

day07作业

一、选择题：

1. 实现下列哪个接口，可以启用比较功能(**D**)

- A. Runnable接口
- B. Iterator接口
- C. Serializable接口
- D. Comparator接口

2. Set集合的特点是 (**B**)

- A. 元素有序
- B. 元素无序，不存储重复元素，没有索引
- C. 存储重复元素
- D. Set集合都是线程安全的

3. 使用**TreeSet**的无参构造创建集合对象存储元素时，该元素必须 (**A**)

- A. 必须实现Comparable接口
- B. 有main方法
- C. 有get和set方法
- D. 必须实现Serializable接口

4. 【多选题】下面的代码用于输出字符数组**ch**中每个字符出现的次数，应该填入位置①的代码是，不允许使用工具运行程序()

```

public class Demo1{
    public static void main(String[] args) {
        char[] ch = { 'a', 'c', 'a', 'b', 'c', 'b' };
        Map map = new HashMap();
        for (int i = 0; i < ch.length; i++) {
            //位置①
        }
        System.out.println(map);
    }
}

```

A.

```

if (map.contains(ch[i])) {
    map.put(ch[i], map.get(ch[i]) + 1);
} else {
    map.put(ch[i], 1);
}

```

B.

```

if (map.contains(ch[i])) {
    map.put(ch[i], (Integer) map.get(ch[i]) + 1);
} else {
    map.put(ch[i], 1);
}

```

C.

```

if (map.containsKey(ch[i])) {
    map.put(ch[i], (int) map.get(ch[i]) + 1);
} else {
    map.put(ch[i], 1);
}

```

D.

```

if (map.containsKey(ch[i])) {
    map.put(ch[i], (Integer) map.get(ch[i]) + 1);
} else {
    map.put(ch[i], 1);
}

```

5. 【多选题】按照课堂要求重写**equals**和**hashCode**后，下面关于这两个方法说法正确的是(BC)

A. 两个对象的hashCode值相同，那么他们调用equals()方法返回值一定为true

- B. 两个对象的hashCode值相同，那么他们调用equals()方法返回值可以为false
- C. hashCode值一般与对象的成员变量有关。
- D. 只要重写equals方法，就一定要重写hashCode方法

6. 题示代码的功能为:循环遍历输出**Map**当中的每一个元素，下列每个选项中分别填入上面三个位置，正确的是（**D**）

```
public class Demol{  
    public static void main(String[] args){  
        Map map = new HashMap();  
        map.put("jessica", 100);  
        map.put("tom", 200);  
        map.put("den", 300);  
        Set 位置① set = 位置②;  
        for (位置③ per : set){  
            System.out.println(per.getKey() + ":" + per.getValue());  
        }  
    }  
}
```

- A.
 <Entry>
 map.keySet()
 Entry
- B.
 <Entry<Integer, String>>
 map.entrySet()
 Entry
- C.
 <Map.Entry<String, Integer>>
 map.keySet()
 Map.Entry
- D.
 <Map.Entry<String, Integer>>
 map.entrySet()
 Map.Entry

7. 下列代码的运行结果是（）

```
HashMap<Integer, Integer> map = new HashMap<>();  
map.put(1, "小明");  
map.put(1, "小红");  
map.put(2, "小刚");  
System.out.println(map.size());
```

- A. 1
- B. 2
- C. 3
- D. 程序编译时报错

8. 以下哪段代码能实现元素存入不重复且元素按照首字母降序输出（C）

```

A、 HashSet<String> set = new HashSet<String>(new Comparator<String>() {
    @Override
    public int compare(String o1, String o2) {
        return o2.charAt(0) - o1.charAt(0);
    }
});
Collections.put(set, "cc", "aa", "bb", "dd");

B、 HashSet<String> set = new HashSet<String>(new Comparator<String>() {
    @Override
    public int compare(String o1, String o2) {
        return o1.charAt(0)-o2.charAt(0);
    }
});
Collections.add(set, "cc", "aa", "bb", "dd");

C、 TreeSet<String> ts = new TreeSet<String>(new Comparator<String>() {
    @Override
    public int compare(String o1, String o2) {
        return o2.charAt(0)-o1.charAt(0);
    }
});
Collections.addAll(ts, "cc", "aa", "bb", "dd");

D、 TreeSet<String> hs = new TreeSet<String>(new Comparator<String>() {
    @Override
    public int compare(String o1, String o2) {
        return o1.charAt(0)-o2.charAt(0);
    }
});
Collections.addAll(hs, "cc", "aa", "bb", "dd");

