**每日作业卷答案**

**就业班JavaSE第13天**

传智播客.黑马程序员

**基础题**

### 练习一：Pedicate接口使用

1. 请在测试类main方法中完成以下需求

已知有Integer[] arr = {-12345, 9999, 520, 0,-38,-7758520,941213}

1. 使用lambda表达式创建Predicate对象p1,p1能判断整数是否是自然数(大于等于0)
2. 使用lambda表达式创建Predicate对象p2,p2能判断整数的绝对值是否大于100
3. 使用lambda表达式创建Predicate对象p3,p3能判断整数是否是偶数

遍历arr，仅利用已创建的Predicate对象(不使用任何逻辑运算符)，完成以下需求

1. 打印自然数的个数
2. 打印负整数的个数
3. 打印绝对值大于100的偶数的个数
4. 打印是负整数或偶数的数的个数

答案

import java.util.function.Predicate;  
public class Test03 {  
 public static void main(String[] args) {  
 Integer[] arr = {-12345, 9999, 520, 0,-38,-7758520,941213};  
  
 //a) 使用lambda表达式创建Predicate对象p1,p1能判断整数是否是自然数  
 Predicate<Integer> p1 = (s) -> s>=0;  
 //b) 使用lambda表达式创建Predicate对象p2,p2能判断整数的绝对值是否大于100  
 Predicate<Integer> p2 = (s) -> Math.abs(s)>100;  
 //c) 使用lambda表达式创建Predicate对象p3,p3能判断整数是否是偶数  
 Predicate<Integer> p3 = (s) -> s%2==0;  
  
 //e) 遍历arr，仅利用已创建的Predicate对象(不使用任何逻辑运算符)，完成以下需求  
 int count1 = 0;  
 int count2 = 0;  
 int count3 = 0;  
 int count4 = 0;  
 for (Integer i : arr) {  
 //统计自然数个数  
 if (p1.test(i)){  
 count1++;  
 }  
 //统计负整数个数  
 if (p1.negate().test(i)){  
 count2++;  
 }  
 //统计绝对值大于100的偶数个数  
 if (p2.and(p3).test(i)){  
 count3++;  
 }  
 //统计是负整数或偶数的数的个数  
 if (p1.negate().or(p3).test(i)){  
 count4++;  
 }  
 }  
 //分别打印结果  
 System.out.println("自然数的个数为："+count1);  
 System.out.println("负整数的个数为："+count2);  
 System.out.println("绝对值大于100的偶数的个数为："+count3);  
 System.out.println("是负整数或偶数的数的个数为："+count4);  
 }  
}

### 练习二：Function接口使用

1. 使用lambda表达式分别将以下功能封装到Function对象中
2. 求Integer类型ArrayList中所有元素的平均数
3. 将Map<String,Integer>中value存到ArrayList<Integer>中
4. 已知学生成绩如下

|  |  |
| --- | --- |
| 姓名 | 成绩 |
| 岑小村 | 59 |
| 谷天洛 | 82 |
| 渣渣辉 | 98 |
| 蓝小月 | 65 |
| 皮几万 | 70 |

1. 以学生姓名为key成绩为value创建集合并存储数据，使用刚刚创建的Function对象求学生的平均成绩

答案

import java.util.\*;  
import java.util.function.Function;  
public class Test04 {  
 public static void main(String[] args) {  
 //1. 使用lambda表达式分别将以下功能封装到Function对象中  
 //a) 求Integer类型ArrayList中所有元素的平均数  
 Function<ArrayList<Integer>,Integer> f1 = (list)->{  
 Integer sum = 0;  
 for (Integer i : list) {  
 sum+=i;  
 }  
 return sum/list.size();  
 };  
  
 //b) 将Map<String,Integer>中value存到ArrayList<Integer>中  
 Function<Map<String,Integer>,ArrayList<Integer>> f2 = (map)->{  
 /\*ArrayList<Integer> list = new ArrayList<>();  
 for (String s : map.keySet()) {  
 Integer i = map.get(s);  
 list.add(i);  
 }\*/  
 Collection<Integer> values = map.values();  
 ArrayList<Integer> list = new ArrayList<>();  
 list.addAll(values);  
 return list;  
 };  
   
