

AJAX



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌

14

案例-图书管理

需求

1. 图书列表
2. 新增图书
3. 删除图书
4. 编辑图书

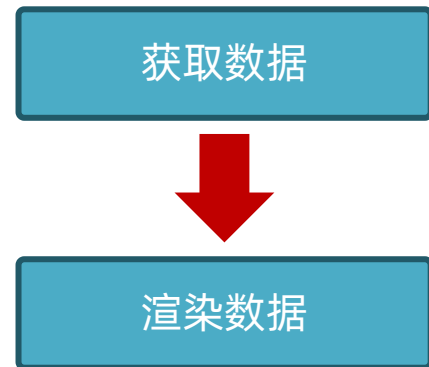
图书管理

[+ 添加](#)

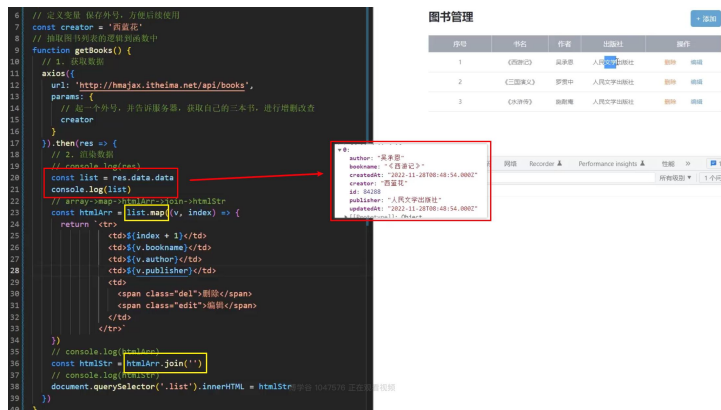
序号	书名	作者	出版社	操作	
1	《西游记》	吴承恩	人民文学出版社	删除	编辑
2	《三国演义》	罗贯中	人民文学出版社	删除	编辑
3	《水浒传》	施耐庵	人民文学出版社	删除	编辑

图书列表

自己的图书数据: 给自己起个**外号**，并告诉服务器，默认会有三本书，基于这三本书做数据的**增删改查**



序号	书名	作者	出版社	操作	
1	《西游记》	吴承恩	人民文学出版社	删除	编辑
2	《三国演义》	罗贯中	人民文学出版社	删除	编辑
3	《水浒传》	施耐庵	人民文学出版社	删除	编辑



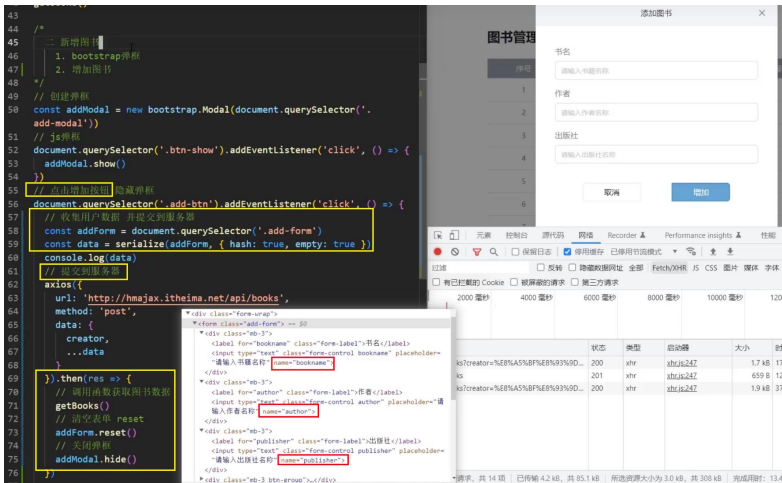
需求

1. 图书列表
2. 新增图书
3. 删除图书
4. 编辑图书

图书管理

+ 添加

序号	书名	作者	出版社	操作	
1	《西游记》	吴承恩	人民文学出版社	删除	编辑
2	《三国演义》	罗贯中	人民文学出版社	删除	编辑
3	《水浒传》	施耐庵	人民文学出版社	删除	编辑



新增图书

1. 通过属性控制

```
<button data-bs-toggle="modal" data-bs-target="css选择器">  
    弹框  
</button>
```

```
<button data-bs-dismiss="modal">隐藏弹框</button>
```

2. 通过js控制

```
// 创建弹框  
const modalDom = document.querySelector('弹框选择器')  
const modal = new bootstrap.Modal(modalDom)  
// 显示弹框  
modal.show()  
// 隐藏弹框  
modal.hide()
```



