



MyBox：简易工具箱

用户手册 - 网络工具

作者：Mara

版本：5.0

日期：2019-4-16

内容目录

1 资源地址.....	3
2 文档.....	3
3 网络工具的菜单.....	4
4 网页编辑器.....	5
4.1 编辑网页.....	5
4.2 网页代码.....	6
4.3 网页浏览器.....	6
4.4 网页截图.....	7
5 微博截图工具.....	8
5.1 为什么需要这个工具?	8
5.2 可以做什么.....	9
5.3 不能做什么.....	13
5.4 参数说明.....	13
5.5 MyBoX 实现微博截图的原理.....	15
5.5.1 程序实现微博截图的主要麻烦.....	15
5.5.2 如何访问微博的公开数据?	15
5.5.3 微博页面的关键数据.....	16
5.5.4 MyBox 程序的主要逻辑.....	17
5.5.5 实现基础: Java 8 的 JavaFX.....	21

1 资源地址

这是利用 JavaFx 开发的图形化界面程序，目标是提供简单易用的功能，免费开源。项目主页：

<https://github.com/Mararsh/MyBox>

每个版本的源代码、编译好的包、和文档都在 Release 目录下：

<https://github.com/Mararsh/MyBox/releases>

欢迎在线提交软件需求和问题报告：

<https://github.com/Mararsh/MyBox/issues>

云盘地址：

https://pan.baidu.com/s/1fWMRzym_jh075OCX0D8y8A#list/path=%2F

2 文档

本文档介绍 MyBox 的网络工具，下载地址：

<https://github.com/Mararsh/MyBox/releases/download/v5.0/MyBox-UserGuide-5.0-NetworkTools-zh.pdf>

其它文档：

《MyBox 用户手册-综述》

<https://github.com/Mararsh/MyBox/releases/download/v5.0/MyBox-UserGuide-5.0-Overview-zh.pdf>

《MyBox 用户手册-图像工具》

<https://github.com/Mararsh/MyBox/releases/download/v5.0/MyBox-UserGuide-5.0-ImageTools-zh.pdf>

《MyBox 用户手册-桌面工具》

<https://github.com/Mararsh/MyBox/releases/download/v5.0/MyBox-UserGuide-5.0-DesktopTools-zh.pdf>

《MyBox 用户手册-PDF 工具》

<https://github.com/Mararsh/MyBox/releases/download/v5.0/MyBox-UserGuide-5.0-PdfTools-zh.pdf>