 //2 将学生姓名和成绩封装到map中  
 Map<String,Integer> map = new HashMap<String, Integer>();  
 map.put("岑小村", 59);  
 map.put("谷天洛", 82);  
 map.put("渣渣辉", 98);  
 map.put("蓝小月", 65);  
 map.put("皮几万", 70);  
  
 //利用Function求平均成绩  
 Integer avg = f2.andThen(f1).apply(map);  
 System.out.println("学生平均成绩为："+avg);  
 }  
}

### 练习三：如何获取流

问题：

简述单列集合、双列集合、数组分别如何获取Stream流对象，并进行演示

答：

1、java.util.Collection接口中加入了default方法stream()获取流对象，因此其所有实现类均可通过此方式获取流。

2、java.util.Map接口想要获取流，先通过keySet()、values()或entrySet()方法获取键、值或键值对的单列集合，再通过stream()获取流对象。

3、数组获取流，使用Stream接口中的的静态方法of(T...values)获取流。

public static void main(String[] args) {  
 List<String> list = new ArrayList<>();  
 Stream<String> stream1 = list.stream();  
  
 Set<String> set = new HashSet<>();  
 Stream<String> stream2 = set.stream();  
  
 Map<String, String> map = new HashMap<>();  
 Stream<String> keyStream = map.keySet().stream();  
 Stream<String> valueStream = map.values().stream();  
 Stream<Map.Entry<String,String>>entryStream = map.entrySet().stream();  
  
 String[] array = {"东邪", "西毒", "南帝", "北丐", "中神通"};  
 Stream<String> stream = Stream.of(array);  
 }

### 练习四：过滤：filter、结果收集(数组)

问题：

有如下7个元素黄药师，冯蘅，郭靖，黄蓉，郭芙，郭襄，郭破虏，使用Stream将以郭字开头的元素存入新数组

答：

import java.util.stream.Stream;  
public class Test {  
 public static void main(String[] args) {  
 Stream<String> stream = Stream.of("黄药师", "冯蘅", "郭靖", "黄蓉", "郭芙", "郭襄", "郭破虏");  
 String[] guos = stream.filter(s -> s.startsWith("郭")).toArray(String[]::new);  
 }  
}

### 练习五：取用前几个：limit、跳过前几个：skip

问题：

已知ArrayList集合中有如下元素{陈玄风、梅超风、陆乘风、曲灵风、武眠风、冯默风、罗玉风}，使用Stream

1、取出前2个元素并在控制台打印输出。

2、取出后2个元素并在控制台打印输出。

答：

import java.util.ArrayList;  
public class Test04 {  
 public static void main(String[] args) {  
 ArrayList<String> list = new ArrayList<>();  
 list.add("陈玄风");  
 list.add("梅超风");  
 list.add("陆乘风");  
 list.add("曲灵风");  
 list.add("武眠风");  
 list.add("冯默风");  
 list.add("罗玉风");

list.stream().limit(2).forEach(System.out::println);  
 list.stream().skip(list.size() - 2).forEach(System.out::println);  
 }  
}

### 练习六：映射：map、逐一消费：forEach

问题：

有如下整数1，-2，-3，4，-5

使用Stream取元素绝对值并打印

答：

import java.util.stream.Stream;  
public class Test {  
 public static void main(String[] args) {  
 Stream<Integer> stream = Stream.of(1, -2, -3, 4,-5);  
 stream.map(Math::abs).forEach(System.out::println);  
 }  
}

### 练习七：组合：concat、结果收集(list)

问题：

已知数组arr1中有如下元素{郭靖，杨康}，arr2中有如下元素{黄蓉，穆念慈}，使用Stream将二者合并到List集合

答：

import java.util.stream.Stream;  
public class Test {  
 public static void main(String[] args) {  
 Stream<String> streamA = Stream.of("郭靖", "杨康");  
 Stream<String> streamB = Stream.of("黄蓉", "穆念慈");  
 List<String> strList = Stream.concat(streamA, streamB).collect(Collectors.toList());  
 }  
}