添加图书 ×

书名

作者

出版社

取消 增加

需求

1. 图书列表
2. 新增图书
3. 删除图书
4. 编辑图书

图书管理

+ 添加

序号	书名	作者	出版社	操作	
1	《西游记》	吴承恩	人民文学出版社	删除	编辑
2	《三国演义》	罗贯中	人民文学出版社	删除	编辑
3	《水浒传》	施耐庵	人民文学出版社	删除	编辑

```
80
81 三 删除图书
82 1. 绑定点击事件
83 2. 删除图书
84
85 // 绑定点击事件 - 事件委托
86 document.querySelector('.list').addEventListener('click', (e) => {
87   // console.log('点了tbody')
88   // 通过事件参数 获取触发事件的元素
89   // console.log(e.target)
90   // 点了删除再执行后续逻辑
91   if (!e.target.classList.contains('del')) {
92     // 点的不是删除 才会执行这里的逻辑
93     return
94   }
95   // 点的是删除 才会执行到这里
96   console.log('点了删除')
97
98   // 2. 删除图书
99   // 获取保存在删除元素父元素上的自定义属性 data-id的值
100  // console.log(e.target.parentNode)
101  // console.log(e.target.parentNode.dataset)
102  const { id } = e.target.parentNode.dataset
103  // console.log(id)
104  axios({
105    url: 'http://hmmajax.itheima.net/api/books/${id}',
106    method: 'DELETE'
107  }).then(res => {
108    // 重新获取数据
109    getBooks()
110  })
111 })
```

```
const htmlArr = list.map((v, index) => {
  return `<tr>
    <td>${index + 1}</td>
    <td>${v.bookname}</td>
    <td>${v.author}</td>
    <td>${v.publisher}</td>
    <td data-id="${v.id}">
      <span class="del">删除</span>
      <span class="edit">编辑</span>
    </td>
  </tr>`
})
```

通过data-拿到id

博学谷 1047576 正在观看视频

元素 控制台 源代码 网络

编辑

<tr></tr>

<td>3</td>

<td>前端与移动开发2</td>

<td>itheima</td>

<td>人民邮电出版社</td>

<td data-id="84325">

删除

编辑

名称	状态	类型	启动器	大小	时间
<input type="checkbox"/> books?creator=%E8%A5%BF%E8%93%9D...	200	xhr	xhr.js:247	1.7 kB	62 ...
<input type="checkbox"/> 84325	204	xhr	xhr.js:247	348 B	31 ...
<input type="checkbox"/> books?creator=%E8%A5%BF%E8%93%9D...	200	xhr	xhr.js:247	1.5 kB	71 ...

第 3 项请求，共 13 项 | 已传输 3.5 kB，共 85.4 kB | 所选资源大小为 2.3 kB，共 308 kB | 完成用时: 5.67 秒

删除图书

绑定点击事件



删除图书

操作	
删除	编辑
删除	编辑
删除	编辑

删除-图书

DELETE <http://hmajax.itheima.net/api/books/{id}>

删除-某本图书

请求参数

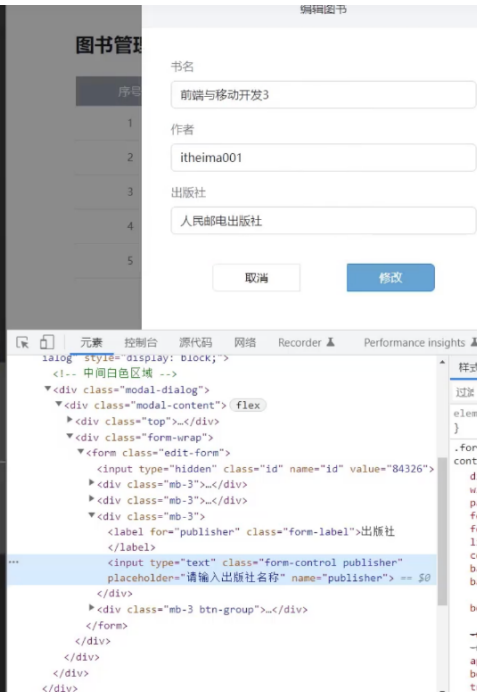
参数名	位置	类型	必填	说明
id	path	number	是	示例值: 1 图书id

```
axios({  
  url: 'http://hmajax.itheima.net/api/books/8910',  
  method: 'delete'  
})
```