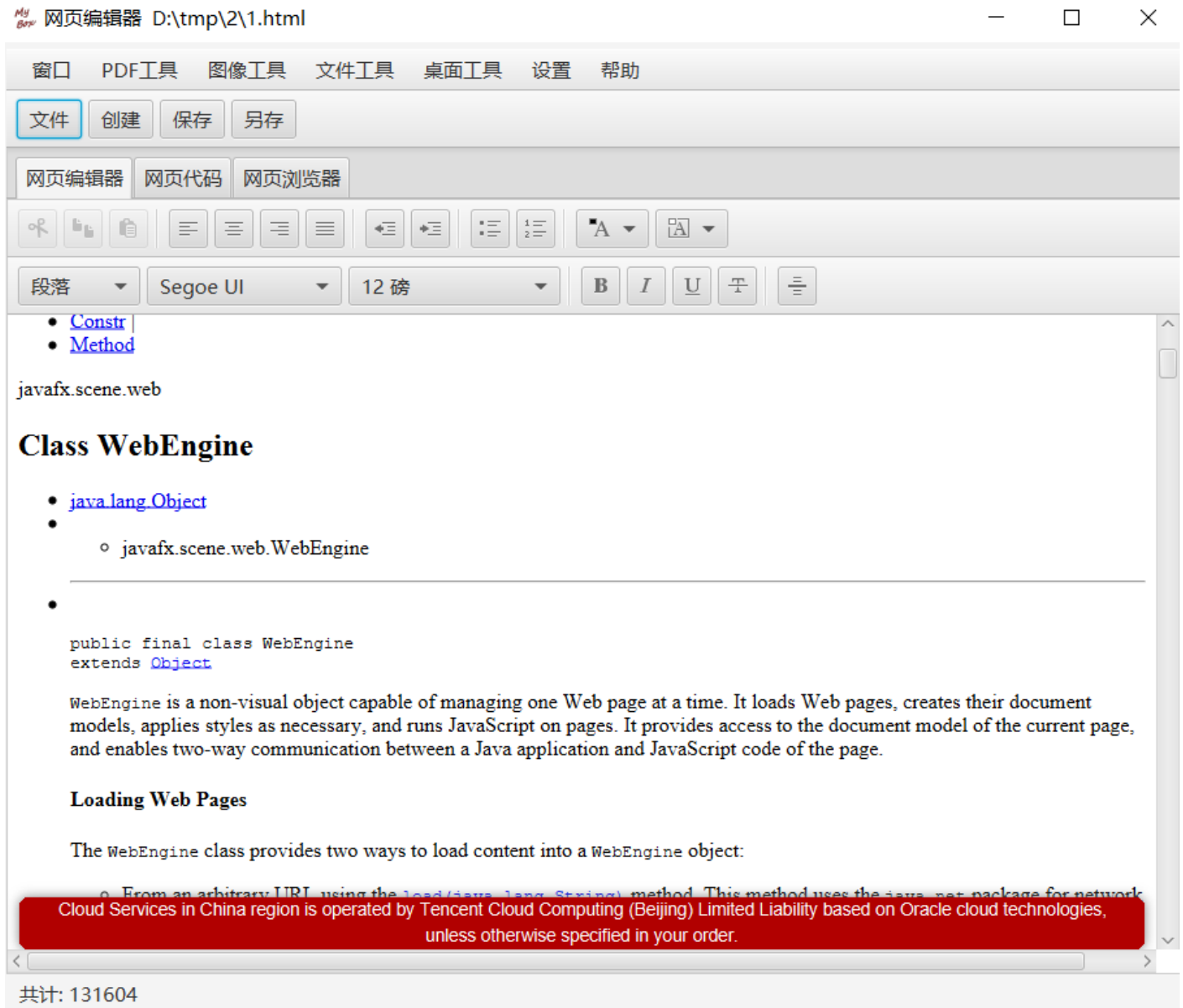
3 网络工具的菜单



4 网页编辑器

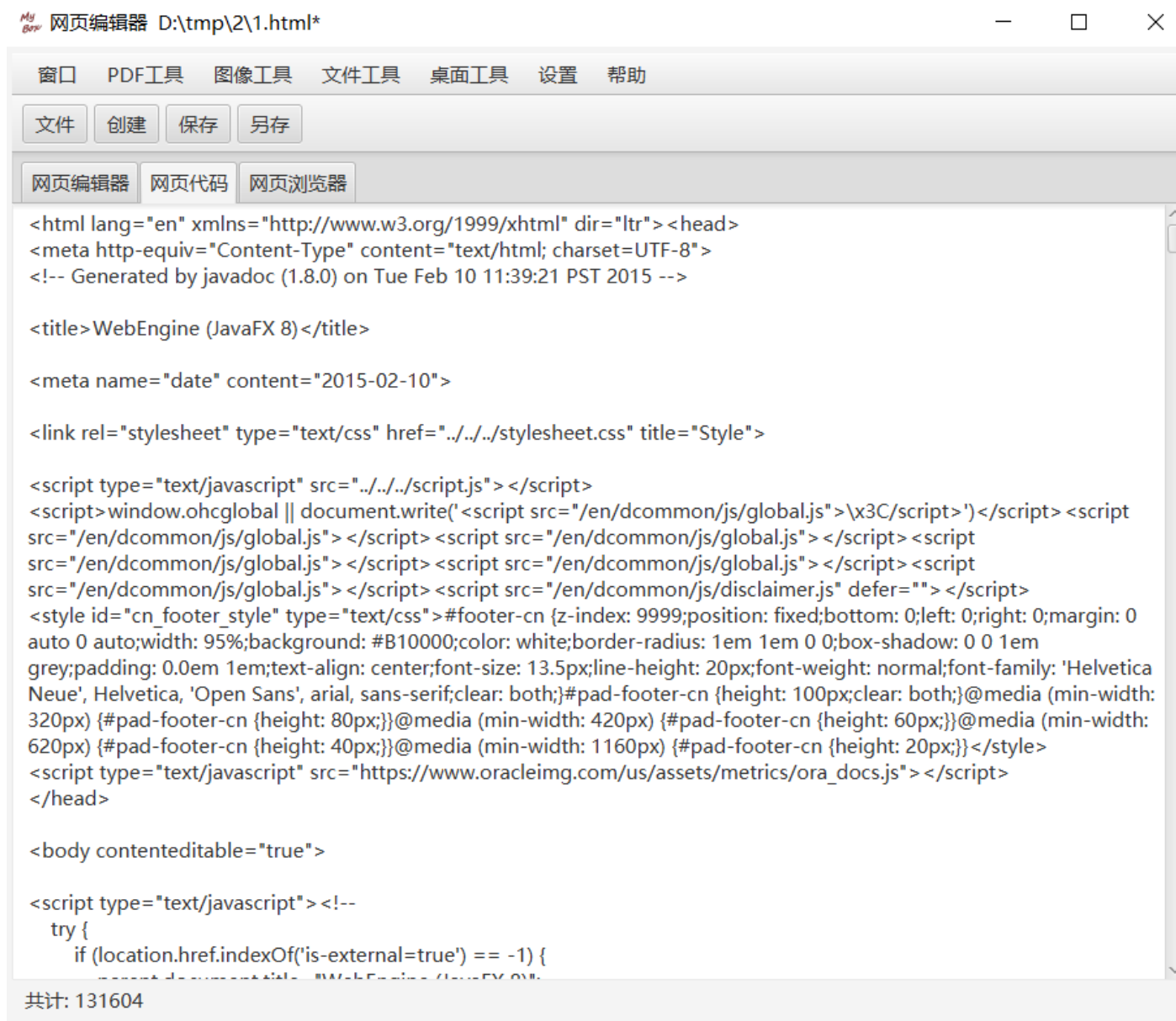
4.1 编辑网页

- 1) 可以打开、新建、保存、另存 html 文件。
- 2) 可以以富文本方式编辑网页，即利用工具条设置文本的颜色、字体、段落等格式。所见即所得。
- 3) 网页编辑器的修改自动同步为网页代码。



