

# 1 简答题

## 1.1 简答题一

简述Stream流的使用思想以及获取Stream流的常见方式？

答：

## 1.2 简答题二

如下程序

```
// 创建一个List集合对象
List<String> list = Arrays.asList("hello", "java8", "stream");

// 获取上述List集合所对应的流对象
Stream<String> stream = list.stream();

// 遍历流中元素
stream.forEach( s -> System.out.println(s) );
stream.forEach( s -> System.out.println(s) );
```

上述程序执行完毕以后，在控制台的输出结果如下所示：

```
C:\develop\Java\jdk1.8.0_172\bin\java.exe ...
hello
java8
stream
Exception in thread "main" java.lang.IllegalStateException Create breakpoint: stream has already been operated upon or closed
    at java.util.stream.AbstractPipeline.sourceStageSpliterator (AbstractPipeline.java:279) <1 internal call>
    at com.itheima.stream.StreamDemo01.main (StreamDemo01.java:32)
```

请说明出现该错误的原因？

答：

## 1.3 简答题三

运行如下程序观察控制台输出结果，说明产生该问题的主要原因？

```
// 创建一个List集合对象
List<String> list = Arrays.asList("hello", "java8", "stream");

// 获取上述list集合所对应的流对象
list.stream().map(s -> {
    list.add("world") ;
    return s.toUpperCase();
}).forEach( s -> System.out.println(s));

// 向list集合中添加元素
// 将流中的元素转换成大写
// 遍历流中的元素
```

答：

## 1.4 简答题四

请说明如下程序执行完毕以后的结果并说明原因？



```
package com.itheima.file;

import java.io.File;

public class FileDemo01 {

    public static void main(String[] args) {

        // 创建一个File对象f1
        File f1 = new File( pathname: "a.txt" ) ;
        f1.mkdir();

        // 创建一个File对象f2
        File f2 = new File( pathname: "itheima/xian" ) ;
        f2.mkdir();

        // 创建一个File对象f3
        File f3 = new File( pathname: "itcast" );
        f3.delete() ;

    }

}
```

答：

## 1.5 简答题五

如下程序是计算1-10之间的数据和，请说明如下程序执行完毕以后的结果并说明原因？如果想让程序进行正常执行，应该如何进行改造？

```
public class FileDemo02 {  
  
    public static void main(String[] args) {  
  
        // 调用calc方法计算1-10之间的数据之和  
        int result = calc( number: 10);  
        System.out.println(result);  
  
    }  
  
    // 计算1-number之间的数据之和  
    public static int calc(int number) {  
        return number + calc( number: number - 1) ;  
    }  
  
}
```

答：

## 2 编程题

### 2.1 编程题目一

**训练目标：**掌握JDK1.8的新特性Stream流的基本使用，以及理解其在实际开发中的应用

**需求背景：**Microsoft Excel是Microsoft(微软)为使用Windows操作系统的电脑编写的一款电子表格软件。在实际的工作中，如果要处理的数据量较大，那么此时会由多个业务人员将要处理的数据

录入到excel中，那么在进行录入的时候可能会产生数据重复录入。因此在录入完毕以后，往往需要进行数据查重和去重操作。

**需求描述：**在素材的stream01项目的提供了两个excel文件(sku01.xlsx、sku02.xlsx)，这两个excel文件存储的都是商品的数据，在这两个文件中存在一些重复性的数据。现需要在该项目的指定

位置补全相关的代码实现数据去重的操作，并且需要按照商品的价格进行降序排序。程序执行完毕以后可以将处理以后的结果数据存储到一个新的excel文件(sku.xlsx)中。新excel文件的内容如下

