

# 1 简答题

## 1.1 简答题一

如下程序执行完毕以后，a.txt文件中的内容是什么并说明原因？

<pre>// 创建FileWriter对象 FileWriter fw = new FileWriter( fileName: "a.txt" ) ;  // 调用write方法写数据 fw.write( str: "传智教育");</pre>	
-----------------------------------------------------------------------------------------------------------------------------	--

答：  
执行效果：在a.txt文件中没有任何内容  
原因说明：使用字符流写数据并不是直接将数据写入到磁盘中，而是先写入到了缓存区中。而代码中没有对缓存区进行刷新也没有关闭流对象，因此数据就没有被写入到磁盘中。

## 1.2 简答题二

简述flush方法和close方法的区别？

答：  
① flush方法表示刷新缓存区，调用完flush方法以后流对象还可以继续使用  
② close方法关闭流对象，在关闭之前会执行刷新缓存区的操作，流对象一旦被关闭了，就不能再使用了。

## 1.3 简答题三

请分析如下程序执行完毕以后a.txt文件的内容是什么并说明原因？如果需要在a.txt文件中保存所有数据应该怎么处理？

<pre>FileWriter fw1 = new FileWriter( fileName: "a.txt"); fw1.write( str: "中国加油!"); fw1.close();  FileWriter fw2 = new FileWriter( fileName: "a.txt"); fw2.write( str: "传智播客!"); fw2.close();</pre>	
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

答：  
① 上述程序执行完毕以后a.txt文件中的内容为：传智播客。因为第二次在写数据的时候，创建了一个新的流对象，这种创建方式会清空原有文件中的内容。  
② 要留所写的所有数据：在创建fw2对象的时候，使用如下方式：  
FileWriter fw2 = new FileWriter("a.txt", true);

## 1.4 简答题四

运行下述程序会出现什么错误？请说明出现该错误的原因以及如何解决该错误？

<pre>public class User {      private int age ;     private String name ;      // 省略get/set方法 }  class Demo01 {      public static void main(String[] args) throws IOException {         ObjectOutputStream oos = new ObjectOutputStream(new FileOutputStream( name: "obj.txt"));         oos.writeObject(new User());         oos.close();     } }</pre>	
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

答：  
① 运行上述程序会出现如下异常：Exception in thread "main" java.io.NotSerializableException: com.itheima.io.User  
② 原因：使用序列化流对对象进行序列化操作的时候，要求序列化对象所对应的类实现序列化接口。而User类并没有实现序列化接口。  
③ 解决方案：当User类实现序列化接口。如果所示：

## 1.5 简答题五

请分析产生如下错误的原因以及对应的解决方案？

```
Exception in thread "main" java.io.InvalidClassException: com.itheima.io.obj.User; local class incompatible: stream classdesc serialVersionUID = 4001622083751882644, local class serialVersionUID = 6175241864707791838
    at java.io.ObjectStreamClass.initNonProxy(ObjectStreamClass.java:687)
    at java.io.ObjectInputStream.readNonProxyDesc(ObjectInputStream.java:1883)
    at java.io.ObjectInputStream.readClassDesc(ObjectInputStream.java:1749)
    at java.io.ObjectInputStream.readOrdinaryObject(ObjectInputStream.java:2040)
    at java.io.ObjectInputStream.readObject0(ObjectInputStream.java:1571)
    at java.io.ObjectInputStream.readObject(ObjectInputStream.java:431)
    at com.itheima.io.obj.Demo01.main(User.java:21)
```

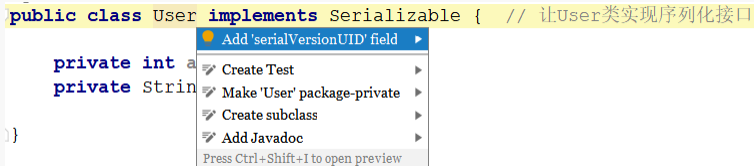
答：

① 错误原因：序列化需要一个serialVersionUID（序列化ID），修改完类文件了以后serialVersionUID与序列化之前不一致了，校验失败了，就抛出异常了。