4.2 网页代码

- 1) 可以查看和修改网页代码。
- 2) 对于网页代码的直接修改将自动同步到网页编辑器。



4.3 网页浏览器

- 1) 可以输入并加载网址内容。
- 2) 可以将浏览器的内容同步到网页编辑器。
- 3) 可以加载网页编辑器的内容。
- 4) 可以放大、缩小网页字体。

4.4 网页截图

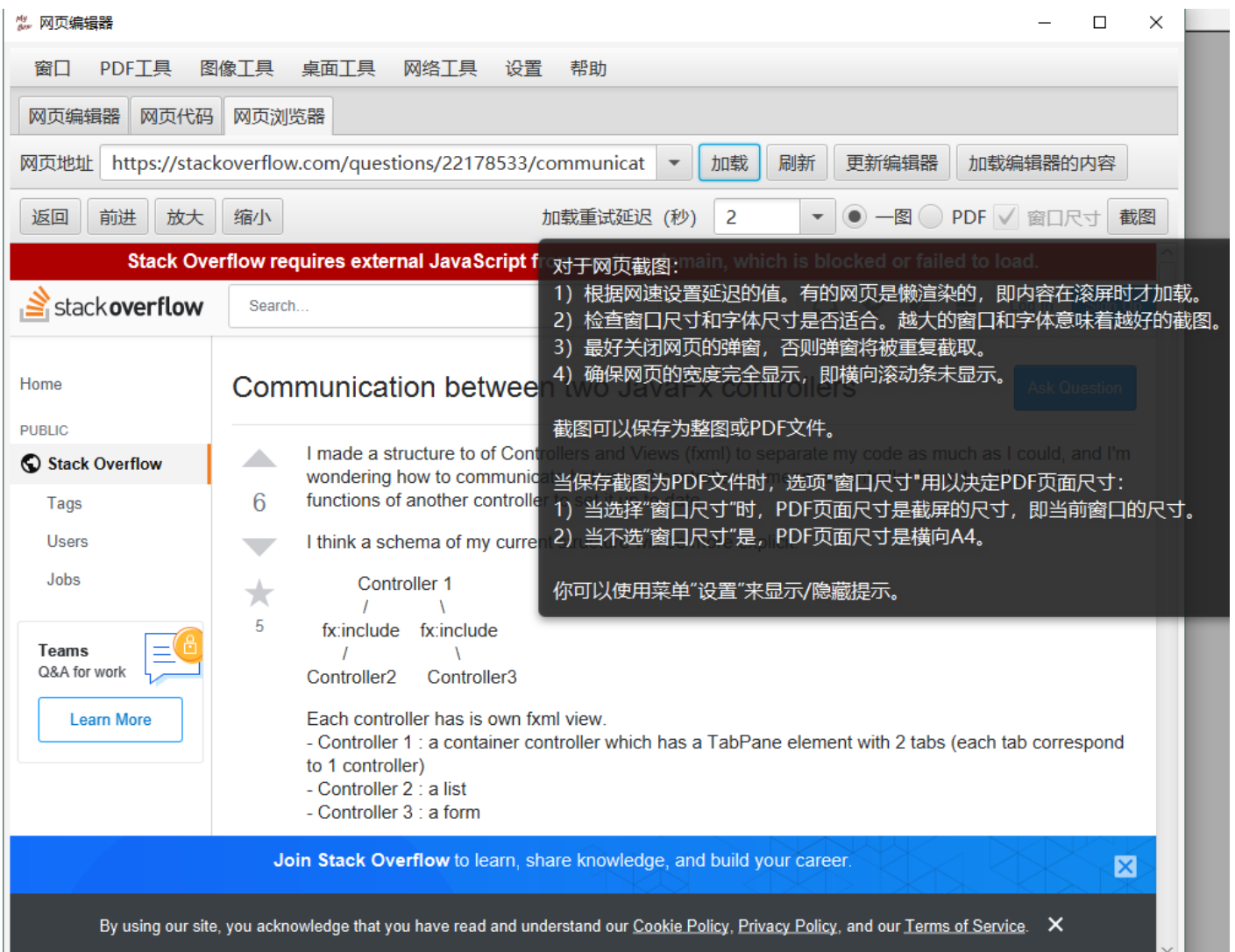
对网页截图之前：

- 1) 确保大部分页面内容已载入并且渲染完毕。
- 2) 根据网速设置延迟的值。有的网页是懒渲染的，即内容在滚屏时才加载。
- 3) 检查窗口尺寸和字体尺寸是否适合。越大的窗口和字体意味着越好的截图。
- 4) 最好关闭网页的弹窗，否则弹窗将被重复截取。
- 5) 确保网页的宽度完全显示，即横向滚动条未显示。

截图可以保存为整图或 PDF 文件。

当保存截图为 PDF 文件时，选项“窗口尺寸”用以决定 PDF 页面尺寸：

- 1) 当选择“窗口尺寸”时，PDF 页面尺寸是截屏的尺寸，即当前窗口的尺寸。
- 2) 当不选“窗口尺寸”是，PDF 页面尺寸是横向 A4。



5 微博截图工具

5.1 为什么需要这个工具?