需求

1. 图书列表
2. 新增图书
3. 删除图书
4. 编辑图书

```
121 // 绑定点击事件 - 事件委托
122 document.querySelector('.list').addEventListener('click', (e) => {
123   if (!e.target.classList.contains('edit')) {
124     return
125   }
126   // console.log('点了编辑')
127   // 获取图书id 通过编辑元素的父元素的自定义属性 data-id获取id
128   // console.log(e.target.parentNode.dataset)
129   const { id } = e.target.parentNode.dataset
130   // console.log(id)
131   // 获取图书数据
132   axios({
133     url: 'http://hmax.itheima.net/api/books/${id}'
134   }).then(res => {
135     // console.log(res)
136     const book = res.data.data // 先从后台获取数据，然后回填数据
137     console.log(book)
138     // 设置获取到的图书数据
139     // document.querySelector('.edit-modal .id').value = book.id
140     // document.querySelector('.edit-modal .bookname').value = book.bookname
141
142     // Object.keys 获取对象的属性名 ['id', 'bookname', 'author', 'publisher']
143     // const keys = Object.keys(book)
144     // console.log(keys)
145     Object.keys(book).forEach(key => {
146       // console.log(key)
147       document.querySelector('.edit-modal .${key}').value = book[key]
148     })
149
150     // 弹出编辑框
151     editModal.show()
152   })
153 })
154 })
```

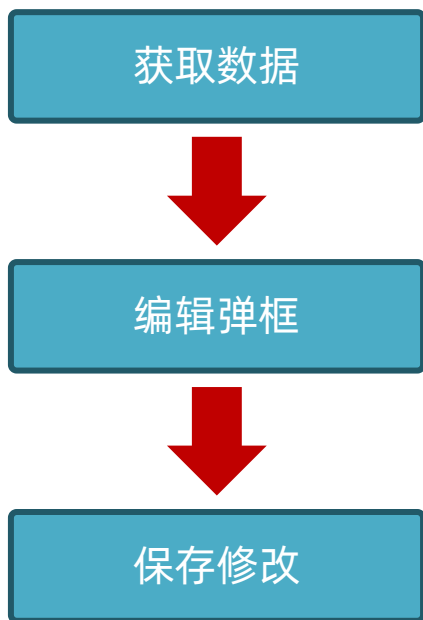


图书管理

序号	书名	作者	出版社	操作
1	《西游记》	吴承恩	人民文学出版社	删除 编辑
2	《三国演义》	罗贯中	人民文学出版社	删除 编辑
3	《水浒传》	施耐庵	人民文学出版社	删除 编辑

```
119 // 通过js控制编辑弹框
120 const editModal = new bootstrap.Modal(document.querySelector('.edit-modal'))
121 > // editModal.show() ...
122 > document.querySelector('.list').addEventListener('click', (e) => { ...
123 })
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

编辑图书



1	《西游记》	吴承恩	人民文学出版社	删除	编辑
2	《三国演义》	罗贯中	人民文学出版社	删除	编辑
3	《水浒传》	施耐庵	人民文学出版社	删除	编辑

编辑图书

书名

《三国演义》

作者

罗贯中

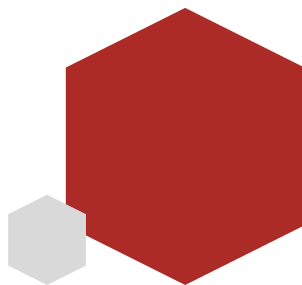
出版社

人民文学出版社

取消

修改





图片上传

```
<script>
/**
 * 图片上传
 * 1. 选择图片
 * 2. FormData对象
 * 3. 调用上传接口
 */
document.querySelector('.upload').addEventListener('change', function () {
  // console.log(this)
  // 1. 选择图片
  // console.log(this.files)
  console.log(this.files[0])

  // 2. FormData对象
  const data = new FormData()
  // 参数1 接口文档要求的参数名
  // 参数2 提交给服务器的数据（常用的是文件）
  data.append('img', this.files[0])

  // 3. 调用上传接口
  axios({
    url: 'http://h Ajax.itheima.net/api/uploading',
    method: 'POST',
    data: data
  }).then(res => {
    // console.log(res)
    const url = res.data.data.url
    console.log(url)
    document.querySelector('.img').src = url
  })
})
```