② 解决方案：给对象生成一个固定serialVersionUID

具体实现步骤：

1. idea设置：setting ---> Inspections ---> 搜索serialVersionUID ---> 勾选"Serializable class without serialVersionUID"
2. 然后光标放在类明上，使用快捷键：alt + enter，此时会弹出如下对话框：



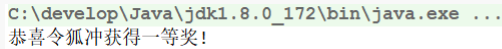
## 2 编程题

### 2.1 编程题目一

**训练目标：**掌握java中高效的字符流的基本使用，以及理解其在实际开发中的应用

**需求描述：**某公司年会需要搞抽奖活动以此感谢公司员工这一年的辛勤付出。要求当某一个员工已经被抽取到了，再抽取下一个奖项的时候该员工不能再次被抽到。请编写程序实现该抽奖活动。

程序的运行效果如下所示：



**实现提示：**

- 1、在提供的素材(lucky)项目中的指定位置补全代码实现功能
- 2、使用BufferedReader读取employee\_names.txt文件中的数据，并将其存储到一个集合中
- 3、生成一个随机数，作为集合元素的索引
- 4、根据该索引从集合中获取一个员工姓名，并将该员工姓名从集合中移除掉
- 5、将集合中剩余的员工姓名通过BufferedWriter再一次写入到employee\_names.txt文件中

**代码实现：**见《答案/代码/lucky》

### 2.2 编程题目二

**训练目标：**掌握java中Properties集合的使用，以及理解其在实际开发中的应用

**需求背景：**电子邮件的应用非常广泛，常见的如在某网站注册了一个账户，自动发送一封激活邮件，通过邮件找回密码，自动批量发送活动信息等。很显然这些应用不可能和我们自己平时发邮件

一样，先打开浏览器，登录邮箱，创建邮件再发送。而是通过程序实现邮件的发送。在素材文件中提供了一个项目prop，可以实现通过Java代码发送邮件。如下所示：

```
// 1. 创建参数配置，用于连接邮件服务器的参数配置
Properties props = new Properties(); // 参数配置
props.setProperty("mail.transport.protocol", "smtp"); // 使用的协议（JavaMail规范要求）
props.setProperty("mail.smtp.host", "smtp.163.com"); // 发件人的邮箱的SMTP服务器地址
props.setProperty("mail.smtp.auth", "true"); // 需要请求认证

// 2. 创建一个Session对象，用于和发送邮件程序进行交互
Session session = Session.getDefaultInstance(props);

// 3. 创建一封邮件
MimeMessage message = new MimeMessage(session);
message.setFrom(new InternetAddress("itcast_hly_01@163.com")); // 设置发件人邮箱地址
message.setRecipient(MimeMessage.RecipientType.TO, new InternetAddress("342245209@qq.com")); // 设置收件人邮箱地址
message.setSubject("打折钜惠"); // 设置邮箱主题
message.setContent("XX用户你好，今天全场5折，快来抢购，错过今天再等一年。。。", "text/html;charset=UTF-8"); // 设置邮箱正文

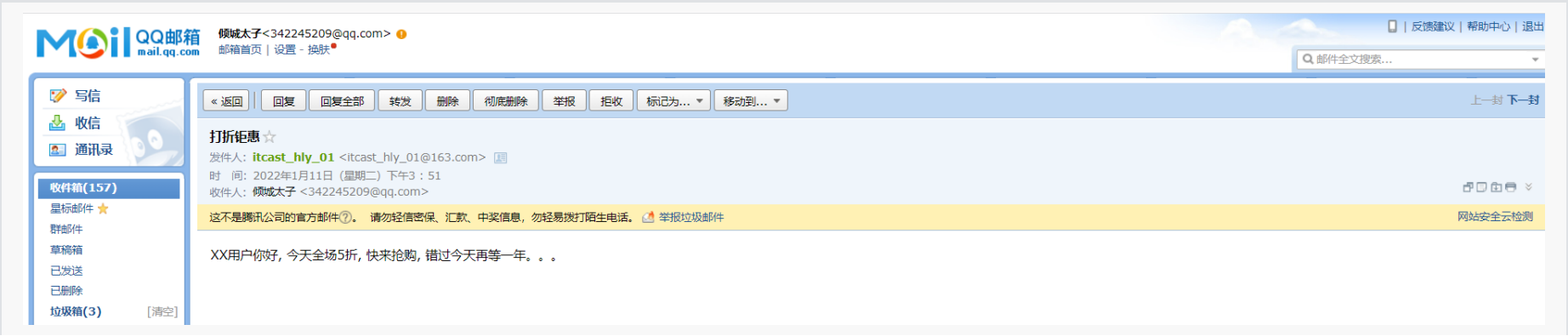
// 4. 根据 Session 获取邮件传输对象
Transport transport = session.getTransport();

// 5. 建立连接
transport.connect(user: "itcast_hly_01@163.com", password: "MJVZWZZTVJKQSWXM");

// 6. 发送邮件，发到所有的收件地址，message.getAllRecipients() 获取到的是在创建邮件对象时添加的所有收件人，抄送人，密送人
transport.sendMessage(message, message.getAllRecipients());

// 7. 关闭连接
transport.close();
```

