1. 定义泛型接口如下：

public interface Generator<T> {

public T method();

}

该实现方法是否正确？ T or F

class GeneratorImpl<T> implements Generator<T>{

@Override

public T method() {

return null;

}

}

该实现方法是否正确？ T or F

class GeneratorImpl<T> implements Generator<String>{

@Override

public String method() {

return "hello";

}

}

2.小知识

2.1 String、StringBuffer 和 StringBuilder 的区别是什么? String 为什么是不可变的?

String 类中使用 final 关键字修饰字符数组来保存字符串，private final char value[]，所以 String 对象是不可变的。

补充：在 Java 9 之后，String 类的实现改用 byte 数组存储字符串 private final byte[] value;

而 StringBuilder 与 StringBuffer 都继承自 AbstractStringBuilder 类，在 AbstractStringBuilder 中也是使用字符数组保存字符串char[] value 但是没有用 final 关键字修饰，所以这两种对象都是可变的。

StringBuilder 与 StringBuffer 的构造方法都是调用父类构造方法也就是AbstractStringBuilder 实现的，可以自行查阅源码。

2.2 == 与 equals

== : 它的作用是判断两个对象的地址是不是相等。即，判断两个对象是不是同一个对象(基本数据类型==比较的是值，引用数据类型==比较的是内存地址)。

equals() : 它的作用也是判断两个对象是否相等。但它一般有两种使用情况：

情况 1：类没有覆盖 equals() 方法。则通过 equals() 比较该类的两个对象时，等价于通过“==”比较这两个对象。

情况 2：类覆盖了 equals() 方法（String类就覆盖了equals() 方法）。一般，我们都覆盖 equals() 方法来比较两个对象的内容是否相等；若它们的内容相等，则返回 true (即，认为这两个对象相等)。

3 判断 T or F

Arrays.asList()是否可以传入基本类型。

List存储的元素是有序的，不可重复的。

Set存储的元素是无序的，不可重复的。

Map使用键值（key, value）存储元素，key和value都是是无序的，不可重复的

4. 编程

给出如下Person类，根据注释补全main代码

**public** **class** Person **implements** Comparable<Person> {

**private** String name;

**private** **int** age;

**public** Person() {

**super**();

}

**public** Person(String name, **int** age) {

**super**();

**this**.name = name;

**this**.age = age;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** **int** getAge() {

**return** age;

}

**public** **void** setAge(**int** age) {

**this**.age = age;

}

@Override

**public** **int** compareTo(Person p) {

**return** **this**.age - p.age;

}

}

需要补全的代码如下：

**import** java.util.ArrayList;

**import** java.util.Collections;

//import java.util.Comparator;//有能力的可以通过重写排序方法按年龄从大到小排序

/\*

\* 用ArrayList存储自定义对象，并按照年龄进行排序

\*/

**public** **class** ArrayListTest {

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

// 创建集合对象

// 创建元素对象按顺序(曹操35岁) (蒋干33岁)(诸葛亮34岁)

// 添加元素

// 排序，按年龄从小到大

// 遍历集合输出整个集合

}

}