

====20201009====

一、排序算法的稳定性

1. 不稳定：快排、堆排、选择、希尔；
2. 稳定：冒泡、插入、归并、基数。

====20201007====

一、单例的特点

1. 私有构造函数；
2. 私有静态全局变量；
3. 公有静态方法

====20201007====

一、01 + 02 = 3.00000004

1. 浮点数错误，二进制无法在有限长度中精确地表示十进制中的0.1和0.2；
2. 单精度32位，双精度64位，位数决定了它们能够表示的精度上限；
3. 解决方法是BigDecimal，它的原理是，十进制整数转换成二进制数时不会有精度问题，那么把十进制小数扩大10的N次方倍，在整数维度上进行计算，并保留相应的精度信息，就不会丢失精度，属性scale保存精度信息，intCompat保存被放大后的整数。
4. 二进制小数点后面是0.5，0.25，每一位都是前一位的一半，0.1和0.2无法精确表示，只能不断接近这个真实值；
5. 单精度float32位，1符号位，8指数位，23小数位；双精度double64位，1符号位，11指数位，52小数位；

====20201006====

一、static关键字

1. 定义：static关键字可用于变量、方法、代码块、内部类，表示某个成员只属于这个类本身，而不属于该类的某个对象。
2. 静态变量：
 - a. 静态变量属于一个类，直接通过类名来访问，不能通过类的实例对象访问静态成员；
3. 静态方法：

- a. 静态方法不能重载，应为静态方法在编译时解析，方法重写发生在运行时阶段，为了多态；
- b. 静态方法不能直接访问成员方法和成员变量；
 - i. 静态成员属于类，在加载的时候就会分配内存，可以通过类名直接访问；非静态方法属于对象，只有在对象实例化之后才存在。类加载是在需要调用的时候加载。
 - ii. 类的非静态成员不存在的时候，静态成员就已经存在了，所以静态成员调用非静态成员会报错。在编译阶段检查这种错误。

4. 静态代码块：

- a. 静态代码块的执行顺序和它在类中的位置是一致的。

5. 静态内部类：

- i. 静态内部类可以访问外部类的静态变量，包括私有静态变量，当时不能访问非静态变量，外部类不能为static。

二、this关键字

- 1. 消除字段歧异；
- 2. 引用类的其他构造方法；
- 3. 作为参数传递；
- 4. 链式调用；
- 5. 在内部类中访问外部类的对象。