上述代码实现了：通过网易邮箱itcast\_hly\_01@163.com向342245209@qq.com发送了一份邮件。效果如下所示：



但是上述程序存在一定的弊端：硬编码。这样不利于后期代码的维护，比如要修改发件人、收件人、邮件主题、邮件正文等信息的时候都需要对代码进行修改。

**需求描述：**使用Properties集合对上发送邮件的代码进行优化，以提高代码的维护性。

**实现提示：**

1、上述程序中关于发件人和收件人是使用老师提供的账号。如果需要使用自己的账号来实现邮件的发送，请注册网易云邮箱，注册地址：<https://mail.163.com/>。注册完毕以后需要开通指定的

服务才可以实现邮件的发送。如下所示：



**注意：**保存好授权密码，程序中需要使用。

2、上述发送邮件程序中可更改的参数：发件人邮箱账号信息、收件人收件账号、邮箱主题、邮箱正文

3、创建一个properties属性文件，将上述可更改的内容定义到该文件中

4、在程序中读取第三步创建的文件内容到一个Properties集合中

5、从Properties集合中获取参数信息更改原有程序

6、电脑需要联网

**代码实现：**见《答案/代码/prop》



## 2.3 编程题目三

**训练目标：**掌握java中高效的字符流和转换流的基本使用，以及理解其在实际开发中的应用

**需求背景：**网络爬虫（又称为网页蜘蛛，网络机器人，在FOAF社区中间，更经常的称为网页追逐者），是一种按照一定的规则，自动地抓取万维网信息的程序。通过程序请求指定的网页，获取网

页数据对应的字节输入流对象，然后通过该输入流读取网页数据，再次对网页数据进行筛选获取有价值的信息，然后再将数据存储起来用于后期做数据分析。

**需求描述：**在素材的server.jar程序中存在两个网页：1.html和2.html。现需要将这两个网页中"西安黑马"就业薪资信息，保存到一个本地文件中。程序运行完毕以后，本地文件中的数据如下所示：

示：

毕业0个工作日，平均薪资10626.67元，就业率24.59%，西安黑马 前端学科20210728班	
毕业19个工作日，平均薪资9549.44元，就业率54.55%，西安黑马JavaEE学科20210711班	
毕业34个工作日，平均薪资9663.64元，就业率83.02%，西安黑马 JavaEE20210619学科班	
毕业59个工作日，平均薪资9421.05元，就业率80.85%，西安黑马 前端学科20210429班	
毕业60个工作日，平均薪资10517.78元，就业率86.54%，西安黑马 Java学科20210510班	
毕业25个工作日，平均薪资9635.14元，就业率90.24%，西安黑马前端学科20210622班	
毕业11个工作日，平均薪资10062.50元，就业率36.36%，西安黑马 JavaEE学科20210724班	
毕业19个工作日，平均薪资9549.44元，就业率54.55%，西安黑马JavaEE学科20210711班	
毕业34个工作日，平均薪资9663.64元，就业率83.02%，西安黑马 JavaEE20210619学科班	
毕业60个工作日，平均薪资10517.78元，就业率86.54%，西安黑马 Java学科20210510班	
毕业11个工作日，平均薪资10062.50元，就业率36.36%，西安黑马 JavaEE学科20210724班	
毕业66个工作日，平均薪资8032.73元，就业率94.83%，西安黑马JavaEE学科20210416班	
毕业66个工作日，平均薪资10362.75元，就业率94.44%，西安黑马JavaEE学科20210326班	

**实现提示：**

1、通过 `java -jar server.jar` 运行server.jar程序，程序运行效果如下所示：

```
PS C:\Users\Administrator\Desktop\JavaSE进阶作业\产出\JavaSE进阶\day12\素材> java -jar .\server.jar

:: Spring Boot :: (v2.3.9.RELEASE)