所示：

	商品编号	商品条码	商品名称	商品价格	库存数量	库存预警数量	商品图片	重量	创建时间	更新时间	规格	销量	商品状态					
1																		
2	1003963	100000003145001003963	华为Meta1003963	913963	100	20	http://www.chai	0	0002-11-30 (2020-11-08	金立		0	0					
3	1003961	1000000003145001003961	华为Meta1003961	913961	100	20	http://www.chai	0	0002-11-30 (2020-11-08	金立		0	0					
4	1003959	1000000003145001003959	华为Meta1003959	913959	100	20	http://www.chai	0	0002-11-30 (2020-11-08	金立		0	0					
5	1002353	1000000003145001002353	华为Meta1002353	912353	100	20	http://www.chai	0	0002-11-30 (2020-04-12	华为		0	0					
6	1002351	1000000003145001002351	华为Meta1002351	912351	100	20	http://www.chai	0	0002-11-30 (2020-04-12	华为		0	0					
7	1001837	1000000003145001001837	华为Meta1001837	911837	100	20	http://www.chai	0	0002-11-30 (2020-02-15	华为		0	0					
8	1001835	1000000003145001001835	华为Meta1001835	911835	100	20	http://www.chai	0	0002-11-30 (2020-02-15	华为		0	0					
9	1001833	1000000003145001001833	华为Meta1001833	911833	100	20	http://www.chai	0	0002-11-30 (2020-02-15	华为		0	0					
10	1003964	1000000003145001003964	华为Meta1003964	613964	100	20	http://www.chai	0	0002-11-30 (2020-11-08	金立		0	0					
11	1003962	1000000003145001003962	华为Meta1003962	613962	100	20	http://www.chai	0	0002-11-30 (2020-11-08	金立		0	0					
12	1003960	1000000003145001003960	华为Meta1003960	613960	100	20	http://www.chai	0	0002-11-30 (2020-11-08	金立		0	0					
13	1002354	1000000003145001002354	华为Meta1002354	612354	100	20	http://www.chai	0	0002-11-30 (2020-04-12	华为		0	0					
14	1002352	1000000003145001002352	华为Meta1002352	612352	100	20	http://www.chai	0	0002-11-30 (2020-04-12	华为		0	0					
15	1001838	1000000003145001001838	华为Meta1001838	611838	100	20	http://www.chai	0	0002-11-30 (2020-02-15	华为		0	0					

**实现提示：**

- 1、读取excel表格中的数据到List集合中代码已经实现
- 2、将一个List集合中的数据写入到一个excel文件中代码已经实现
- 3、在Entry类中指定位置补全相关代码
- 4、在Sku类中的指定位置补全代码
- 5、如果商品的条码是相同的则认为是同一条数据

### 2.2 编程题目二

**训练目标：**掌握JDK1.8的新特性Stream流的基本使用，以及理解其在实际开发中的应用

**需求背景：**在完成数据查询功能的时候，如果查询到的满足条件的数据量比较大，为了减少系统内存的消耗。那么此时就需要使用分页查询。分页查询就是一次查询一部分的数据进行返回。比如

京东商城的搜索功能实现的时候使用的就是分页查询。如下所示：



默认查询的是第一页的数据，当点击对应的页码的时候就会查询指定页码的这一部分的数据。分页查询需要已知两个参数：页码，每页查询的数据条数。通过这两个数据就可以计算出要查询的数据