### 练习八：获取并发流

问题：

请分别写出获取并发流的两种方式。

答：

public class Test {  
 public static void main(String[] args) {  
 Collection<String> coll = new ArrayList<>();  
 Stream<String> parallelStream1 = coll.parallelStream();

Stream<Integer> parallelStream2 = Stream.of(100, 200, 300, 400).parallel();  
 }  
}

## 扩展题

### 练习九：Stream综合练习

问题：

现在有两个 ArrayList 集合存储队伍当中的多个成员姓名，要求使用Stream方式进行以

下若干操作步骤：

1. 第一个队伍只要名字为3个字的成员姓名；

2. 第一个队伍筛选之后只要前6个人；

3. 第二个队伍只要姓张的成员姓名；

4. 第二个队伍筛选之后不要前1个人；

5. 将两个队伍合并为一个队伍；

6. 根据姓名创建Student对象；

7. 打印整个队伍的Student对象信息。

两个队伍（集合）的代码如下：

|  |
| --- |
| **import** java.util.ArrayList; **import** java.util.List; **import** java.util.stream.Stream;  **public class** Test { **public static void** main(String[] args) {  List<String> one = **new** ArrayList<>();  one.add(**"陈玄风"**);  one.add(**"梅超风"**);  one.add(**"陆乘风"**);  one.add(**"曲灵风"**);  one.add(**"武眠风"**);  one.add(**"冯默风"**);  one.add(**"罗玉风"**);  List<String> two = **new** ArrayList<>();  two.add(**"宋远桥"**);  two.add(**"俞莲舟"**);  two.add(**"俞岱岩"**);  two.add(**"张松溪"**);  two.add(**"张翠山"**);  two.add(**"殷梨亭"**);  two.add(**"莫声谷"**);  } } |

而Student类的代码为：

|  |
| --- |
| **public class** Student { **private** String **name**;  **public** Student() {  }  **public** Student(String name) { **this**.**name** = name;  }  @Override **public** String toString() { **return " Student {name='"** + **name** + **"'}"**;  }  **public** String getName() { **return name**;  }  **public void** setName(String name) { **this**.**name** = name;  }  } |

答：

import java.util.ArrayList;  
import java.util.List;  
import java.util.stream.Stream;  
public class Test {  
 public static void main(String[] args) {  
 List<String> one = new ArrayList<>();  
 one.add("陈玄风");  
 one.add("梅超风");  
 one.add("陆乘风");  
 one.add("曲灵风");  
 one.add("武眠风");  
 one.add("冯默风");  
 one.add("罗玉风");  
 List<String> two = new ArrayList<>();  
 two.add("宋远桥");  
 two.add("俞莲舟");  
 two.add("俞岱岩");  
 two.add("张松溪");  
 two.add("张翠山");  
 two.add("殷梨亭");  
 two.add("莫声谷");  
  
 // 第一个队伍只要名字为3个字的成员姓名；  
 // 第一个队伍筛选之后只要前6个人；  
 Stream<String> streamOne = one.stream().filter(s -> s.length() == 3).limit(6);  
  
 // 第二个队伍只要姓张的成员姓名；  
 // 第二个队伍筛选之后不要前1个人；  
 Stream<String> streamTwo = two.stream().filter(s -> s.startsWith("张")).skip(1);  
  
 // 将两个队伍合并为一个队伍；  
 // 根据姓名创建Student对象；  
 // 打印整个队伍的Student对象信息。

Stream.concat(streamOne,streamTwo).map(Student::new).forEach(System.out::println);  
 }  
}

### 练习十：Stream综合练习

问题：

以下是某不知名机构评出的全球最佳影片及华人最佳影片前十名 ：

全球  
　　1、 《教父》  
　　2、 《肖申克的救赎》  
　　3、 《辛德勒的名单》  
　　4、 《公民凯恩》  
　　5、 《卡萨布兰卡》   
　　6、 《教父续集》  
　　7、 《七武士》

