**一: 必须要懂得概念**

1. **类是什么?类的组成是哪些?**

答案：

类：可以理解为构造对象的一个蓝图或者模板，是抽象的概念。

类由构造方法 成员变量和成员方法组成

1. **对象是什么?什么时候需要使用对象?**

答案：

对象：是以类为模型创建的具体实例，是对类的一种具体化。

当需要访问成员变量或成员方法的时候，需要使用对象

1. **封装是什么?**

答案：指隐藏对象的属性和实现细节，仅对外提供公共访问方式

1. **继承是什么?**

答案：

Java继承是面向对象的最显著的一个特征。继承是从已有的类中派生出新的类，新的类能吸收已有类的数据属性和行为，并能扩展新的能力。

Java继承是使用已存在的类的定义作为基础建立新类的技术。新类的定义可以增加新的数据或新的功能，也可以用父类的功能，但不能选择性地继承父类。

1. **多态是什么?**

答案：

多态按字面的意思就是“多种状态”。在面向对象语言中，接口的多种不同的实现方式即为多态。

多态性是允许你将父对象设置成为何一个或更多的他的子对象相等的技术，赋值之后，父对象就可以根据当前赋值给它的子对象的特性以不同的方式运作。

简单的说，就是一句话： 允许将子类类型对象的引用 赋值给父类类型的引用

1. **抽象类是什么?**

答案：使用了关键字abstract声明的类叫做“抽象类”。如果一个类里包含了一个或多个抽象方法，类就必须指定成abstract（抽象）。“抽象方法”，属于一种不完整的方法，只含有一个声明，没有方法主体。

1. **接口是什么?**

答案：Java接口是一系列方法的声明，是一些方法特征的集合，一个接口只有方法的特征没有方法的实现，因此这些方法可以在不同的地方被不同的类实现，而这些实现可以具有不同的行为（功能）。

1. **匿名对象是什么?怎么用?**

答案：匿名对象，就是没有名字的对象。是对象的一种简化表示形式。

匿名对象的两种使用情况

1. 对象调用方法仅仅一次的时候
2. 作为实际参数传递
3. **构造方法的特点?构造方法的注意事项?**

答案：

构造方法的特点：

方法名与类名相同

没有返回值类型，连void都没有

没有具体的返回值

构造方法注意事项：

如果我们没有给出构造方法，系统将自动提供一个无参构造方法

如果我们给出了构造方法，系统将不再提供默认的无参构造方法。

注意：这个时候，如果我们还想使用无参数构造方法，就必须自己给出。建议永远自己给出无参数构造方法。

1. **方法重写是什么?注意事项是什么?**

答案：

方法重写：指子类中出现了和父类中一模一样的方法声明，也被称为方法覆盖，方法复写

使用特点：

如果方法名不同，就调用对应的方法

如果方法名相同，最终使用的是子类自己的

需要注意的问题：

父类中私有方法不能被重写

子类重写父类方法时，访问权限不能更低

父类静态方法，子类也必须通过静态方法进行重写。（其实这个算不上方法重写，但是现象确实如此）

**12、请说说下面关键字分别是什么，有什么用?**

**private**

**this**

**super**

**static**

**final**

**abstract**

**interface**

**extends**

**implements**

答案：

private：是一个权限修饰符，可以修饰成员（成员变量和成员方法），被private修饰的成员只能在本类中访问。

this：方法被哪个对象调用，this就代表那个对象。this用来解决成员变量与局部变量重名问题。

super：代表父类存储空间的标识（可以理解为父类引用）。

static：静态的意思，static修饰的成员，被类的所有对象共享，也可以通过类名调用。

final：最终的意思，可以修饰类，成员变量，成员方法。

abstract：抽象的意思，用来修饰抽象类与抽象方法

interface：接口的意思，用来定义接口。

extends：继承的意思，通过它可以类与类之间产生继承关系。

implements：实现的意思，通过它可以让类与接口之间产生实现关系。

**比较散的概念（多看几遍）**

**这些概念不会单独体现出来，在我们后面学习的过程就包含了**

**就类似于我们 第一次开车 心里要想着 先开车门 调座椅**

**安全带 左脚离合 右脚油门 拧钥匙 挂挡 给油……**

**对于一个熟练地司机来说 这些都会变成一种习惯**

**给自己信心 做一名老司机**

1. **final关键字是什么，可以干什么?分别是哪些特点?**

答案：final最终的意思。

在编写程序的过程中，已定义好的数据或者方法不希望被其他人改动，可以通过final来实现。

final修饰类，此类不能被继承，类中的所有方法默认为final修饰。

final修饰变量，变量就成了常量，只能被赋值一次

final修饰方法，方法不能被重写

**2、 private关键字在哪里使用？ 被private修饰的成员有什么特点？**

答案：

类中的成员需要私有的时候使用private关键字

特点：

是一个权限修饰符。

可以修饰成员(成员变量和成员方法)