```

9. 【多选题】下列遍历map集合中各元素的键和值正确的是（AB）

```
A、 Set<Integer> keys = map.keySet();
for (Integer key : keys) {
    System.out.println(key + ":" + map.get(key));
}
B、 Set<Map.Entry<Integer,String>> entrySet = map.entrySet();
for (Map.Entry<Integer, String> entry : entrySet) {
    System.out.println(entry.getKey() + ":" + entry.getValue());
}
C、 Set<Map.Entry<Integer,String>> entrySet = map.entrySet();
for (int i = entrySet.size() - 1; i >= 0; i--) {
    System.out.println(entrySet);
}
D、 Set<Integer> keys = map.keySet();
for (int i = keys.size() - 1; i >= 0; i--) {
    System.out.println(i);
}
```

10. 【多先吃】下列能正确遍历**Map**集合的是（**BD**）

```

public class Demo1{
    public static void main(String[] args){
        HashMap<String, String> hashMap = new HashMap<>();
        hashMap.put("张无忌","赵敏");
        hashMap.put("杨过","小龙女");
        hashMap.put("郭靖","黄蓉");
        hashMap.put("令狐冲","任盈盈");
        // 遍历
    }
}
A、 for (int i = 0; i < hashMap.size(); i++) {
    String s = hashMap.get(i);
    System.out.println(s);
}
B、 Set<String> keySet = hashMap.keySet();
    for (String key : keySet) {
        String value = hashMap.get(key);
        System.out.println(key + "=" + value);
    }
C、 for (String s:hashMap){
    String s1 = hashMap.get(s);
    System.out.println(s1);
}
D、 Set<Map.Entry<String, String>> entrySet = hashMap.entrySet();
    for (Map.Entry<String, String> entry : entrySet) {
        String key = entry.getKey();
        String value = entry.getValue();
        System.out.println(key + "==" + value);
    }
}

```

11. 下列关于可变参数说法中，正确的是（C）

- A. 可变参数可以在参数列表的任何位置
- B. 一个方法中允许包含多个可变参数
- C. 可变参数的本质是一个数组
- D. 调用一个包含可变参数的方法时，只能传入多个参数，不能传入数组。

12. 关于可变参数的说法正确的是（ACD）

- A.可变参数的格式为[数据类型 ...变量名]
 - B.可变参数不能打印
 - C.直接打印可变参数,得到的是一个地址值
 - D.可变参数可以传递任意多个同类型对象
-

二、简答题:

1. Set集合有什么特点? TreeSet、HashSet分别有何特点?

答:

2. HashSet底层数据结构是什么? 请简述?

答:

3. LinkedHashSet和HashSet有何区别?

答:

4. 可变参数本质是什么? 使用时需要注意什么?

答:

5. TreeSet集合中两个比较器的区别? 什么时候使用?

答:

6. Map集合有什么特点? TreeMap、HashMap分别有何特点?

答:

7. Map集合能直接遍历吗? 有几种遍历方式?

答:

三、排错题:

排错题1:


```

public class Student {    // 定义学生类

    private int age ;    // 定义age成员变量
    private String name ;    // 定义name成员变量

    public Student(int age, String name) {
        this.age = age;
        this.name = name;
    }

    // 省略get和set方法

    @Override
    public String toString() {
        return "Student{" + "age=" + age + ", name=" + name + '}';
    }
}

class TreeSetDemo01 {

    public static void main(String[] args) {

        // 创建TreeSet集合对象
        TreeSet<Student> students = new TreeSet<>();

        // 添加元素
        students.add(new Student( age: 16 , name: "itheima")) ;
        students.add(new Student( age: 16 , name: "itcast")) ;
        students.add(new Student( age: 16 , name: "传智播客")) ;

        // 遍历集合
        students.forEach( s -> System.out.println(s) );

    }

}

```

4 2 ^

上述程序执行完毕以后会在控制台输出如下错误内容：

```

C:\develop\Java\jdk1.8.0_172\bin\java.exe ...
Exception in thread "main" java.lang.ClassCastException: Create breakpoint : com.itheima.set.demo01.Student cannot be cast to java.lang.Comparable
    at java.util.TreeMap.compare(TreeMap.java:1294)
    at java.util.TreeMap.put(TreeMap.java:538)
    at java.util.TreeSet.add(TreeSet.java:255)
    at com.itheima.set.demo01.TreeSetDemo01.main(Student.java:29)