微博用户可能想要保存自己账户的微博内容，也可能想保存自己在意的微博账户的内容，手动逐页保存？没有意义，因为微博内容是动态加载的，即使把网页保存在本地，也不能正确打开显示。目前还没有发现官方功能可以帮助用户自动备份微博内容。



这个工具很简单，就是利用代码：打开浏览器、载入微博地址，然后截图和保存微博内容。

5.2 可以做什么

- 1) 自动保存任意微博账户的任意月份的微博内容。
- 2) 设置起止月份。
- 3) 确保页面完全加载, 可以展开页面包含的评论、可以展开页面包含的所有图片。
- 4) 将页面保存为本地 html 文件。由于微博是动态加载内容, 本地网页无法正常打开, 仅供获取其中的文本内容。
- 5) 将页面截图保存为 PDF。可以设置页尺寸、边距、作者、以及图片格式。
- 6) 将页面包含的所有图片的原图全部单独保存下来。
- 7) 实时显示处理进度。
- 8) 可以随时中断处理。程序自动保存上次中断的月份并填入作本次的开始月份。
- 9) 可以设置错误时重试次数。若超时错误则自动加倍最大延迟时间。





狩夜猫的博客-2018-09.pdf - Adobe Acrobat Reader DC

文件 编辑 视图(V) 窗口(W) 帮助(H)

主页 工具

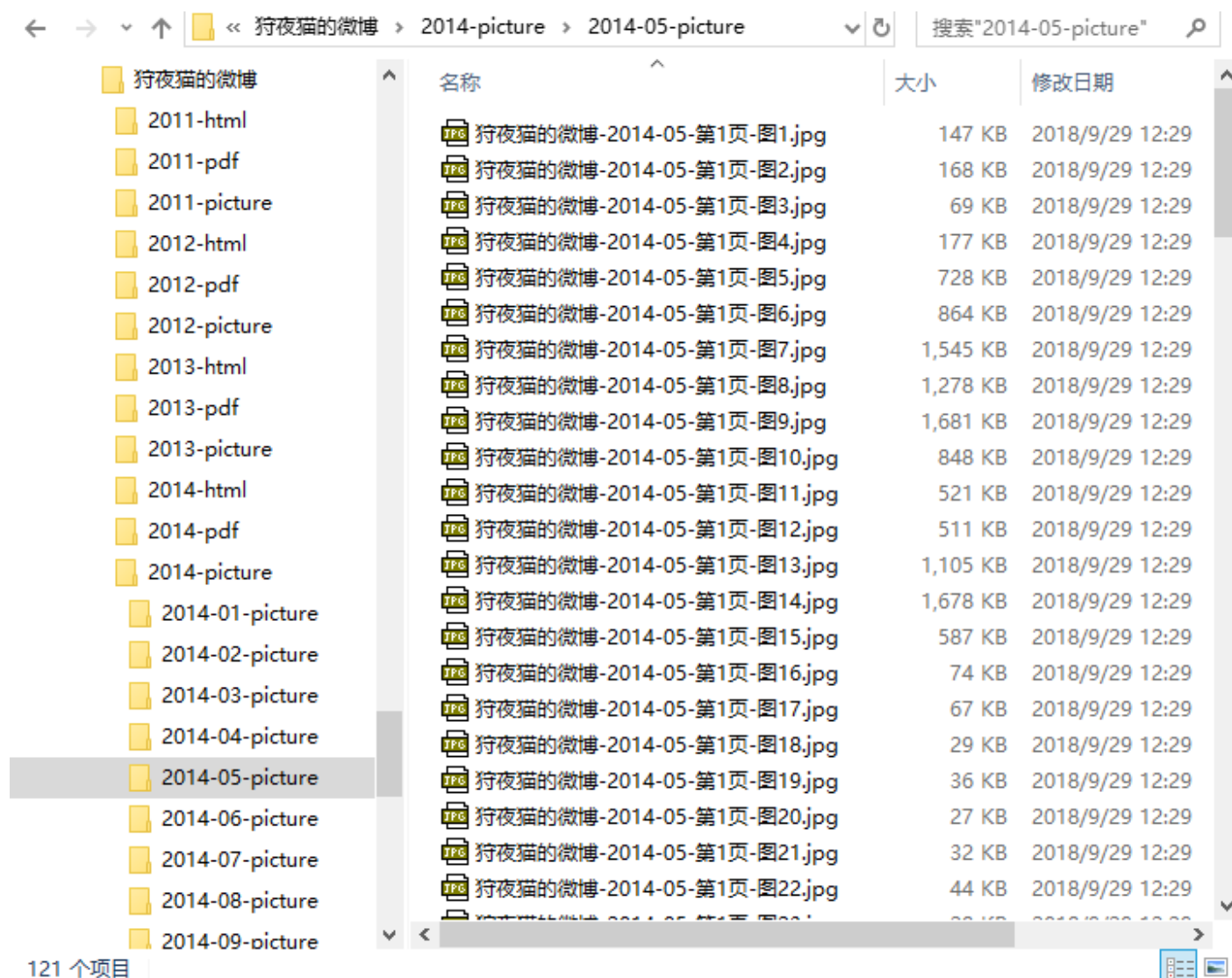
狩夜猫的博客-201... x



收起 | 查看大图 | 向左旋转 | 向右旋转

手工点击无法实现的效果：微博消息中每张图都可以点开、并且同时显示在消息下





截图PDF、html文件、页面原图分别保存在各目录下

5.3 不能做什么

点开长微博全文需要登录所以无法展开。

5.4 参数说明

除了“微博主页地址”和“文件路径”，其它缺省参数是我认为比较好的选择。

MyBox 微博截图

窗口 PDF工具 图像工具 桌面工具 网络工具 设置 最近访问 帮助

▼ 网页选项

微博主页地址 小提示

开始月份 (空白表示从最开始) 开始页 结束月份 (空白表示到现在)

