

高级web 问题与心得

1. 文件描述和文件如何同时添加？

在最初添加文件时，是将input和uploader绑定在一起。

```
<input type="file" ng2FileSelect [uploader]="uploader"
```

这样的做法使选择了文件之后文件的FileItem就自动添加到了uploader.queue里，不能同时添加文件描述到存储文件描述的链表里。这里会出现添加了文件却没有及时添加文件描述的隐患。查了api后发现手动添加FileItem到uploader.queue里的函数addToQueue ()。于是之后没有将uploader和input绑定，而是从input里获取文件的FileItem对象后，使用addToQueue函数手动添加到uploader.queue里，并且把使用addToQueue手动添加的操作和将文件描述添加到链表里的操作合并成一个函数。

Input标签，调用selectFile将文件的FileItem对象赋给selectedFiles

```
<li><input type="file" id = "selectfile" (change)="selectFile($event)"></li>
```

```
selectFile(event) {  
  console.log('select file');  
  this.selectedFiles = event.target.files;  
}
```

addFile () 函数，从selectedFiles里获取FileItem然后调用addToQueue函数，并将文件描述添加到文件描述的链表

```
addFile() {  
  const file = [];  
  file.push(this.selectedFiles.item(0));  
  const filedescribe = this.elementRef.nativeElement.querySelector('#fd').value;  
  console.log('filedescribe: ' + filedescribe);  
  this.filedescription.push(filedescribe);  
  console.log('filedescription: ');  
  console.log(this.filedescription);  
  this.uploader.addToQueue(file);  
}
```

文件描述的链表:

```
public filedescription: LinkedList<string> = new LinkedList();
```

2. 如何获取前端修改后的节点和字体颜色值？

一开始，想到是一个笨方法，就是通过一系列li的点击事件，li中给出几种具体的颜色，每个li的value值代表了这种颜色，然后通过点击事件把value传到逻辑处理中。这种方法可行，但是局限了颜色的选择设置，影响用户体验。后来，知道了有一个type为color的input，是网页自带的颜色选择器，把它的值用双向绑定到变量名就好了。

3. 截图实现中遇到的问题

截图代码只有一行，但是尝试了很久。一开始是上网搜的用的是screenshot()，然后又看jsmind/js/jsmind.screenshot.js文件

```
jsMind.screenshot = function(jm){  
    this.jm = jm;  
    this.canvas_elem = null;  
    this.canvas_ctx = null;  
    this._inited = false;  
};  
  
jsMind.screenshot.prototype = {
```

```
shoot:function(callback){
    this.init();
    this._watermark();
    var jms = this;
    this._draw(function(){
        if (!!callback){
            callback(jms);
        }
        jms.clean();
    });
},

shootDownload: function(){
    this.shoot(function(jms){
        jms._download();
    });
},
```

结合网上后续搜到的一些参考，决定使用screenshot.shootDownload()，但还是莫名失败了，后来正好问了一位刚写完这个功能的同学，他说要修改jsmind/js/jsmin.js中的一行代码

```
// export jsmind
if (typeof module !== 'undefined' && typeof ex
    module.exports = jm;
    ...
    $w['__name__'] = jm; // new
} else if (typeof define === 'function' && (de
    define(function() { return jm; }));
} else {
    $w['__name__'] = jm;
}
```

在该文件最后部分，注释new的就是后来加的一行。然后，成功了。

4. 部署的时候打成war包出错

经过网上查阅，了解到IDEA可以打开maven的依赖关系图，然后我看到了依赖版本的冲突，然后改动了版本，解决了这一问题。

5. 部署的时候数据库的地址问题

一开始mongoDB跑在27017端口上，我在服务器中设置，打开了27017端口的入站和出站流量，但是外界还是无法访问，后来将其改为127.0.0.1: 27017即可，因为后端和数据库之间的联系均在服务器上完成，这也正是前后端分离的好处。