```

请说明产生该问题的原因？如果想先按照学生的年龄进行从大到小排序，如果年龄相同再按照学生的姓名进行排序，那么可以对代码进行怎样的改造？

答：

排错题2：

```

class Person {
    private int idCard;
    private String userName;
    // 以下是getter和setter/toString/构造方法，省略
    public boolean equals(Object o) {
        if (this == o) return true;
        if (o == null || getClass() != o.getClass()) return false;
        Person person = (Person) o;
        return idCard == person.idCard && Objects.equals(userName,
person.userName);
    }
    public int hashCode() {
        return 0;
    }
}
public class Demo1 {
    public static void main(String[] args) {
        Set<Person> set = new HashSet<>();
        set.add(new Person(154,"zhaoliu"));
        set.add(new Person(101,"zhangsan"));
        set.add(new Person(101,"zhangsan"));
        set.add(new Person(133,"wangwu"));
        set.add(new Person(115,"zhangsan"));
        set.add(new Person(115,"zhangsan"));
        set.add(new Person(112,"lisi"));
        System.out.println(set.size());
        // set集合中，最终会有几个元素？为什么？
    }
}

```

答：

排错题3:

// 要求：以下代码针对Person-idCard排序(降序)，请问代码是否会报错？如果有错？如何解决？

```
class Person {
    private int idCard;
    private String userName;
    // 以下是getter和setter方法
    // 省略
}

public class Demo4 {
    public static void main(String[] args) {
        Set<Person> set = new TreeSet<>();
        set.add(new Person(154, "zhaoliu"));
        set.add(new Person(101, "zhangsan"));
        set.add(new Person(101, "zhangsan"));
        set.add(new Person(133, "wangwu"));
        set.add(new Person(115, "zhangsan"));
        set.add(new Person(115, "zhangsan"));
        set.add(new Person(112, "lisi"));
        System.out.println(set);
    }
}
```

答：

排错题4：

// 下面代码是否能正确遍历map集合？如果把键key换成String类型，遍历代码是否还能正常运行？

// 如果想要正常运行，请给出你的答案。

```
public class Demo4 {  
    public static void main(String[] args) {  
        Map<Integer,String> map = new HashMap<>();  
        map.put(1,"a");  
        map.put(2,"b");  
        map.put(3,"c");  
        map.put(4,"d");  
        map.put(5,"e");  
  
        for (int i = 0; i < map.size(); i++) {  
            String value = map.get(i);  
            System.out.println(i + " - " + value);  
        }  
    }  
}
```

答：

四、代码题：

第一题：分析以下需求，并用代码实现

训练目标：掌握Set集合的使用

需求背景：一个Set集合中，存入很多身份证数据，但是有些身份证是用户随意输入的错误号码，请使用代码从集合中，将这些错误代码删除。

```
Set<String> set = new HashSet<>();  
set.add("3726221951dg010416");  
set.add("37222219700211041X");  
set.add("37242219680208044x");  
set.add("354221968020850444");  
set.add("37262319800904575x");  
set.add("3722231af303a99673");  
set.add("3711fdg98603040033");  
set.add("373522198511190042");  
set.add("3755sdf1991112998X");
```

需求描述：

- (1)定义Set集合，存入多个字符串
- (2)删除集合中不满足身份证号码的数据

身份证要求：

- 长度必须18位
- 数字0不能开头
 - 除了最后一位，中间不允许有数字
 - 最后一位可以是数字或者Xx

- (3)然后利用迭代器遍历集合元素并输出

程序的运行效果如下所示：

```
Demo4 x
D:\develop\Java\jdk1.8.0_162\bin\java.exe
354221968020850444
37262319800904575x
37222219700211041X
373522198511190042
37242219680208044x
```

第二题：分析以下需求，并用代码实现

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

需求背景：中国福利彩票"双色球"投注区分为红色球号码区和蓝色球号码区，红色球号码区由1-33共三十三个数组成，蓝色球号码区由1-16共十六个数组成。如下所示：

选号

模拟投注为虚拟玩法，切勿当真

普通投注

胆拖投注

幸运选号

模拟摇奖

奖金计算

第2022009期 2022-01-20 20:00:00截止

红球区 至少选择6个红球

6个 机选红球 清空

遗漏

热号

01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18

1 0 0 13 4 2 0 27 0 3 11 3 7 8 2 3 5 5

19 20 21 22 23 24 25 26 27 28 29 30 31 32 33

3 0 0 1 1 4 1 8 11 1 15 4 2 4 18

蓝球区 至少选择1个蓝球

1个 机选蓝球 清空

遗漏

热号

01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16

18 2 7 8 9 5 14 6 11 17 62 61 0 4 37 1

您选择了0个红球，0个蓝球，共0注，共0元

选好了

双色球号码冷热分析

请选择: 近20期 区间怎么分?