☒ 展开评论 ☒ 展开图片 失败时重试次数

▼ 截图选项

页面放大比例 网页宽度 (宽度合适时页面上无多余区域)

格式 ☒ PNG (无损) ☐ JPEG质量 %

☐ 黑白CCITT group 4 (T6)阈值 (空白以取缺省值) ☐ 抖动处理

▼ PDF选项

页尺寸 ☒ 图片尺寸 ☐ 标准尺寸 72 dpi ☐ 定制 (像素)

页边 作者 缺省显示比例 %

PDF使用的最大主内存 ☐ 500M ☒ 1G ☐ 2G ☐ 不限制

▼ 目标文件

文件路径 选择... 归类文件 ☒ 按月分目录 ☐ 在一个目录下

☒ 保存HTML文件 ☒ 保存图片 ☒ 保存截图为PDF文件 ☒ 保留每一页的PDF

☒ 喵 ☒ 结束时打开目录 ☒ 开始以后关闭本窗口 建议的设置 以魔兽世界的微博为示例 开始

完成时或错误时发出来自乖乖的提示

1) “微博主页地址”即某个微博账户主页地址中“?”之前的字符串。

如何得知主页地址：点击账户的“主页”，它的地址将显示在浏览器的地址栏里。注意：昵称地址不是真的地址。例如，“博物杂志”有昵称地址“https://weibo.com/bowu”，但是它的合法地址实际是“https://weibo.com/p/1002061195054531”。

2) “开始月份”空白表示从第一条微博开始（程序自动判断），“结束月份”空白表示直到现在。注意，微博上线月份是 2009-08。

- 3) “展开评论”可以展开博主消息的评论。“展开图片”可以展开博主消息中的所有图片。
- 4) “失败时重试次数”是由于网络状态不佳而导致页面长时间停滞时程序刷新页面的次数。
- 5) “页面放大比例”可以把页面所有元素缩放数倍。一般来说原尺寸（1 倍）就可以了。

6) “网页宽度”可以设置截图的页面宽度，可以是选择/输入具体值或者全屏。一般来说宽度 700 可以使得微博页面上多余的区域消失、只剩下消息区域，以利于提高截图速度并减小 PDF 大小。若选择全屏，则页面上所有元素都被截图，速度和文件大小都不合适，除非用户就是需要全屏截图。

当调整“页面放大比例”时，相应的也应调整“网页宽度”，否则可能有内容截取不到。

- 7) “格式”：无损的 png、可设置质量的 jpg、黑白色，文件大小依次减小而质量反之。

- 8) “页尺寸”：若方便查看则建议选“图片尺寸”，若方便打印则建议选“A4-横向”。

9) “PDF 使用的最大主内存”：若限制此值，则写 PDF 文件所需内存超过此值时，程序会使用临时文件。此选项很必要：大量图片写入 PDF 文件时，PDF 模块对内存需求量很大，在我的机器上不限制此值时程序直接崩溃。一般设为 1/8 物理内存加好了。

10) “保留每一页的 PDF”，每页的微博会保存为一个 PDF，然后程序会自动合并多页的 PDF 为单月的 PDF，若要同时保留每一页的 PDF 文件，则可选此项。这是冗余数据。

11) 归类文件若选“按月分目录”，则各类文件自动存放在不同的年/月目录下；否则 pdf/html/ 图片文件分别放在一个目录下。

- 11) 选择“喵”则在任务完成时或者出错时会发出来自我家乖乖的提示。

12) 点击按钮“以魔兽世界的微博为示例”，则自动截图和保存魔兽世界某个月的微博内容，可供观察此工具的运行方式。

- 13) 按钮“建议的设置”用来自动填写缺省值。

5.5 MyBoX 实现微博截图的原理

5.5.1 程序实现微博截图的主要麻烦

- 1) 如何访问微博页面？需要解决浏览器加装微博证书的问题。
- 2) 如何判断页面已加载完毕？微博页面是动态加载的，当页面滚动时，页面才逐屏显示。程序需要模拟滚屏、并且正确判断是否加载完毕。当需要展开图片时，更没有确凿的依据来判断页面是否已加载完毕。
- 3) 如何避免浮动窗口覆盖而丢失的数据？微博页面底部总是浮动一条窗口，没找到怎么关闭的方法，造成每屏截图底部会丢失一部分信息，例如一条消息被覆盖若干行，很影响阅读。解决办法是截图时取一些重复量，即上下两屏中间有相同的若干行像素。
- 4) 如果想保存页面图片的原图，如何找到地址？经过分析微博页面代码，可以找到了规律。

5.5.2 如何访问微博的公开数据？

- 1) 微博用户主页的右侧是每个月历史的访问地址，无需登录可访问。



- 2) 如以下网址可直接访问月球车玉兔 2013 年 12 月的第一页微博内容：

https://weibo.com/u/3926428816?is_all=1&stat_date=201312&page=1

5.5.3 微博页面的关键数据

1) 在主页代码中搜索 “stat_date”，即得所有月份的列表。如：

```
<a href="javascript:void(0);" class="S" action-type="login" action-data="is_all=1&stat_date=201807"
suda-uatrack="key=Profile_V6_Timeline&value=month"><em class="bor_t"></em><em
class="S_dot"></em><em class="bor_b"></em><span>7 月</span></a>
```

