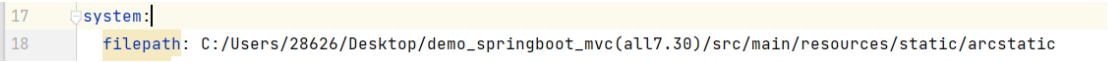
Spring boot 文件上传

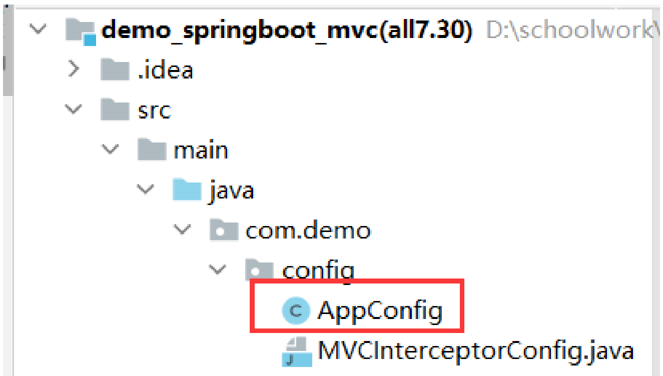
1. Application.yml 中添加 filepath



```
system:
  filepath: C:/Users/28626/Desktop/demo_springboot_mvc(all7.30)/src/main/resources/static/arcstatic
```

2. 创建配置类 AppConfig.java

项目目录：


demo_springboot_mvc(all7.30) D:\schoolwork\
.idea
src
main
java
com.demo
config
AppConfig
MVCInterceptorConfig.java

代码：

```
package com.demo.config;


import org.springframework.boot.context.properties.ConfigurationProperties;

import org.springframework.stereotype.Component;


/**

 * @Author LiuHong

 * @CreateTime 2021/7/26 12:14

 * @Version 1.0.0

 */

@Component

@ConfigurationProperties(prefix = "system") // 接收 application.yml 中的 myProps 下面的属性

public class AppConfig {

    public String filepath;

    public String urlpath;
    public String getFilepath() {
        System.out.println("in appConfig, filepath:"+filepath);
        return filepath;
    }

    public void setFilepath(String filepath) {
        this.filepath = filepath;
    }

    public String getUrlpath() {
        return urlpath;
    }

    public void setUrlpath(String urlpath) {
        this.urlpath = urlpath;
    }
}
```

3. Controller

```
    @RequestMapping("/editor/saveArc")
    @ResponseBody
    public Map<String, String> saveArc(Model model, HttpServletRequest request, HttpServletResponse response,
                                        HttpSession session, @RequestBody TemArc temArc ) throws Exception {
        /************************* debug *******************************/
//          System.out.println("here is saveArc");
//          System.out.println(temArc.toString());
        /************************* debug *******************************/
        //添加文章属性
        //合成语音文件
        try{
            String arcAudioFile="";
            if(temArc.isTemArcIfAudio() == true){
                SendController audioCont=new SendController();

arcAudioFile=audioCont.textDetect(temArc.getTextCont(),appConfig.getFilepath(
)+"/article/audio");
                if(arcAudioFile==""){
                    System.out.println("合成语音错误！ ");
                }
                else{

arcAudioFile=appConfig.getUrlpath()+"/article/audio/"+arcAudioFile;
                }
            }

            //检测文章是否含敏感词，设置文章状态
            String arcStatus="posted";//arcStatus=arcCheck();
            /**************************                              debug
********************************/

            //存储文章内容
            Helper helper=new Helper();
            String arcFile=helper.saveArcContent(temArc.getHtmlCont());

            //生成文章摘要和封面图
            String arcAbstract=temArc.getTemArcAbstract();
            if(arcAbstract=="" || arcAbstract==null) {
                Integer endIndex=49;
            if((temArc.getTextCont()).length()<50){
                endIndex=(temArc.getTextCont()).length();
            }
            arcAbstract = (temArc.getTextCont()).substring(0, endIndex);
        }
        String
arcImgFile=helper.getImgFromHtml(temArc.getHtmlCont());
        TUser user=(TUser) session.getAttribute("user");
        model.addAttribute("user",user);
        temArc.setTemArcAuthorId(user.getUserId());

        //更新数据库
        TArticle                                                  arc=new
TArticle(temArc.getTemArcTypeId(),temArc.getTemArcAuthorId(),arcFile,temAr
c.isTemArcIfAudio()
                ,arcAudioFile,arcStatus,temArc.getTemArcTit(),arcAbstr
act,arcImgFile);
        itArticleMapper.addArc(arc);

        //返回文章发布结果
        Map<String, String> map = new HashMap<String, String>();
        //System.out.println("saveArc-succeed");
        /**************************                              debug
********************************/
        map.put("result", "发布成功！");
        return map;
    } catch (Exception e) {
        e.printStackTrace();
        System.out.println("saveArc-failed");
        Map<String, String> map = new HashMap<String, String>();
        /**************************                              debug
********************************/
```

```
            map.put("result", e.getMessage());
            return map;
        }

    }
@Data
    private class Helper{
        public String saveArcContent( String value) {
            /************************** debug ********************************/
//            System.out.println("here is saveArcContent");
//            System.out.println(value);
            /************************** debug ********************************/
            StringBuilder sb = new StringBuilder();
            sb.append(value);
            try {
                // 防止重名
                // String filename = UUID.randomUUID().toString().substring(0,5) + ".html";
                //用时间作为文件名，便于调试
                String filename =System.currentTimeMillis() + ".html";
                String savePath = appConfig.getFilepath()+"/article/arc/" + filename;
                File file = new File(savePath);
                if(!file.getParentFile().exists()){ //判断文件父目录是否存在
                    file.getParentFile().mkdirs();
                }
                if (!file.exists()) {
                    //System.out.println("save:文件不存在");
                    file.createNewFile();
                    //System.out.println("save:文件创建成功");
                }
                FileOutputStream outputStream = new FileOutputStream(savePath);
                BufferedWriter bufferedWriter = new BufferedWriter(new OutputStreamWriter(outputStream, "UTF-8"));
                bufferedWriter.write(sb.toString());
                bufferedWriter.newLine();// 换行
                bufferedWriter.flush();
                bufferedWriter.close();
                return filename;
            } catch (IOException e) {
                return null;
            }
        }

        //获取 html 文章中的第一张图片
        public String getImgFromHtml(String htmlcontent){
            if(htmlcontent!=null){
                String regEx_img ="<img.*src\\s*=\\s*(.*?)[^>]*?>";
                Pattern p_image = Pattern.compile(regEx_img,Pattern.CASE_INSENSITIVE);
                Matcher m_image = p_image.matcher(htmlcontent);
                if(m_image.find()){
                    String img = m_image.group(0);
                    Matcher m = Pattern.compile("src\\s*=\\s*\"?(.*?)(\"|>|\\s+)").matcher(img);
                    if(m.find()){
                        if(m.group(0)!=null){
                            return m.group(0).substring(5, m.group(0).length()-1);
                        }
                    }
                }
            }
            return "";
        }

    }
```