2022-01-11 13:08:24.738 INFO 7260 --- [main] com.itheima.crawler.Entry : Starting Entry v1.0-SNAPSHOT on DESKTOP-NB5P4S5 with PID 7260 (C:\Users\Administrator\Desktop\JavaSE进阶作业\产出\JavaSE进阶\day12\素材\server.jar started by Administrator in C:\Users\Administrator\Desktop\JavaSE进阶作业\产出\JavaSE进阶\day12\素材)
2022-01-11 13:08:24.740 INFO 7260 --- [main] com.itheima.crawler.Entry : No active profile set, falling back to default profiles: default
2022-01-11 13:08:25.578 INFO 7260 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 80 (http)
2022-01-11 13:08:25.587 INFO 7260 --- [main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2022-01-11 13:08:25.587 INFO 7260 --- [main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.43]
2022-01-11 13:08:25.634 INFO 7260 --- [main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2022-01-11 13:08:25.634 INFO 7260 --- [main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 842 ms
2022-01-11 13:08:25.756 INFO 7260 --- [main] o.s.s.concurrent.ThreadPoolTaskExecutor : Initializing ExecutorService 'applicationTaskExecutor'
2022-01-11 13:08:25.876 INFO 7260 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 80 (http) with context path ''
2022-01-11 13:08:25.883 INFO 7260 --- [main] com.itheima.crawler.Entry : Started Entry in 1.456 seconds (JVM running for 1.737)
2022-01-11 13:08:57.095 INFO 7260 --- [p-nio-80-exec-2] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring DispatcherServlet 'dispatcherServlet'
2022-01-11 13:08:57.095 INFO 7260 --- [p-nio-80-exec-2] o.s.web.servlet.DispatcherServlet : Initializing Servlet 'dispatcherServlet'
2022-01-11 13:08:57.102 INFO 7260 --- [p-nio-80-exec-2] o.s.web.servlet.DispatcherServlet : Completed initialization in 7 ms
```

2、server.jar程序启动起来以后，可以直接通过浏览器访问该程序中的1.html和2.html查看页面内容，如下所示访问的是1.html页面中的内容：



查看2.html网页内容，在浏览器的地址栏输入该地址：<http://localhost/html/2.html>

3、通过程序访问server.jar程序中的网页代码已在素材crawler项目中提供好了

4、在素材crawler项目的指定位置补全代码实现功能

5、进行字符串中特定字符的替换可以考虑使用正则表达式

**代码实现：**见《答案/代码/crawler》