```
<!-- file标签 -->
<input type="file" class="upload">

<!-- 渲染服务器返回的图片URL -->
<img src="" class="img" alt="">
```

图片上传

1. 选择图片

```
<input type="file" class="upload" >
```

```
document.querySelector('.upload').addEventListener('change', function ()  
{  
  console.log(this.files[0])  
})
```

2. FormData对象

内置对象，负责向服务器传递表单数据，常用于文件上传

```
const data = new FormData()  
data.append(key,value)
```

3. 调用上传接口

```
axios({  
  url:'http://hmajax.itheima.net/api/uploading',  
  method:'post',  
  data:formData对象  
})
```

上传-图片

POST http://hmajax.itheima.net/api/uploading

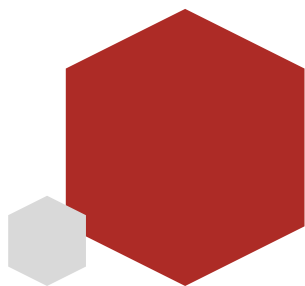
上传-图片

注意: 上传的图片必须在2MB以内

请求参数

Body 参数(multipart/form-data)

参数名	类型	必填	说明
img	file	是	图片文件(2MB以内)

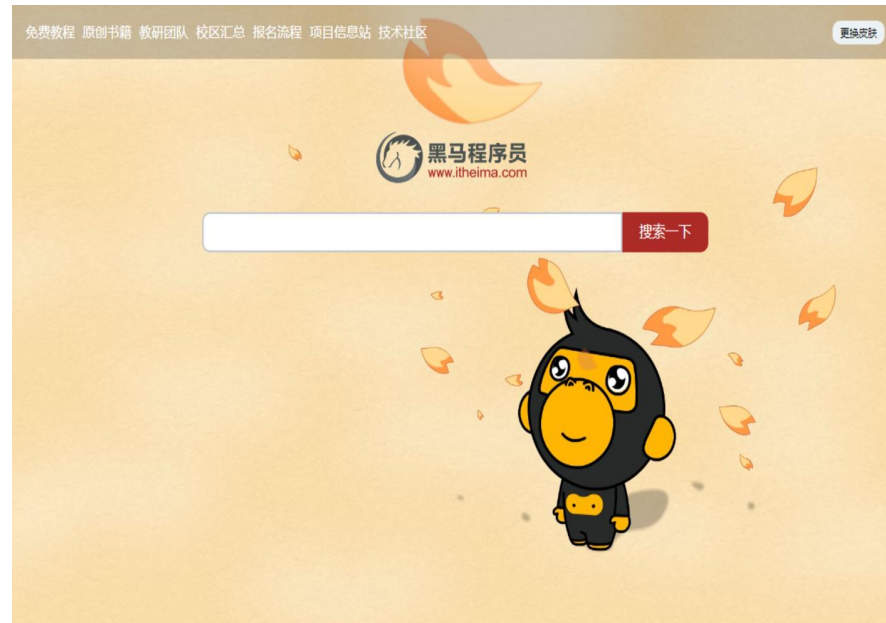


案例-网站换肤

需求

- 网站换肤 (核心功能)
 - 上传图片
 - 设置body的背景图片
- 本地持久化图片URL
 - 保存图片URL
 - 读取图片URL

```
17 axios({
18   url: 'http://hmajax.itheima.net/api/uploading',
19   method: 'POST',
20   data
21 }).then(res => {
22   // console.log(res)
23   const url = res.data.data.url
24   console.log(url)
25
26   // 1.2 设置body的背景图
27   document.body.style.backgroundImage = `url(${url})`
28
29   // 2.1 保存图片URL
30   localStorage.setItem('img', url)
31 })
32
33
34 /**
35  * 二 本地持久化图片URL
36  * 1. 保存图片URL
37  * 2. 读取图片URL
38  * */
39
40 // 2.2 读取图片URL
41 // 这里写代码会在页面打开时直接执行
42 const url = localStorage.getItem('img')
43 console.log(url)
44 // 设置body的背景图 逻辑与&&，左侧为真，执行右侧
45
46 // url存在 设置背景图 加小括号是因为逻辑与和=的
47 // url为null 不需要设置背景图 优先级原因
48 url && ((document.body.style.backgroundImage = `url(${url})`))
49
```





总结

1. label标签使用技巧