2) 页面的状态：

- (1) 包含"还没有发过微博"表示此月无数据。
- (2) 包含“查看更早微博”表示页面内容全部加载，这是此月最后一页。
- (3) 包含"currentPage="和"countPage="表示页面内容全部加载，这是当前月的当前页和总页数。

3) 消息的评论链接：

```
<a href="javascript:void(0);" class="S_txt2" action-type="fl_comment" action-
data="oid=1444865141&location=profile&comment_type=0" suda-
uatrack="key=profile_feed&value=comment:923874989"><span class="pos"><span class="line
S_line1" node-type="comment_btn_text"><span><em class="W_ficon ficon_repeat
S_ficon"> </em><em>149</em></span></span></span></a>
```

4) 消息的图片链接有三类，点击所有这些元素就可以展开页面中所有图片了：

```
<li class="WB_pic S_bg2 bigcursor" suda-
uatrack="key=comment_pic_click&value=cmt_thumbnail_click" action-type="comment_media_img"
action-data="pid=0061XAWoly1furz0kqwrlj30sp0xrnbw&cid=4278804458837161"><i class="W_loading"
style="margin: 74px 74px 74px -90px; display: none;"></i></li>
```

```
<li class="WB_pic li_1 S_bg1 S_line2 bigcursor li_n_mix_w" action-
data="isPrivate=0&relation=0&pid=561ee475gy1fursmpp6pbj20zk0qotid&object_ids=10420
18%3A76c124f20aaf2b908244de97cbdc37ce&photo_tag_pids=&uid=1444865141&mid=42
78749829157046&pic_ids=561ee475gy1fursmpp6pbj20zk0qotid&pic_objects=" action-
type="feed_list_media_img" suda-
uatrack="key=tblog_newimage_feed&value=image_feed_unfold:4278749829157046:561ee475gy1fur
smpp6pbj20zk0qotid:1444865141:0"> <i class="W_loading"
style="display: none;"></i>
```

```
<li class="WB_pic li_1 S_bg1 S_line2 bigcursor " action-
data="isPrivate=0&relation=0&pic_id=8ccdd811gy1fvmsp4wf77j218s0q2k09" action-
type="fl_pics" suda-
uatrack="key=tblog_newimage_feed&value=image_feed_unfold:4288467662528718:8ccdd811gy1fv
msp4wf77j218s0q2k09:2362300433:0"></li>
```


5) 以上三类图片链接中, 把以下字串换成 “large” 就是原图的地址:

```

```

```

```

```

```

5.5.4 MyBox 程序的主要逻辑


1) 主要逻辑都在 WeiboSnapRunController.java 中

2) loadMain() 的主要逻辑: 在主页 html 代码中找到微博账户名及其发布微博起止月份

```
int posAccount2 = contents.indexOf("_微博</title>");
if (posAccount1 > 0 && posAccount2 > 0) {
    accountName = contents.substring(posAccount1 + "<title>".length(), posAccount2);
    int posfirst1 = contents.indexOf("&stat_date=");
    if (posfirst1 > 0) {
        String s = contents.substring(posfirst1 + "&stat_date=".length());
        int posfirst2 = s.indexOf("\\");
        if (posfirst2 > 0) {
            try {
                s = s.substring(0, posfirst2);
                lastMonth = DateTools.parseMonth(s.substring(0, 4) + "-" + s.substring(4, 6));
                logger.debug(DateTools.datetimeToString(lastMonth));
                int posLast1 = contents.lastIndexOf("&stat_date=");
                if (posLast1 > 0) {
                    s = contents.substring(posLast1 + "&stat_date=".length());
                    int posLast2 = s.indexOf("&page=");
                    if (posLast2 > 0) {
                        try {
                            s = s.substring(0, posLast2);
                            firstMonth = DateTools.parseMonth(s.substring(0, 4) + "-" + s.substring(4, 6));
                            logger.debug(DateTools.datetimeToString(firstMonth));
                            loadCompleted = true;
                        } catch (Exception e) {
                            logger.error(e.toString());
                        }
                    }
                }
            } catch (Exception e) {
                logger.error(e.toString());
            }
        }
    }
} catch (Exception e) {
    logger.error(e.toString());
}
```