的开始行号。进而就可以实现分页查询。

**需求描述：**将编程题目一中处理完毕的以后的数据集中的第3页数据写入到新的excel中。在素材的stream02项目的指定位置补全相关代码，新的excel文件的内容如下所示：

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	商品编号	商品条码	商品名称	商品价格	库存数量	库存预警数量	商品图片	重量	创建时间	更新时间	规格	销量	商品状态				
2	1009049	100000003145001009049	华为Meta1009049	118849	10000	100	https://m.360buy	10	2019-05-01 00	2019-05-01 00:0	白色10090	0	0				
3	1009981	100000003145001009981	华为Meta1009981	118681	10000	100	https://m.360buy	10	2019-05-01 00	2019-05-01 00:0	白色10099	0	0				
4	1008924	100000003145001008924	华为Meta1008924	118524	10000	100	https://m.360buy	10	2019-05-01 00	2019-05-01 00:0	白色10089	0	0				
5	1008887	100000003145001008887	华为Meta1008887	118487	10000	100	https://m.360buy	10	2019-05-01 00	2019-05-01 00:0	白色10088	0	0				
6	1008895	100000003145001008895	华为Meta1008895	118395	10000	100	https://m.360buy	10	2019-05-01 00	2019-05-01 00:0	白色10088	0	0				
7	1009651	100000003145001009651	华为Meta1009651	118351	10000	100	https://m.360buy	10	2019-05-01 00	2019-05-01 00:0	白色10096	0	0	共10条数据			
8	1008822	100000003145001008822	华为Meta1008822	118222	10000	100	https://m.360buy	10	2019-05-01 00	2019-05-01 00:0	白色10088	0	0				
9	1009115	100000003145001009115	华为Meta1009115	118215	10000	100	https://m.360buy	10	2019-05-01 00	2019-05-01 00:0	白色10091	0	0				
10	1009674	100000003145001009674	华为Meta1009674	118174	10000	100	https://m.360buy	10	2019-05-01 00	2019-05-01 00:0	白色10096	0	0				
11	1009066	100000003145001009066	华为Meta1009066	118166	10000	100	https://m.360buy	10	2019-05-01 00	2019-05-01 00:0	白色10090	0	0				

**实现提示：**

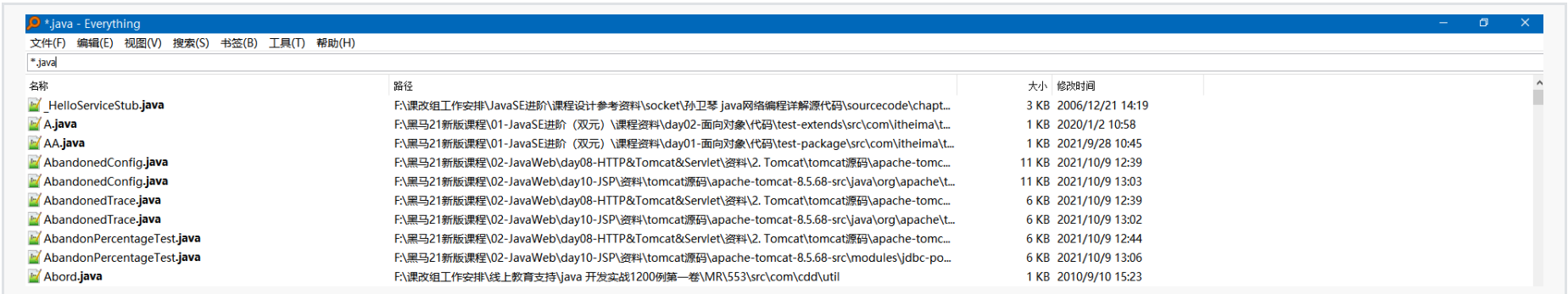
- 1、读取excel表格中的数据到List集合中代码已经实现
- 2、将一个List集合中的数据写入到一个excel文件中代码已经实现
- 3、在Entry类中指定位置补全相关代码
- 4、在Sku类中的指定位置补全代码
- 5、如果商品的条码是相同的则认为是同一条数据
- 6、考虑使用stream流中的skip方法和limit方法(每页查询10条数据)