```
<label for="upload">更换皮肤</label>  
<input id="upload" class="skin-ipt" type="file">
```

2. 网站换肤

上传图片

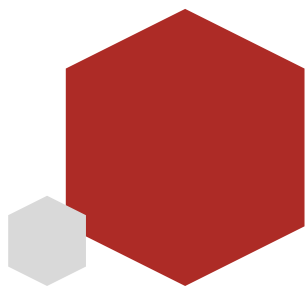
设置body的背景图片

3. 本地持久化图片URL

保存图片URL

读取图片URL

```
localStorage.setItem(key, value)  
localStorage.getItem(key)
```



案例-个人设置

需求

- 信息渲染
- 修改头像
- 修改信息

```
// 定义外号，获取属于自己的数据
const creator = '西瓜花'
axios({
  url: 'http://hmax.itheima.net',
  params: {
    creator
  }
}).then(res => {
  // console.log(res)
  const userInfo = res.data.data

  // 2. 渲染数据
  Object.keys(userInfo).forEach(key => {
    // console.log(key)
    // 2.1 头像 src属性
    if (key === 'avatar') {
      document.querySelector(`.${key}`).src = userInfo[key]
    }

    // 2.2 性别 checked属性
    else if (key === 'gender') {
      // 获取所有性别标签 伪数组[男Radio, 女Radio]
      const genderRadios = document.querySelectorAll(`.${key}`)
      console.log(genderRadios)
      // 获取服务器返回的性别 整数 0 男 1 女
      const gender = userInfo[key]
      // 找到对应标签 选中 性别作为索引即可获取对应的标签
      genderRadios[gender].checked = true
    }


    // 2.3 邮箱 昵称 个人简介 value属性
    else {
      document.querySelector(`.${key}`).value = userInfo[key]
    }
  })
})
```

```
avatar: "http://hmax.itheima.net/avatar/avatar2.png"
desc: "我是西瓜花"
email: "itheima@itcast.cn"
gender: 0
nickname: "itheima"
```

avatar string 用户头像地址
email string 用户邮箱
nickname string 用户昵称
gender integer 用户性别
desc string 用户简介

安全设置 账号绑定 新消息通知

邮箱 itheima@itcast.cn

头像  更换头像

昵称 itheima

性别 ☒ 男 ☐ 女

个人简介 我是西瓜花

提交

元素 控制台 源代码 网络 Recorder Performance insights

<h3 class="title">基本设置</h3>
<form class="user-form" action="javascript:;">
 <div class="form-item"></div>
 <div class="form-item"></div>
 <div class="form-item"></div>
 <div class="form-item">
 <label for="desc">个人简介</label>
 <textarea id="desc" name="desc" class="desc" placeholder="请输入个人简介" cols="20" rows="10" autocomplete="off">
 </div>
 <div class="form-item">
 <div class="avatar-box"></div> <div> </div>
 </div>
</form>
</div>

样式 计算样式

过渡: hover .cls

element.style {

.content .user-form textarea {

padding-top: 10px;

width: 350px;

resize: none;

.form-item

input: not([type=radio])

textarea {

display: block;

width: 350px;

outline: none;

基本设置

安全设置 账号绑定 新消息通知

邮箱 请输入邮箱

头像  更换头像

昵称 请输入昵称

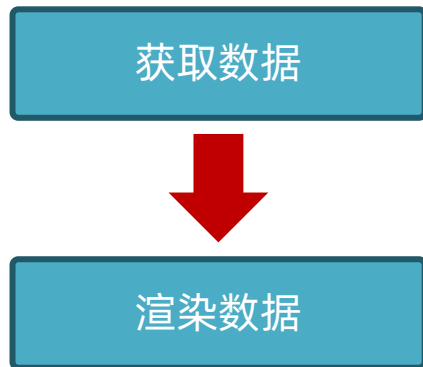
性别 ☐ 男 ☐ 女

个人简介 请输入个人简介

提交

信息渲染

自己的用户信息: 给自己起个外号，并告诉服务器，获取对应的用户信息



基本设置

安全设置

账号绑定

新消息通知

基本设置

邮箱

昵称


性别
☒ 男 ☐ 女

个人简介

我是播仔

提交

头像



更换头像

需求

- 信息渲染
- 修改头像
- 修改信息

```
/**
 * 二 修改头像
 * 1. 获取头像
 * 2. 调用修改头像接口
 * 3. 更新图片URL
 * */
// 1. 获取头像
document.querySelector('#upload').addEventListener('change', function () {
    console.log(this.files)