loadMain() 的主要逻辑:
在主页html代码中找到微博账户名及其发布微博起止月份

3) loadPage()的主要逻辑：判断页面内容是否已完全加载完毕



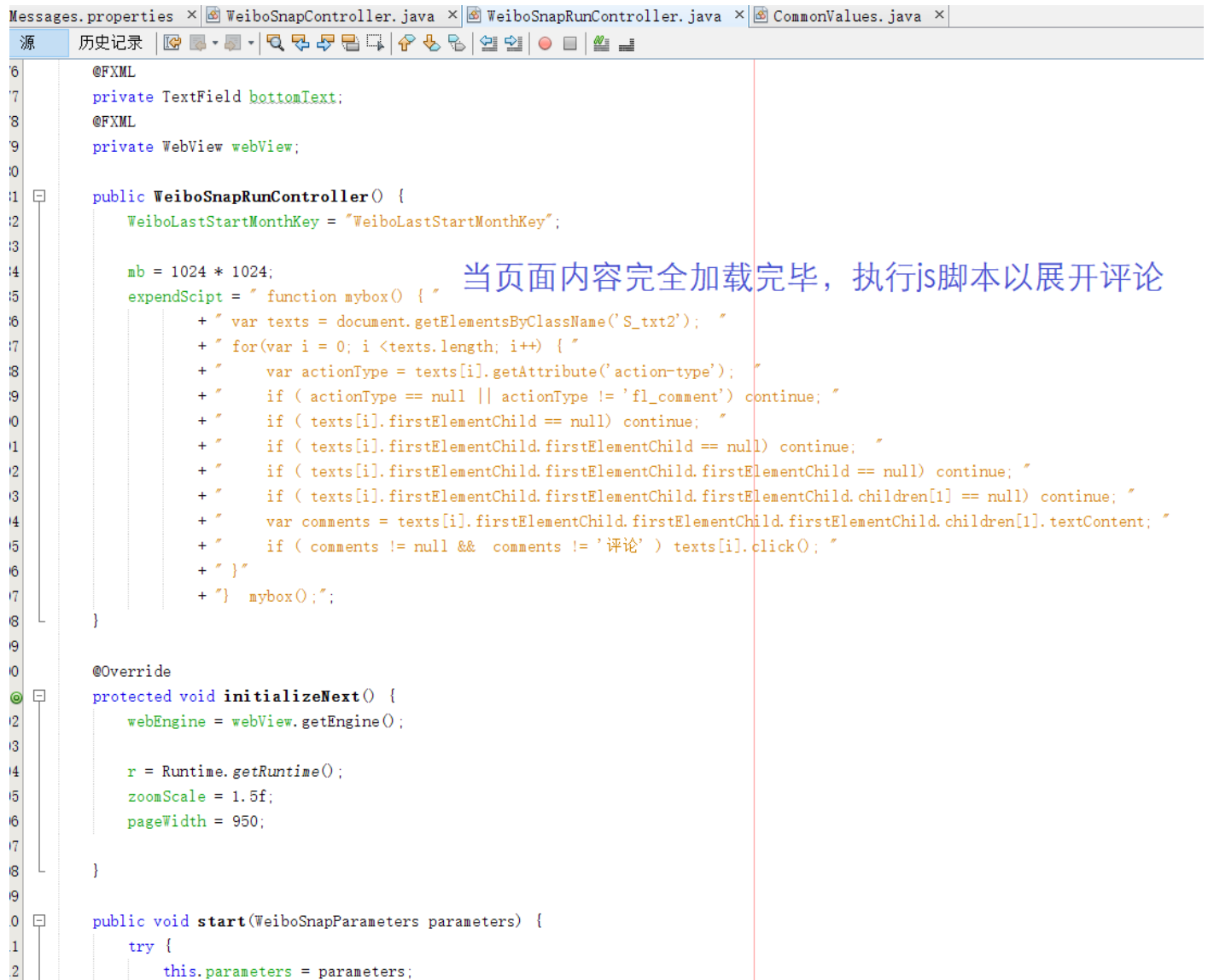
```

ler.java x WeiboSnapRunController.java x CommonValues.java x
contents = (String) webEngine.executeScript("document.documentElement.outerHTML");
if (mainCompleted) {
    loadCompleted = true;
} else if (contents.contains("查看更早微博")) {
    currentPage = 1;
    pageCount = 1;
    webEngine.executeScript(expendScript);
    Thread.sleep(loadDelay);
    mainCompleted = true;
} else if (contents.contains("还没有发过微博")) {
    currentPage = 0;
    pageCount = 0;
    loadCompleted = true;
    emptyPage = true;
} else {
    int pos1 = contents.indexOf("action-data=\"currentPage=");
    if (pos1 > 0) {
        String s1 = contents.substring(pos1 + "action-data=\"currentPage=".length());
        int pos2 = s1.indexOf("&countPage=");
        if (pos2 > 0) {
            int pos3 = s1.indexOf("\"");
            if (pos3 > 0) {
                try {
                    currentPage = Integer.valueOf(s1.substring(0, pos2));
                    pageCount = Integer.valueOf(s1.substring(pos2 + "&countPage=".length(), pos3));
                    if (parameters.isExpandComments()) {
                        webEngine.executeScript(expendScript);
                        Thread.sleep(loadDelay);
                    } else {
                        loadCompleted = true;
                    }
                    mainCompleted = true;
                } catch (Exception e) {
                    loadFailed = loadCompleted = true;
                    errorString = e.toString();
                    logger.debug(e.toString());
                }
            }
        }
    }
}