　　8、 《星球大战》  
　　9、 《美国美人》

　　10、 《飞跃疯人院》

华人

　 1、 《霸王别姬》  
　　2、 《大闹天宫》  
　　3、 《鬼子来了》  
　　4、 《大话西游》  
　　5、 《活着》  
　　6、 《饮食男女》  
　　7、 《无间道》

　　8、 《天书奇谭》  
　　9、 《哪吒脑海》

　　10、 《春光乍泄》

1、现将两个榜单中的影片名，分别按排名顺序依次存入两个ArrayList集合

2、通过流的方式

1）打印全球影片排行榜中的前三甲影片名

2）打印华人影片排行榜中倒数5名的影片名

3）将两个排行榜中的前5名挑出来共同存入新的集合

4）定义电影Film类，以影片名为name创建Film对象并保存至集合

答：

import java.util.ArrayList;  
import java.util.List;  
import java.util.stream.Collectors;  
import java.util.stream.Stream;  
public class Test03 {  
 public static void main(String[] args) {  
 //将原始数据存入集合  
 ArrayList<String> global = new ArrayList<>();  
 global.add("《教父》");  
 global.add("《肖申克的救赎》");  
 global.add("《辛德勒的名单》");  
 global.add("《公民凯恩》");  
 global.add("《卡萨布兰卡》");  
 global.add("《教父续集》");  
 global.add("《七武士》");  
 global.add("《星球大战》");  
 global.add("《美国美人》");  
 global.add("《飞跃疯人院》");  
 ArrayList<String> china = new ArrayList<>();  
 china.add("《霸王别姬》");  
 china.add("《大闹天宫》");  
 china.add("《鬼子来了》");  
 china.add("《大话西游》");  
 china.add("《活着》");  
 china.add("《饮食男女》");  
 china.add("《无间道》");  
 china.add("《天书奇谭》");  
 china.add("《哪吒脑海》");  
 china.add("《春光乍泄》");  
  
 //1）打印全球影片排行榜中的前三甲影片名  
 global.stream().limit(3).forEach(System.out::println);  
  
 //2）打印华人影片排行榜中倒数5名的影片名  
 china.stream().skip(china.size()-5).forEach(System.out::println);  
  
 //3）将两个排行榜中的前5名挑出来共同存入新的集合  
 List<String> list = Stream.concat(global.stream().limit(5), china.stream().limit(5)).collect(Collectors.toList());  
  
 //4）将所有影片以影片名为name创建Film对象并保存至集合  
 List<Film> filmList = Stream.concat(global.stream(), china.stream()).map(Film::new).collect(Collectors.toList());  
 }  
}

public class Film {  
 private String name;  
 public Film() {  
 }  
 public Film(String name) {  
 this.name = name;  
 }  
 public String getName() {  
 return name;  
 }  
 public void setName(String name) {  
 this.name = name;  
 }  
}

### 练习十一：Stream综合练习

问题：

我国有34个省级行政区，分别是：

23个省：

河北省、山西省、吉林省、辽宁省、[黑龙江省](https://www.baidu.com/s?wd=%E9%BB%91%E9%BE%99%E6%B1%9F%E7%9C%81&from=1012015a&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y4rHRLuWnvP16dnj79uW040ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnWmvrjmsrjfv" \t "_blank)、陕西省、甘肃省、青海省、山东省、福建省、

