复习：

1. 继承：用extends关键字 --- 单继承；单继承和多继承的比较

2. super：在子类中表示父类对象的引用，用于调用父类中的方法和属性。--- super语句，用在子类构造方法调用父类对应形式的构造方法 --- 创建一个子类对象必然伴随着创建一个父类对象

3. 多态：

编译时多态：方法的重载

运行时多态：向上造型（用父类来声明，用子类来创建）、方法的重写（两等两小一大）

## static

static可以修饰变量、方法、内部类和代码块

### 静态变量

也叫类变量。随着类的加载而加载到方法区，并且在方法区中被赋予了默认值。由于静态变量是随着类出来的，比对象先出现，可以通过类名来调用。这个类的所有对象存的是静态变量在方法去中的地址，所以所有的对象都使用的是同一个静态变量。

静态变量只能定义在类中。

### 静态方法

在类加载的时候加载到方法区中，在方法区中不执行只是存储，在方法被调用的时候到栈内存中执行。静态方法比对象要先出来，所以可以通过类名来调用静态方法。

Arrays.toString() Arrays.sort() Arrays.copyOf() System.arraycopy()

在静态方法中可以定义静态变量吗？ --- 不可以 --- 静态方法只有在被调用的时候才会执行，只有执行的时候才会将方法中的变量来进行初始化，方法在栈内存中执行，所以方法中的变量也是存储在栈内存中；静态变量在类加载的时候初始化，静态变量存储在方法区

在静态方法中可以直接调用本类中的非静态方法吗？ --- 不可以 --- 非静态方法在本类中是通过this调用，this代表当前对象的引用，静态方法在对象之前出来的，也就意味着静态方法执行的时候可以没有对象

public class A {

public void m1(){}

public static void m2(){ m1(); }

}

静态方法可以重载吗？ --- 可以

静态方法可以被继承吗？ --- 可以

静态方法可以重写吗？ --- 不可以

父子类中是否可以存在方法签名一致的静态方法？---可以 --- 隐藏(hide) --- 如果父子类中存在了方法签名一致的方法，要么都是非静态，要么都是静态

### 静态代码块

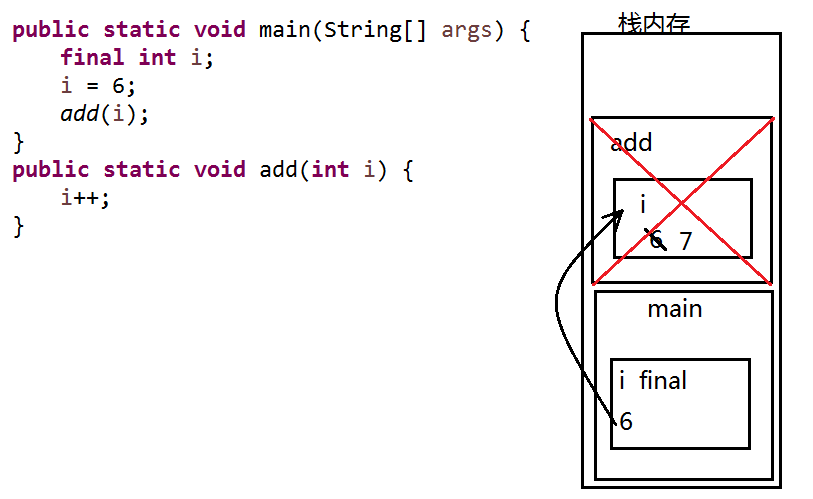
用static修饰的代码块就称之为静态代码块。--- 只在类加载的时候执行一次。

执行顺序：父类静态 -> 子类静态 -> 父类构造 -> 子类的构造

## final

修饰数据、方法以及类

final修饰数据---常量，定义好之后值不可变。如果是基本类型的常量，指的是实际值不可变；如果引用类型的常量，表示地址不可变，而其中的元素或者属性可以改变。



public class A {

final int i;

public A(){ this(5); }

public A(int i){ this.i = i; }

}

A a = new A(3);

最终方法 --- 不能被重写/隐藏。可以被重载，可以被继承。

最终类 --- 不能被继承。