```

loadPage() 的主要逻辑：
判断页面内容是否已完全记载完毕

4) 页面内容完全加载完毕以后, 执行 javascript 脚本以展开评论, 并保存 html 代码

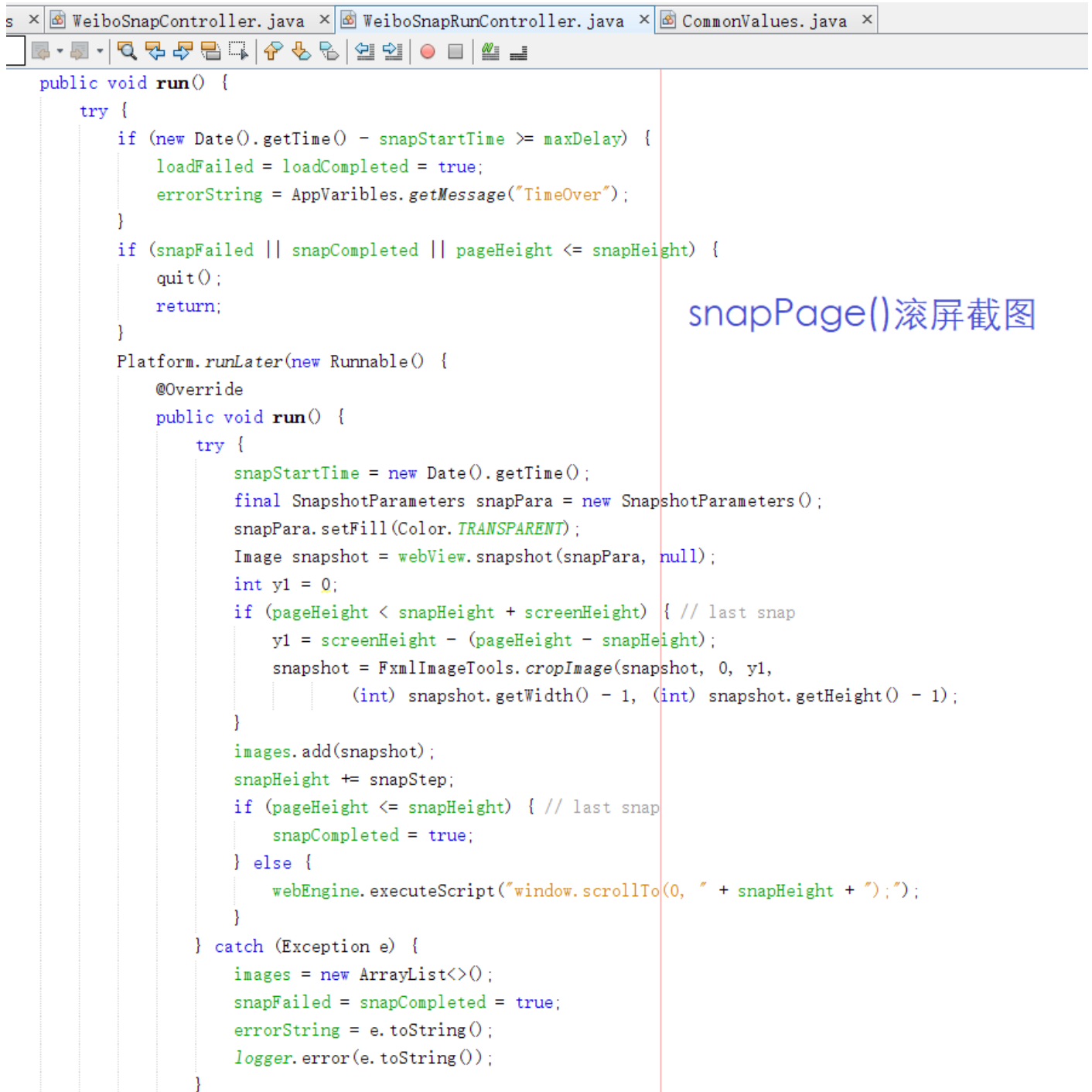


```

Messages.properties x WeiboSnapController.java x WeiboSnapRunController.java x CommonValues.java x
源 历史记录
6 @FXML
7 private TextField bottomText;
8 @FXML
9 private WebView webView;
10
11 public WeiboSnapRunController() {
12     WeiboLastStartMonthKey = "WeiboLastStartMonthKey";
13
14     mb = 1024 * 1024;
15     expendScript = " function mybox() { " 当页面内容完全加载完毕, 执行js脚本以展开评论
16         + " var texts = document.getElementsByClassName('S_txt2'); "
17         + " for(var i = 0; i < texts.length; i++) { "
18         + "     var actionType = texts[i].getAttribute('action-type'); "
19         + "     if ( actionType == null || actionType != 'fl_comment') continue; "
20         + "     if ( texts[i].firstElementChild == null) continue; "
21         + "     if ( texts[i].firstElementChild.firstElementChild == null) continue; "
22         + "     if ( texts[i].firstElementChild.firstElementChild.firstElementChild == null) continue; "
23         + "     if ( texts[i].firstElementChild.firstElementChild.firstElementChild.children[1] == null) continue; "
24         + "     var comments = texts[i].firstElementChild.firstElementChild.firstElementChild.children[1].textContent; "
25         + "     if ( comments != null && comments != '评论' ) texts[i].click(); "
26         + " } "
27         + " } mybox();"
28     }
29
30 @Override
31 protected void initializeNext() {
32     webEngine = webView.getEngine();
33
34     r = Runtime.getRuntime();
35     zoomScale = 1.5f;
36     pageWidth = 950;
37
38 }
39
40 public void start(WeiboSnapParameters parameters) {
41     try {
42         this.parameters = parameters;
43     }
44 }

```

5) snapPage()滚屏截图并保存到 PDF



5.5.5 实现基础：Java 8 的 JavaFX

1) JavaFX 的内置浏览器 WebView

2) WebView 执行 Javascript 与页面交互：

```
webEngine.executeScript("window.scrollTo(0, " + snapHeight + ");");
```

3) JavaFX 任意 Node 的截图函数

```
Image snapshot = webView.snapshot(snapPara, null);
```

4) Apache 开源库 PDFBox

<https://pdfbox.apache.org/>

<文档结束>