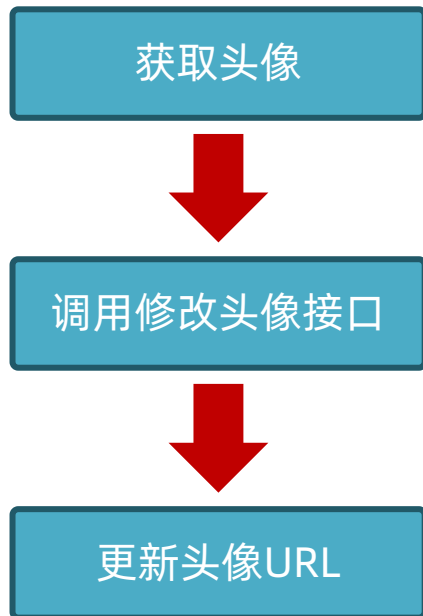
    // 2. 调用修改头像接口
    const data = new FormData()
    // 新头像
    data.append('avatar', this.files[0])
    // 外号，告诉服务器修改谁的数据，和获取数据的外号要一致
    data.append('creator', creator)
    axios({
        url: 'http://hmajax.itheima.net/api/avatar',
        method: 'PUT',
        data
    }).then(res => {
        // console.log(res)
        // 3. 更新图片URL
        const avatar = res.data.data.avatar
        document.querySelector('.avatar').src = avatar
    })
})
```

博学谷 1047576 正在观看视频



The image shows a user profile settings interface. On the left is a sidebar with links: '基本设置' (Basic Settings), '安全设置' (Security Settings), '账号绑定' (Account Binding), and '消息通知' (Message Notification). The main area is titled '基本设置' and contains several input fields: '邮箱' (Email) with a placeholder '请输入邮箱', '昵称' (Nickname) with a placeholder '请输入昵称', and '性别' (Gender) with radio buttons for '男' (Male) and '女' (Female). There is also a '个人简介' (Personal Introduction) text area with a placeholder '请输入个人简介'. At the bottom is a blue '提交' (Submit) button. To the right of the form is a circular '头像' (Avatar) placeholder with a person icon and a '更换头像' (Change Avatar) button below it.

修改头像



修改-个人头像

PUT <http://hmajax.itheima.net/api/avatar>

修改-个人头像

请求参数

Body 参数(multipart/form-data)

参数名	类型	必填
avatar	file	是
creator	string	是

需求

- 信息渲染
- 修改头像
- 修改信息

```
* 三 修改信息
* 1. 收集表单数据
* 2. 调用修改信息接口
* 3. 提示用户
* */

document.querySelector('.submit').addEventListener('click', () => {
  // 1. 收集表单数据
  const form = document.querySelector('.user-form')
  const formData = serialize(form, { hash: true, empty: true })
  // console.log(formData)

  // 2. 调用修改信息接口
  // 性别的类型转为 整数（文档要求）
  formData.gender = +formData.gender

  // 需要和creator合并为一个对象（告诉服务器修改的是谁的数据）
  const data = {
    ...formData,
    creator
  }
  console.log(data)

  axios({
    url: 'http://hmajax.itheima.net/api/settings',
    method: 'PUT',
    data
  })
})
```

```
desc: "我是西蓝花"
email: "itheima@itcast.cn"
gender: "1"
nickname: "itheima001"
▶ [[Prototype]]: Object
```



基本设置

邮箱

昵称

性别 ☐ 男 ☐ 女

个人简介

头像 

修改信息

收集表单数据

↓

调用修改信息接口

↓

提示用户


基本

操作成功

邮箱

itheim111@itcast.cn

头像



更换头像

昵称

itheima111

性别

☐ 男 ☒ 女

个人简介

我是测试用1111

提交

```
<div class="toast" data-bs-delay="1500" >  
  // 内容略  
</div>
```

```
// 创建弹框  
const toastDom = document.querySelector('.toast')  
const toast = new bootstrap.Toast(toastDom)  
// 显示弹框  
toast.show()
```



总结

1. 信息渲染

获取 -> 渲染

2. 修改头像

调用接口 -> 更新头像URL

3. 修改信息

收集数据 -> 调用接口->提示用户

```
<div class="toast" data-bs-delay="1500" >  
  // 内容略  
</div>
```

```
// 创建弹框  
const toastDom = document.querySelector('提示框选择器')  
const toast = new bootstrap.Toast(toastDom)  
// 显示弹框  
toast.show()
```



传智教育旗下高端IT教育品牌