

学号	姓名	提交时间	总分	成绩
20225102010101	吴迪	2024-04-03 16:33:27	100	100

1. 【单选题】列哪个选项最准确地描述了 Map 的特点？

得分：5/5

- A. Map 可以存储重复的键，但不能存储重复的值。
- B. Map 存储的键值对是有序的。
- C. Map 的键和值可以是任意类型的对象。
- D. Map 中的键是唯一的，但值可以重复。

学生答案：D

参考答案：D

2. 【单选题】Map 的特点包括以下哪个？

得分：5/5

- A. 按照插入顺序保存键值对。
- B. 通过索引访问元素。
- C. 键是有序的。
- D. 可以存储重复的键和值。

学生答案：A

参考答案：A

3. 【单选题】在 Java 中，下列哪个选项描述了 Set 接口的特点？

得分：5/5

- A. 元素是有序的
- B. 可以存储重复的元素
- C. 元素是唯一的
- D. 允许通过索引访问元素

学生答案：C

参考答案：C

4. 【单选题】下列哪个方法用于将集合转换为 Stream 流

得分：5/5

- A. stream()
- B. convert()
- C. toStream()
- D. getStream()

学生答案：A

参考答案：A

5. 【单选题】在 Stream 流中，下列哪个方法用于过滤元素？

得分：5/5

- A. map()
- B. filter()

C.collect()

D.forEach()

学生答案：B

参考答案：B

6.【单选题】在 Stream 流中，下列哪个方法用于对元素进行转换或映射？

得分：5/5

A.filter()

B.collect()

C.forEach()

D.map()

学生答案：D

参考答案：D

7.【多选题】下列哪些选项描述了 HashMap、LinkedHashMap 和 TreeMap 的特点？

得分：5/5

A.HashMap 不保证元素的顺序。

B.LinkedHashMap 保持插入顺序。

C.TreeMap 不允许存储 null 键。

D.LinkedHashMap 使用哈希表实现。

学生答案：ABC

参考答案：ABC

8.【多选题】下列哪些方法是 Java 中 Map 接口的常用方法

得分：5/5

A.remove(key)

B.containsKey(key)

C.get(key)

D.values()

E.put(key, value)

学生答案：ABCD

参考答案：ABCD

9.【多选题】下列哪些方法可以用于遍历 Map 中的键值对

得分：5/5

A.使用 keySet() 方法获取键的集合，get()获取值 然后遍历集合

B.使用 iterator() 方法获取迭代器，然后遍历 Map。

C.使用 forEach() 方法遍历 Map 的键

D. 使用 entrySet()方法获取键值的集合，然后遍历集合。

学生答案：AD

参考答案：AD

10. 【多选题】下列哪些选项描述了 HashSet 的特点

得分：5/5

A. 元素是有序的

B. 允许存储 null 元素

C. 元素是唯一的

D. 不保证元素的顺序

E. 底层实现使用链表

学生答案：BCD

参考答案：BCD

11. 【多选题】 TreeSet 的特点

得分：5/5

A. 元素是有序的

B. 允许存储重复的元素

C. 允许存储 null 元

D. 底层实现使用红黑树

学生答案：ACD

参考答案：ACD

12. 【多选题】下列哪些选项描述了 LinkedHashMap 的特点

得分：5/5

A. 元素是有序的

B. 元素是唯一的

C. 底层实现使用链表和哈希表

D. 不保证元素的顺序

学生答案：ABC

参考答案：ABC

13. 【判断题】 IMap集合是键值对集合

得分：5/5

A. 正确

B. 错误

学生答案：A

参考答案：A

14. 【判断题】 Map集合应用场景|需要存储一一对应的数据时，就可以考虑使用Map集合来做

得分：5/5

A. 正确

B. 错误

学生答案：A

参考答案：A

参考答案 : public class MapDemo04 {

```
public static void main(String[] args) {

    //1.80个学生选择景点的数据准备好
    List<String> data = new ArrayList<>();

    String[] selects = {"A", "B", "C", "D"};

    Random r = new Random();

    for (int i = 1; i <= 80; i++) {

        // 每次模拟一个学生选择一个景点，存入到集合中去。
        int index = r.nextInt(4); // 0 1 2 3
        data.add(selects[index]);

    }

    System.out.println(data);

    //2.开始统计每个景点的投票人数
    //准备一个Map集合用于统计最终结果
    Map<String,Integer> result=new HashMap<>();

    //3.开始遍历80点数据
    for(String s:data){

        //问问Map集合中是否存在该景点
        if(result.containsKey(s)){

            //说明这个景点被人统计过，其值+1，存入到Map集合中
            result.put(s,result.get(s)+1);

        }else{

            //说明这个景点第一次统计，存入
            result.put(s,1);

        }

    }

    System.out.println(result);

}

}
```