('/file/download/sensor/' + item.id, item.filename+"传感器数据", '.zip')
8701.         .catch((res) => {
8702.             uni.showToast({
8703.                 title: "任务时间过短或数据不存在",
8704.                 icon: "error",
8705.             })
8706.         })
8707.     } else if (res.cancel) {
8708.         console.log('用户点击取消');
8709.     }
8710. }
8711. });
8712. }
8713. //下载
8714. function handleDownload(row) {
8715.     console.log(row)
8716.     let filename = row.filename
8717.     if (row.filename === null || row.filename === "") {
8718.         let date = new Date();
8719.         filename =
            `${date.getFullYear()}-${date.getMonth()+1}-${date.getDay()}-${date.getHours()}:${date.getMinutes()}:${date.getUTCSeconds()}`
            +
8720.     }
8721.     uni.showModal({
8722.         title: '系统提示',
8723.         content: '确认下载文件【' + filename + '】吗？',
8724.         success: function (res) {
8725.             if (res.confirm) {
8726.                 console.log('用户点击确定');
8727.                 uni_downloadFile('/file/download/file/' + row.id, filename || row.pathname, ".zip")
8728.                 // if (props.fileTypeEnum !== "SENSOR")
8729.                 //     uni_downloadFile('/file/download/attach/' + row.id, filename || row.pathname, ".txt")
8730.             } else if (res.cancel) {
8731.                 console.log('用户点击取消');
8732.             }
8733.         }
8734.     });
8735. }
8736. //删除
8737. function handleDelete(row, index) {
8738.     console.log(row)
8739.     uni.showModal({
8740.         title: '系统提示',
8741.         content: '确认删除文件【' + row.filename + '】吗？！',

```

```
8742.         success: function (res) {
8743.             if (res.confirm) {
8744.                 console.log('用户点击确定');
8745.                 deleteFile(row, index)
8746.             } else if (res.cancel) {
8747.                 console.log('用户点击取消');
8748.             }
8749.         }
8750.     });
8751. }
8752. function deleteFile(row, index) {
8753.     delete_fileSysFileId(row.id).then(data => {
8754.         // that.tableData = that.tableData.filter((item,itemIndex) => index !== itemIndex)
8755.         that.tableData = that.tableData.filter((item, itemIndex) => item.id !== row.id)
8756.     })
8757. }
8758. onMounted(() => {
8759.     console.log('mounted...');
8760.     // 加载表格数据
8761.     pagingQueryFile()
8762. })
8763. //暴露 pagingQueryFile 方法
8764. defineExpose({
8765.     pagingQueryFile,
8766.     clickMore
8767. })
8768. </script>
8769. <style>
8770. .example {
8771.     padding: 15px;
8772.     background-color: #fff;
8773. }
8774. .uni-group {
8775.     display: flex;
8776.     align-items: center;
8777.     justify-content: center;
8778.     flex-wrap: wrap;
8779.     word-break: keep-all;
8780. }
8781. /* 搜索框样式 */
8782. .uni-search {
8783.     width: 268px;
8784.     height: 28px;
8785.     line-height: 28px;
```

```
8786.    font-size: 12px;
8787.    color: #606266;
8788.    padding: 0 10px;
8789.    border: 1px #dcdfe6 solid;
8790.    /* margin-right: 10px; */
8791.    border-radius: 3px;
8792. }
8793. /* 容器 */
8794. .uni-container {
8795.    box-sizing: border-box;
8796. }
8797. /* 上边的按钮 */
8798. .top-button {
8799.    margin: 10px;
8800. }
8801. .example-body {
8802.    background-color: #fff;
8803.    padding: 10px 0;
8804. }
8805. .box {
8806.    @include flex;
8807. }
8808. .uni-icons {
8809.    margin-left: 10px;
8810. }
8811. </style>
8812. <template>
8813.    <view class="charts-box">
8814.        <qj-un-data-charts type="line" :opts="that.opts" :chartData="that.chartData" :ontouch="true"/>
8815.        <view class="example-body">
8816.            <button type="default" size="mini" @click="searchLastData('online')">
8817.                实时数据
8818.            </button>
8819.            <button type="default" size="mini" @click="searchLastData('yesterday')">
8820.                昨天数据
8821.            </button>
8822.            <button type="default" size="mini" @click="searchLastData('lastWeek')">
8823.                最近一周
8824.            </button>
8825.            <!-- <button type="default" size="mini" @click="searchLastData('lastMonth')">-->
8826.                最近一个月-->
8827.            <!-- </button>-->
8828.        </view>
8829.        <uni-table ref="table" border stripe emptyText="暂无更多数据">
```