被private修饰的成员只在本类中才能访问

**3、为什么要有this关键字？this关键字代表的对象什么时候存在的？**

答案：

为什么要有this关键字？

this用来解决成员变量与局部变量重名问题

this关键字代表的对象什么时候存在的？

当方法被调用时，this关键字存在

this关键字代表所在类的对象引用；方法哪个对象调用，this就代表哪个对象。

**4、内部类有哪些访问特点？** 答案：

1. 内部类可以直接访问外部类的成员，包括私有
2. 外部类要访问内部类的成员，必须先创建内部类对象

**辨析的一些概念 有些概念容易混淆 就拿出来 给大家做个整理总结**

**看的时候如果有问题的话 记录在 一个txt中 可以在上课之前发给我**

1. **成员变量与局部变量的区别？**

答案：

A:在类中的位置不同

成员变量：在类中方法外

局部变量：在方法定义中或者方法声明上

B:在内存中的位置不同

成员变量：在堆内存

局部变量：在栈内存

C:生命周期不同

成员变量：随着对象的创建而存在，随着对象的消失而消失

局部变量：随着方法的调用而存在，随着方法的调用完毕而消失

D:初始化值不同

成员变量：有默认初始化值

局部变量：没有默认初始化值，必须定义，赋值，然后才能使用。

**2、思考：什么时候使用this关键字？**

**什么时候使用private？**

**如何给成员变量赋值？**

答案：

使用this关键字：

创建对象，为对象中的成员赋值时使用

成员变量与局部变量重名时使用

使用private关键字：

封装类中成员时使用，隐藏实现细节

给成员变量赋值：

通过构造方法赋值

通过setXxx()方法赋值

**3、 抽象类和接口的区别?**

答案：

抽象类：含有abstract修饰的类即为抽象类，抽象类不能创建实例对象。含有abstract方法的类必须定义为抽象类，抽象类中的方法不必是抽象的。抽象类中定义抽象方法必须在具体子类中实现，所以，不能有抽象构造方法或抽象静态方法。如果子类没有实现抽象父类中的所有抽象方法，那么子类也必须定义为abstract类型。

接口：可以说成是抽象类的一种特例，接口中的所有方法都必须是抽象的。接口中的方法定义默认为public abstract类型，接口中的成员变量类型默认为public static final。

下面比较一下两者的语法区别

1. 抽象类可以有构造方法，接口中不能有构造方法。
2. 抽象类中可以有普通成员变量，接口中没有普通成员变量
3. 抽象类中可以包含非抽象的普通方法，接口中的所有方法必须都是抽象的，不能有非抽象的普通方法。
4. 抽象类中的抽象方法的访问类型可以使public、protected和默认类型，但接口中的抽象方法只能是public类型的，并且默认修饰即为public abstract类型。
5. 抽象类中可以包含静态方法，接口中不能包含静态方法
6. 抽象类和接口中都可以包含静态成员变量，抽象类中的静态成员变量的访问类型可以任意，但接口中定义的变量只能是public static final类型，并且默认即为public static final类型。
7. 一个类可以实现多个接口，但只能继承一个抽象类。

**4、方法重写和重载有什么区别?**

答案：

重载Overload表示同一个类中可以有多个名称相同的方法，但这些方法的参数列表各不相同(即参数个数或类型不同)。

重写Override表示子类中的方法可以与父类的某个方法的名称和参数完全相同，通过子类创建的实例对象调用这个方法时，将调用子类中的定义方法，这相当于把父类中定义的那个完全相同的方法给覆盖了，这也是面向对象编程的多态性的一种表现。子类覆盖父类的方法时，只能比父类抛出更少的异常，或者是抛出父类抛出的异常的子异常，因为子类可以解决父类的一些问题，不能比父类有更多的问题。子类方法的访问权限只能比父类的更大，不能更少。如果父类的方法是private类型，那么，子类则不存在覆盖的限制，相当于子类中增加了一个全新的方法。