红球1区 红球2区 红球3区 蓝球

状态	号码	次数	理论次数	偏差
热号	5	6	4	2
	10	6	4	2
	7	5	4	1
冷号	8	0	4	4
	11	2	4	2
	6	2	4	2

彩民之家

提供专业、权威、精准的预测分析

超多方案随意看

资深专家任你选

选号投注变简单

扫码查看

投注时选择6个红色球号码和1个蓝色球号码组成一注进行单式投注，每注金额人民币2元。

需求描述：现通过程序模拟双色球随机生成一注号码。程序运行结果如下所示：

```
|  
C:\develop\Java\jdk1.8.0_172\bin\java.exe ...  
红球: [5, 10, 24, 25, 27, 28] | 蓝球: 12
```

实现提示:

- 1、生成的双色球号码不能重复，因此可以考虑使用TreeSet集合存储双色球号码
- 2、针对红球的生成，需要生成多次，因此可以考虑使用循环

第三题：分析以下需求，并用代码实现

训练目标：掌握Java中Map集合的使用

需求描述：有类似这样的字符串：**"1.2, 3.4, 5.6, 7.8, 5.56, 44.55"**请按照要求，依次完成以下试题。

- 以逗号作为分隔符，把已知的字符串分成一个String类型的数组，数组中的每一个元素类似于"1.2","3.4"这样的字符串
- 把数组中的每一个元素以"."作为分隔符，把"."左边的元素作为key，右边的元素作为value，封装到Map中，Map中的key和value都是Object类型。
- 把map中的key封装的Set中，并把Set中的元素输出。
- 把map中的value封装到Collection中，把Collection中的元素输出。

程序的运行效果如下所示：

```
Demo4 x  
D:\develop\Java\jdk1.8.0_162\bin\java.exe .  
[44, 1, 3, 5, 7]  
[55, 2, 4, 56, 8]
```

第四题：【选做题】分析以下需求，并用代码实现

训练目标：掌握Java中Map集合的使用，以及了解工作开发中的场景。

需求描述：

1. 将以下车站对应关系的数据存储到map集合中，key: 表示站编号，value: 表示站名，并遍历打印(可以不按顺序打印):

2. 计算地铁票价规则:
总行程 3站内（包含3站）收费3元，3站以上但不超过5站（包含5站）的收费4元，5站以上的，在4元的基础上，每多1站增加2元，10元封顶;

3. 需要对键盘录入的上车站和到达站进行判断，如果没有该站，提示重新输入，直到站名存在为止,每站需要2分钟.

长沙轨道交通6号线时间表

序号	站名	序号	站名
1	黄花机场T1T2	2	木马垵
3	大路村	4	龙峰
5	曹家坪	6	檀木桥
7	龙华	8	韶光
9	东湖	10	农科院农大
11	隆平水稻博物馆	12	花桥
13	人民东路	14	芙蓉区政府
15	朝阳村	16	窑岭湘雅二医院
17	迎宾路口	18	烈士公园南
19	湘雅医院	20	文昌阁

21	六沟垅	22	湘雅三医院
23	白鸽咀	24	湖南工商大学
25	涧塘	26	麓谷公园
27	麓谷体育公园	28	长丰
29	和馨园	30	长庆
31	一师范西校区	32	中塘
33	象鼻窝	34	谢家桥

程序的运行效果如下所示：

```
D:\develop\Java\jdk1.8.0_162\bin\java.exe ...
```

1 - 分别为黄花机场T1T2 2 - 木马墩 3 - 大路村 4 - 龙峰 5 - 曹家坪
6 - 檀木桥 7 - 龙华 8 - 韶光 9 - 东湖 10 - 农科院农大
11 - 隆平水稻博物馆 12 - 花桥 13 - 人民东路 14 - 芙蓉区政府 15 - 朝阳村
16 - 窑岭湘雅二医院 17 - 迎宾路口 18 - 烈士公园南 19 - 湘雅医院 20 - 文昌阁
21 - 六沟垅 22 - 湘雅三医院 23 - 白鸽咀 24 - 湖南工商大学 25 - 涧塘
26 - 麓谷公园 27 - 麓谷体育公园 28 - 长丰 29 - 和馨园 30 - 长庆
31 - 一师范西校区 32 - 中塘 33 - 象鼻窝 34 - 谢家桥

请输入上车站:

aa

站点不存在, 请重新输入

请输入上车站:

bb

站点不存在, 请重新输入

请输入上车站:

檀木桥

请输入到达站

cc

站点不存在, 请重新输入

请输入到达站

dd

站点不存在, 请重新输入

请输入到达站

麓谷体育公园

从[檀木桥]到[麓谷体育公园]共经过21站, 收费10元, 大约需要42分钟