```

8830.         <uni-tr>
8831.             <uni-th prop="index">序号</uni-th>
8832.             <uni-th prop="conductivity">电导率</uni-th>
8833.             <uni-th prop="createTime">时间</uni-th>
8834.         </uni-tr>
8835.         <uni-tr v-for="(item, index) in that.tableData" :key="index">
8836.             <uni-td>{{ index + 1 }}</uni-td>
8837.             <uni-td>{{ item.conductivity }}</uni-td>
8838.             <uni-td>{{ item.createTime }}</uni-td>
8839.         </uni-tr>
8840.     </uni-table>
8841. </view>
8842. </template>
8843. <script setup>
8844. import {onLoad,} from '@dcloudio/uni-app';
8845. import {onMounted, reactive, ref} from "vue";
8846. import {get_nodeDataList} from "@static/js/sysSensorDataApi";
8847. import {timestampToTime} from "@static/js/timeUtil";
8848. const params = ref()
8849. const that = reactive({
8850.     tableData: {},
8851.     lightData: [],
8852.     chartData: {},
8853.     //您可以通过修改 config-ucharts.js 文件中下标为 ['line'] 的节点来配置全局默认参数，如都是默认参数，此处可以不传 opts 。实际应用过程中 opts 只需传入与全局默认参数中不一致的【某一个属性】即可实现同类型的图表显示不同的样式，达到页面简洁的需求。
8854.     opts: {
8855.         color: ["#FAC858", "#91CB74"],
8856.         padding: [15, 10, 0, 15],
8857.         // enableScroll: true,
8858.         dataLabel: false,
8859.         dataPointShape: false,
8860.         legend: {},
8861.         xAxis: {
8862.             disableGrid: true,
8863.             // scrollShow: true,
8864.             // itemCount: 2
8865.         },
8866.         yAxis: {
8867.             gridType: "dash",
8868.             dashLength: 2,
8869.             data: [{
8870.                 min: 20000,
8871.                 max: 10000

```

```
8872.         }}
8873.     },
8874.     extra: {
8875.         line: {
8876.             type: "curve",
8877.             width: 2,
8878.             activeType: "hollow",
8879.             // linearType: "custom"
8880.         }
8881.     }
8882. }
8883. })
8884. function searchLastData(type) {
8885.     let startTime = null
8886.     let endTime = null
8887.     if (type === 'online') {
8888.         that.params.pageSize = 200
8889.         startTime = new Date().getTime() - 1000 * 60 * 60 * 24
8890.         endTime = new Date().getTime()
8891.     } else if (type === 'yesterday') {
8892.         that.params.pageSize = 1000000
8893.         startTime = new Date().getTime() - 1000 * 60 * 60 * 24 * 2
8894.         endTime = new Date().getTime() - 1000 * 60 * 60 * 24
8895.     } else if (type === 'lastWeek') {
8896.         uni.showModal({
8897.             title: '提示',
8898.             content: '最近一周数据量较大，加载时间较长，是否继续？',
8899.             success: function (res) {
8900.                 if (res.confirm) {
8901.                     that.params.pageSize = 1000000
8902.                     startTime = new Date().getTime() - 1000 * 60 * 60 * 24 * 7
8903.                     endTime = new Date().getTime()
8904.                 } else if (res.cancel) {
8905.                     console.log('用户点击取消')
8906.                     searchLastData('online')
8907.                 }
8908.             }
8909.         })
8910.     } else if (type === 'lastMonth') {
8911.         that.params.pageSize = 1000000
8912.         startTime = new Date().getTime() - 1000 * 60 * 60 * 24 * 30
8913.         endTime = new Date().getTime()
8914.     }
8915.     that.params.startTime = timestampToTime(startTime)
```

```
8916.     that.params.endTime = timestampToTime(endTime)
8917.     mountedLastData()
8918. }
8919. /**
8920.  * 数据查询
8921.  */
8922. function mountedLastData() {
8923.     uni.showLoading({
8924.         title: '加载中'
8925.     });
8926.     get_nodeDatalist(that.params.sysRodId, that.params.sensorEnum, that.params.startTime, that.params.endTime,
that.params.pageNumber, that.params.pageSize).then(data => {
8927.         that.lightData = data.records.reverse().map(item => {
8928.             return [item.createTime, item.conductivity]
8929.         })
8930.         // 在通过 mounted 调用即可
8931.         getServerData();
8932.         //表格数据
8933.         that.tableData = data.records.reverse().map(item => {
8934.             item.conductivity += ' S/m'
8935.             return item
8936.         })
8937.     }).finally(() => {
8938.         uni.hideLoading();
8939.     })
8940. }
8941. function getServerData() {
8942.     // console.log(that.lightData.map(item => item[0]))
8943.     let dataList = that.lightData.map(item => item[1])
8944.     let maxVal = Math.max(...dataList)
8945.     let minVal = Math.min(...dataList)
8946.     let average = (maxVal + minVal) / 100
8947.     that.opts.yAxis.data[0].max = maxVal + average
8948.     that.opts.yAxis.data[0].min = minVal - average
8949.     let arr = that.lightData.map(item => item[0])
8950.     let arrLength = arr.length
8951.     if (arrLength > 1) {
8952.         let arrFirst = arr[0].replace('T', '')
8953.         let arrLast = arr[arrLength - 1].replace('T', '')
8954.         arr = arr.map(item => '')
8955.         arr[Math.floor(arrLength / 2)] = arrFirst + "      " + arrLast
8956.     }
8957.     // console.log(arr)
8958.     //模拟服务器返回数据，如果数据格式和标准格式不同，需自行按下面的格式拼接
```

