Java8高级教程培训试题

一、单选题（3分）

1. 下面哪个接口是函数式接口？ （A）
2. public interface Adder{

int add(int a, int b);

}

1. public interface SmartAdder extends Adder{

int add(double a, double b);

}

1. public interface Nothing{ }
2. 全是
3. java.util.function.Predicate<T>接口定义了一个名叫test的抽象方法，以下说法正确的是（A）
4. 它接受泛型T对象，并返回一个boolean
5. 它接受泛型T的对象，没有返回（void）
6. 它接受一个泛型T的对象，并返回一个泛型R的对象
7. 其他说法都不正确
8. java.util.function.Function<T, R>接口定义了一个叫作apply的方法，以下说法正确的是（C）
   1. 它接受泛型T对象，并返回一个boolean
   2. 它接受泛型T的对象，没有返回（void）
   3. 它接受一个泛型T的对象，并返回一个泛型R的对象
   4. 其他说法都不正确
9. Java8的核心模块为（A）
10. Lambda 表达式与Stream API
11. Lambda 表达式与 默认方法
12. Lambda 表达式与 时间 API
13. 默认方法表达式与 时间 API
14. Lambda 表达式组成部分为（A）
15. 参数列表，箭头，Lambda主体
16. 参数列表，箭头，函数名
17. 参数列表，函数名，Lambda主体
18. 函数名箭头，Lambda主体
19. 关于函数式接口，下列说法正确的是（A）
20. 函数式接口只能有一个抽象方法
21. 函数式接口只能有一个方法
22. 函数式接口不能包含静态方法
23. 函数式接口不能包含默认方法
24. 对于Lambda表达式（args）->ClassName.staticMehod(args)，可以用下列哪个方法引用代替（A）
25. ClassName:: staticMehod
26. ClassName:: staticMehod()
27. ClassName:: staticMehod(args)
28. ClassName:staticMehod
29. 关于Java8新特性的说法错误的是（D）
30. 引入了Lambda表达式
31. 强大的StreamAPI
32. 引入了最大化减少空指针异常的Optional
33. Java8代码不能并行
34. 下列选项属于Stream的终端操作的是（A）
35. ForEach
36. Filter
37. Map
38. sorted
39. 下列流的操作不属于短路操作的是（D）

(A)anyMatch

(B)allMatch

(C)noneMatch

(D)map

二、判断题（2分）

1. 哪怕有很多默认方法，只要接口只定义了一个抽象方法，它就仍然是一个函数式接口。

（正确）

1. 行为参数化即让方法接受多种行为（或策略）作为参数，并在内部使用，来完成不同的行为。

（正确）

1. (String s) -> s.length()为有效的Lambda表达式。

（正确）

1. (Integer i) -> return "Alan" + i;为有效的Lambda表达式（错误）
2. 函数式接口可以定义多个抽象方法。

（错误）

1. Lambda表达式()-> System.out.println ("This is awesome!!")可以看成是是Runnable接口中run方法的签名。

（正确）

1. Predicate<Apple> p=(Apple a)->a.getWeight();是有效的Lambda 表达式示例。

（错误）

1. 函数式接口必须添加@FunctionalInterface注解。

（错误）

1. java.util.function.Predicate<T>接口定义了一个名叫test的抽象方法，它接受泛型T对象，并返回一个boolean。

（正确）

1. java.util.function.Consumer<T>定义了一个名叫accept的抽象方法，它接受泛型T的对象，没有返回（void） 。

（正确）

1. java.util.function.Function<T, R>接口定义了一个叫作apply的方法，它接受一个泛型T的对象，并返回一个泛型R的对象。

（正确）

1. IntPredicate evenNumbers = (int i) -> i % 2 == 0; evenNumbers.test(1000);该操作会默认对1000进行装箱操作。

（错误）

1. 如果一个Lambda的主体是一个语句表达式， 它就和一个返回void的函数描述符兼容（当然需要参数列表也兼容）。

（正确）

1. 假设有一个构造函数没有参数，它适合Supplier的签名() -> Apple。那么对于Lambda表达式Supplier<Apple> c1 = Apple::new等价于Supplier<Apple> c1 = () -> new Apple()。（正确）
2. Comparator.comparing(Apple::getWeight);表示对苹果按照重量进行递减排序。

（错误）

1. 谓词符合中，and和or方法是按照在表达式链中的位置，从左向右确定优先级的。

（正确）

1. 可以把 Lambda表达式理解为是一段可以传递的代码（将代码像数据一样进行传递）。

（正确）

1. Stream的处理对象主要是集合，可以执行查找过滤、映射等操作。

（正确）

1. Stream可以存储元素。

（错误）

1. Stream的中间操作的每个环节都会返回一个数据结果集。

（错误）

1. 由有序集合生成的流会保持原有的顺序。

（正确）

1. Stream只有在执行终端操作时才会返回一个数据结果集。

（正确）。

1. 对Stream进行筛选的操作使用filter。

（正确）

1. 截取流的前3个数据使用limit(3)。

（正确）

1. 映射Map会修改数据源的元素。

（错误）

1. flatmap方法让你把一个流中的每个值都换成另一个流，然后把所有的流连接起来成为一个流。

（正确）

1. 对流的操作中，如果不关心返回的元素是哪个，应使用findAny。

（正确）

1. parallelStream（）方法会返回一个并行流。

（正确）

1. 由数组创建流需要使用Arrays 的静态方法 stream()。

（正确）

1. Java8中默认方法由default修饰符修饰。

（正确）

1. 解决默认方法冲突的规则中，类中的方法优先级最高。

（正确）

1. LocalDateTime类的实例是可变对象。

（错误）

1. 计算两个“时间间隔”使用Duration。

（正确）

1. 和老的java.util.DateFormat相同，Java8 中的所有的DateTimeFormatter实例也是非线程安全的。

（错误）

1. 可以向Optional对象的of方法传递一个Null值。

（错误）