浙江省、台湾省、河南省、湖北省、湖南省、江西省、江苏省、安徽省、广东省、海南省、四川省、贵州省、云南省。

4个直辖市：

北京市、天津市、上海市、重庆市。

5个自治区：

[内蒙古自治区](https://www.baidu.com/s?wd=%E5%86%85%E8%92%99%E5%8F%A4%E8%87%AA%E6%B2%BB%E5%8C%BA&from=1012015a&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y4rHRLuWnvP16dnj79uW040ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnWmvrjmsrjfv" \t "_blank)、[新疆维吾尔自治区](https://www.baidu.com/s?wd=%E6%96%B0%E7%96%86%E7%BB%B4%E5%90%BE%E5%B0%94%E8%87%AA%E6%B2%BB%E5%8C%BA&from=1012015a&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y4rHRLuWnvP16dnj79uW040ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnWmvrjmsrjfv" \t "_blank)、[宁夏回族自治区](https://www.baidu.com/s?wd=%E5%AE%81%E5%A4%8F%E5%9B%9E%E6%97%8F%E8%87%AA%E6%B2%BB%E5%8C%BA&from=1012015a&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y4rHRLuWnvP16dnj79uW040ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnWmvrjmsrjfv" \t "_blank)、[广西壮族自治区](https://www.baidu.com/s?wd=%E5%B9%BF%E8%A5%BF%E5%A3%AE%E6%97%8F%E8%87%AA%E6%B2%BB%E5%8C%BA&from=1012015a&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y4rHRLuWnvP16dnj79uW040ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnWmvrjmsrjfv" \t "_blank)、[西藏自治区](https://www.baidu.com/s?wd=%E8%A5%BF%E8%97%8F%E8%87%AA%E6%B2%BB%E5%8C%BA&from=1012015a&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y4rHRLuWnvP16dnj79uW040ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnWmvrjmsrjfv" \t "_blank)。

2个特别行政区：

[香港特别行政区](https://www.baidu.com/s?wd=%E9%A6%99%E6%B8%AF%E7%89%B9%E5%88%AB%E8%A1%8C%E6%94%BF%E5%8C%BA&from=1012015a&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y4rHRLuWnvP16dnj79uW040ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnWmvrjmsrjfv" \t "_blank)、[澳门特别行政区](https://www.baidu.com/s?wd=%E6%BE%B3%E9%97%A8%E7%89%B9%E5%88%AB%E8%A1%8C%E6%94%BF%E5%8C%BA&from=1012015a&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y4rHRLuWnvP16dnj79uW040ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnWmvrjmsrjfv" \t "_blank)。

使用流：

1、统计三个字的省份的个数

2、统计名字中包含方位名词的省份（东西南北）的个数

3、打印名字中包含方位名词的普通省份（非自治区直辖市特别行政区）的名字

4、将所有的特殊省份（自治区直辖市特别行政区）提取出来并放到新数组中

答：

import java.util.Arrays;  
import java.util.stream.Stream;  
public class Test02 {   
 public static void main(String[] args) {  
 String[] provinces = {"河北省", "山西省", "吉林省", "辽宁省",  
 "黑龙江省", "陕西省", "甘肃省", "青海省", "山东省", "福建省", "浙江省",  
 "台湾省", "河南省", "湖北省", "湖南省", "江西省", "江苏省", "安徽省",  
 "广东省", "海南省", "四川省", "贵州省", "云南省", "北京市", "天津市",  
 "上海市", "重庆市", "内蒙古自治区", "新疆维吾尔自治区", "宁夏回族自治区",  
 "广西壮族自治区", "西藏自治区", "香港特别行政区", "澳门特别行政区"};  
  
 //1、统计三个字的省份的个数  
 long threeCount = Stream.of(provinces).filter(s -> s.length() == 3).count();  
 System.out.println("三个字的省份的个数："+threeCount);  
  
 //2、统计名字中包含方位名词的省份（东西南北）的个数  
 long count = Stream.of(provinces).filter(s -> s.contains("东") || s.contains("西") || s.contains("南") || s.contains("北")).count();  
 System.out.println("包含方位名词的省份（东西南北）的个数:"+count);  
  
 //3、打印名字中包含方位名词的普通省份（非自治区直辖市特别行政区）的名字  
 System.out.println("包含方位名词的普通省份有：");  
 Stream.of(provinces).filter(s -> s.contains("东") || s.contains("西") || s.contains("南") || s.contains("北")).filter(s->s.contains("省")).forEach(System.out::println);  
  
 //4、将所有的特殊省份（自治区直辖市特别行政区）提取出来并放到新数组中  
 String[] pros = Stream.of(provinces).filter(s -> !s.contains("省")).toArray(String[]::new);  
 System.out.println("新数组:"+Arrays.toString(pros));  
 }  
}