```
8959.     let res = {
8960.         categories: arr,
8961.         series: [{
8962.             name: that.params.deviceName,
8963.             data: dataList
8964.         }]
8965.     };
8966.     that.chartData = JSON.parse(JSON.stringify(res));
8967. }
8968. onLoad((option) => {
8969.     that.params = option
8970. })
8971. onMounted(() => {
8972.     console.log('onMounted。。')
8973.     mountedLastData()
8974. })
8975. </script>
8976. <style scoped>
8977. /* 请根据实际需求修改父元素尺寸，组件自动识别宽高 */
8978. .charts-box {
8979.     width: 100%;
8980.     height: 300px;
8981. }
8982. </style>
8983. <template>
8984.     <view class="charts-box">
8985.         <qj-un-data-charts type="line" :opts="that.opts" :chartData="that.chartData" :ontouch="true"/>
8986.         <view class="example-body">
8987.             <button type="default" size="mini" @click="searchLastData('online')">
8988.                 实时数据
8989.             </button>
8990.             <button type="default" size="mini" @click="searchLastData('yesterday')">
8991.                 昨天数据
8992.             </button>
8993.             <button type="default" size="mini" @click="searchLastData('lastWeek')">
8994.                 最近一周
8995.             </button>
8996.             <!-- <button type="default" size="mini" @click="searchLastData('lastMonth')">-->
8997.             <!-- 最近一个月-->
8998.             <!-- </button>-->
8999.         </view>
9000.         <uni-table ref="table" border stripe emptyText="暂无更多数据">
9001.             <uni-tr>
9002.                 <uni-th prop="index">序号</uni-th>
```



```

9003.         <uni-th prop="temperature">温度</uni-th>
9004.         <uni-th prop="humidity">湿度</uni-th>
9005.         <uni-th prop="createTime">时间</uni-th>
9006.     </uni-tr>
9007.     <uni-tr v-for="(item, index) in that.tableData" :key="index">
9008.         <uni-td>{{ index + 1 }}</uni-td>
9009.         <uni-td>{{ item.temperature }}</uni-td>
9010.         <uni-td>{{ item.humidity }}</uni-td>
9011.         <uni-td>{{ item.createTime }}</uni-td>
9012.     </uni-tr>
9013. </uni-table>
9014. </view>
9015. </template>
9016. <script setup>
9017. import {onLoad,} from '@dcloudio/uni-app';
9018. import {onMounted, reactive, ref} from "vue";
9019. import {get_nodeDataList} from "@static/js/sysSensorDataApi";
9020. import {timestampToTime} from "@static/js/timeUtil";
9021. const params = ref()
9022. const that = reactive({
9023.     tableData: {},
9024.     temperatureData: [
9025.         ["2000-06-05", 116],
9026.         ["2000-06-06", 129],
9027.         ["2000-06-07", 135]
9028.     ],
9029.     humidityData: [
9030.         ["2000-06-05", 116],
9031.         ["2000-07-16", 64],
9032.         ["2000-07-17", 69]
9033.     ],
9034.     chartData: {},
9035.     //您可以通过修改 config-ucharts.js 文件中下标为 ['line'] 的节点来配置全局默认参数，如都是默认参数，此处可以不传 opts 。实际应用过程中 opts 只需传入与全局默认参数中不一致的【某一个属性】即可实现同类型的图表显示不同的样式，达到页面简洁的需求。
9036.     opts: {
9037.         color: ["#FAC858", "#73C0DE", "#3CA272", "#FC8452"],
9038.         padding: [15, 10, 0, 15],
9039.         dataLabel: false,
9040.         dataPointShape: false,
9041.         legend: {},
9042.         xAxis: {
9043.             disableGrid: true
9044.         },

```