## 2.3 编程题目三

**训练目标：**掌握java中File的基本使用，以及理解其在实际开发中的应用

**需求背景：**当电脑磁盘上存储的文件越来越多的时候，要从系统中查找某一个文件就变的比较麻烦。为了快速从磁盘上找的某一类文件，此时就可以借助于类似于Everything这样的文件搜索工具

实现文件的搜索。如下所示，就是借助于Everything搜索本地磁盘上所有以.java结尾的文件的效果：



**需求描述：**现通过程序模拟Everything搜索工具搜索效果，在素材everthing项目中的指定位置补全代码实现搜索功能。案例的实现效果如下所示：

名称	路径	大小(字节)	最后一次修改时间
AdminServerApplication.java	E:\idea-edu-workplace\everthing\tanhua-admin\src\main\java\com\tanhua\admin\AdminServerApplication.java	779	2021/06/30 11:47:57
ManagerController.java	E:\idea-edu-workplace\everthing\tanhua-admin\src\main\java\com\tanhua\admin\controller\ManagerController.java	2403	2021/06/30 09:22:05
SystemController.java	E:\idea-edu-workplace\everthing\tanhua-admin\src\main\java\com\tanhua\admin\controller\SystemController.java	1804	2021/06/30 09:33:27
BusinessException.java	E:\idea-edu-workplace\everthing\tanhua-admin\src\main\java\com\tanhua\admin\exception\BusinessException.java	259	2021/06/29 21:52:22
ExceptionAdvice.java	E:\idea-edu-workplace\everthing\tanhua-admin\src\main\java\com\tanhua\admin\exception\ExceptionAdvice.java	795	2021/06/26 11:46:48
AdminHolder.java	E:\idea-edu-workplace\everthing\tanhua-admin\src\main\java\com\tanhua\admin\interceptor\AdminHolder.java	660	2021/06/29 21:52:22
TokenInterceptor.java	E:\idea-edu-workplace\everthing\tanhua-admin\src\main\java\com\tanhua\admin\interceptor\TokenInterceptor.java	1308	2021/06/29 08:59:15
WebConfig.java	E:\idea-edu-workplace\everthing\tanhua-admin\src\main\java\com\tanhua\admin\interceptor\WebConfig.java	820	2021/06/26 11:46:48
AuditListener.java	E:\idea-edu-workplace\everthing\tanhua-admin\src\main\java\com\tanhua\admin\listener\AuditListener.java	2683	2022/01/12 16:51:14
LogListener.java	E:\idea-edu-workplace\everthing\tanhua-admin\src\main\java\com\tanhua\admin\listener\LogListener.java	1865	2022/01/12 14:16:06
AdminMapper.java	E:\idea-edu-workplace\everthing\tanhua-admin\src\main\java\com\tanhua\admin\mapper\AdminMapper.java	197	2021/06/29 21:52:22
AnalysisMapper.java	E:\idea-edu-workplace\everthing\tanhua-admin\src\main\java\com\tanhua\admin\mapper\AnalysisMapper.java	206	2021/06/30 11:29:18
LogMapper.java	E:\idea-edu-workplace\everthing\tanhua-admin\src\main\java\com\tanhua\admin\mapper\LogMapper.java	1012	2022/01/12 15:43:25
AdminService.java	E:\idea-edu-workplace\everthing\tanhua-admin\src\main\java\com\tanhua\admin\service\AdminService.java	2566	2021/06/29 21:52:22
AnalysisService.java	E:\idea-edu-workplace\everthing\tanhua-admin\src\main\java\com\tanhua\admin\service\AnalysisService.java	2780	2021/10/23 15:03:00
ManagerService.java	E:\idea-edu-workplace\everthing\tanhua-admin\src\main\java\com\tanhua\admin\service\ManagerService.java	4760	2022/01/12 09:44:00
AnalysisTask.java	E:\idea-edu-workplace\everthing\tanhua-admin\src\main\java\com\tanhua\admin\task\AnalysisTask.java	812	2021/10/23 16:34:01
GreenTemplateTest.java	E:\idea-edu-workplace\everthing\tanhua-admin\src\test\java\com\tanhua\admin\GreenTemplateTest.java	1067	2021/06/30 13:42:19
LogTest.java	E:\idea-edu-workplace\everthing\tanhua-admin\src\test\java\com\tanhua\admin\LogTest.java	2207	2022/01/12 16:09:27

实现提示：

- 1、需要使用到递归算法
- 2、控制台输出内容对齐可以调用padRight方法通过补全字符的方式进行实现
- 3、使用SimpleDateFormat对日期数据进行格式化