```
9045.     yAxis: {
9046.         gridType: "dash",
9047.         dashLength: 2,
9048.         data: [{
9049.             min: 20000,
9050.             max: 10000
9051.         }]
9052.     },
9053.     extra: {
9054.         line: {
9055.             type: "curve",
9056.             width: 2,
9057.             activeType: "hollow"
9058.         }
9059.     }
9060. }
9061. })
9062. function searchLastData(type) {
9063.     let startTime = null
9064.     let endTime = null
9065.     if (type === 'online') {
9066.         that.params.pageSize = 200
9067.         startTime = new Date().getTime() - 1000 * 60 * 60 * 24
9068.         endTime = new Date().getTime()
9069.     } else if (type === 'yesterday') {
9070.         that.params.pageSize = 1000000
9071.         startTime = new Date().getTime() - 1000 * 60 * 60 * 24 * 2
9072.         endTime = new Date().getTime() - 1000 * 60 * 60 * 24
9073.     } else if (type === 'lastWeek') {
9074.         uni.showModal({
9075.             title: '提示',
9076.             content: '最近一周数据量较大，加载时间较长，是否继续？',
9077.             success: function (res) {
9078.                 if (res.confirm) {
9079.                     that.params.pageSize = 1000000
9080.                     startTime = new Date().getTime() - 1000 * 60 * 60 * 24 * 7
9081.                     endTime = new Date().getTime()
9082.                 } else if (res.cancel) {
9083.                     console.log('用户点击取消')
9084.                     searchLastData('online')
9085.                 }
9086.             }
9087.         })
9088.     } else if (type === 'lastMonth') {
```

```
9089.         that.params.pageSize = 1000000
9090.         startTime = new Date().getTime() - 1000 * 60 * 60 * 24 * 30
9091.         endTime = new Date().getTime()
9092.     }
9093.     that.params.startTime = timestampToTime(startTime)
9094.     that.params.endTime = timestampToTime(endTime)
9095.     mountedLastData()
9096. }
9097. /**
9098.  * 数据查询
9099.  */
9100. function mountedLastData() {
9101.     uni.showLoading({
9102.         title: '加载中'
9103.     });
9104.     get_nodeDataList(that.params.sysRodId, that.params.sensorEnum, that.params.startTime, that.params.endTime,
that.params.pageNumber, that.params.pageSize).then(data => {
9105.         let data2 = data.records.reverse();
9106.         that.temperatureData = data2.map(item => {
9107.             return [item.createTime, item.temperature]
9108.         })
9109.         that.humidityData = data2.map(item => {
9110.             return [item.createTime, item.humidity]
9111.         })
9112.         // 在通过 mounted 调用即可
9113.         getServerData();
9114.         //表格数据
9115.         that.tableData = data.records.reverse().map(item => {
9116.             item.temperature += ' °C'
9117.             item.humidity += '%rh'
9118.             return item
9119.         })
9120.     }).finally(() => {
9121.         uni.hideLoading();
9122.     })
9123. }
9124. function getServerData() {
9125.     const temperatureValueList = that.temperatureData.map(function (item) {
9126.         return item[1];
9127.     });
9128.     const humidityValueList = that.humidityData.map(function (item) {
9129.         return item[1];
9130.     });
9131.     let dataList = temperatureValueList.concat(humidityValueList)
```

```
9132.   let maxValue = Math.max(...dataList)
9133.   let minValue = Math.min(...dataList)
9134.   let average = (maxValue + minValue) / 10
9135.   that.opts.yAxis.data[0].max = maxValue + average
9136.   that.opts.yAxis.data[0].min = 0
9137.   let timeList = that.humidityData.map(item => item[0])
9138.   let timeListLength = timeList.length
9139.   if (timeListLength > 1) {
9140.       let timeListFirst = timeList[0].replace('T', ' ')
9141.       let timeListLast = timeList[timeListLength - 1].replace('T', ' ')
9142.       timeList = timeList.map(item => "")
9143.       timeList[parseInt(timeListLength / 2)] = timeListFirst + "    " + timeListLast
9144.   }
9145.   // console.log(timeList)
9146.   //模拟服务器返回数据，如果数据格式和标准格式不同，需自行按下面的格式拼接
9147.   let res = {
9148.       categories: timeList,
9149.       series: [{
9150.           name: "温度",
9151.           data: temperatureValueList
9152.       }, {
9153.           name: "湿度",
9154.           data: humidityValueList
9155.       }]
9156.   };
9157.   that.chartData = JSON.parse(JSON.stringify(res));
9158. }
9159. onLoad((option) => {
9160.   that.params = option
9161. })
9162. onMounted(() => {
9163.   console.log('onMounted。。')
9164.   mountedLastData()
9165. })
9166. </script>
9167. <style scoped>
9168. /* 请根据实际需求修改父元素尺寸，组件自动识别宽高 */
9169. .charts-box {
9170.   width: 100%;
9171.   height: 300px;
9172. }
9173. </style>
```