**软件综合训练**

1. 课程地位
2. 发展方向
3. 就业方向

第1周

java基础

面向对象基础、类之间的关系、抽象类和接口（4）

核心类（1）

泛型和集合（1）

**一、面向对象基础**

在Java编程思想中提到了面向对象程序设计的五个特征：

* 万物皆为对象。也就是说世界上所有的事物都能通过抽象变成一个可以实现的对象。比如动物、建筑物、服务。
* 程序是对象的集合，他们通过发送消息来告知彼此所有做的。程序是把要解决的问题抽象为若干的对象，这些对象之间通过消息来通知彼此做什么和传递数据。比如：动物吃食物，这个程序包含了动物、食物两种对象，通过吃这一动作来告知动物吃食物。
* 每个对象都有自己由其他对象构成的存储。意思是对象之间可以是包含的关系，一个对象中可以含有另外一个对象，这样就构成了一个新的对象。**比如**：发动机，汽车，汽车这一对象中包含发动机。
* 每个对象都拥有其类型。每个对象都是某个类的一个实例，这个类就是抽象出来的一类事物，这个类与其他类之间的重要区别就是行为，它能接受什么样的消息。比如：动物和食物，动物有吃的行为，移动的行为，而食物是没有的。
* 某一特定类型的所有对象都可以接受同样的消息。这是判断某一对象是不是某个类型的标准。比如你定义了动物的一些行为，比如移动，那么所有的动物都要相应移动这一消息，无论你是跑着移动、飞着移动，还是游着移动。

上面是面向对象程序设计的基本特征，下面介绍面向对象中用到的一些关键词。

**类：**

类描述了具有相同特性（数据元素）和行为（功能）的对象集合，也是一个数据类型。比如说动物，它有重量、大小的特征（数据元素），也有移动、吃的行为（功能）。

**对象：**

对象是具有状态、行为和标识的，是上面类的一个实例。每一个对象都有内部数据(它描述该对象的状态)和方法（它产生行为），并且每一个对象都可以与其他对象区分开来，就是一个对象在内存中都有一个唯一的地址。在Java中对象都必须通过new来产生。

**继承：**

把某些具有相同属性和行为的事物抽象起来集中到一个类中，一个新类可以从这个类中派生，新类具有父类的所有的行为和特征，这个过程称为类继承。新类继承了原始类的特性，新类称为原始类的派生类（子类），而原始类称为新类的基类（父类），在Java中是通过 ChildCalss extends FartherClass 这种方式实现的。

**抽象：**

抽象在编程中就是把你要解决的问题分解为一个一个的小问题，直到你不能再分解为止，然后把你分解的这些小问题用一个一个的对象描述对应起来，将问题集合分解为**对象集合**。实际中就是不断的问需要哪些服务，需要哪些对象才能履行它们的服务。良好的面向对象的设计中，每个对象都可以很好的完成一项任务，但是它并不试图做更多的事。

**多态：**

多态指的是在继承关系中子类继承父类的行为，这种行为在子类具体实现时可以是不一样的。比如动物这一父类中有move()方法，在鸟类子类中和鱼类子类中move的实现是不一样的。在Java中多态是通过重写(override：把父类的方法重新实现，动态的)和重载(overload：通过方法参数的不同在子类中来重新创建名字相同的方法，静态的)来实现的。

**面向对象三要素：封装、继承、多态**

①. 封装：封装的意义，在于明确标识出允许外部使用的所有成员函数和数据项，或者叫**接口**。

②. 继承：

继承基类的方法，并做出自己的扩展；

声明某个子类兼容于某基类（或者说，接口上完全兼容于基类），外部调用者可无需关注其差别（内部机制会自动把请求派发dispatch到合适的逻辑）。

③. 多态：基于对象所属类的不同，外部对同一个方法的调用，实际执行的逻辑不同。很显然，多态实际上是依附于继承的第二种含义的。

## 一. 多态

方法签名：方法名 + 参数列表(参数类型、个数、顺序)

1.1 重写

子类重写父类方法，只有实例方法可以被重写，重写后的方法必须仍为实例方法。成员变量和静态方法都不能被重写，只能被隐藏。

重写实例方法：超类Parent中有实例方法A，子类child定义了与A 相同签名和子集返回类型 的实例方法B，子类对象ChildObj只能调用自己的实例方法B。

方法的重写（override）两同两小一大原则：

* 方法名相同，参数类型相同
* 子类返回类型小于等于父类方法返回类型
* 子类抛出异常小于等于父类方法抛出异常
* 子类访问权限大于等于父类方法访问权限（比如，A类的方法一，是public修饰的，B类继承A类，B类即为子类，重写A类的方法一，不能将此时B类的方法一的public修饰词改为protected或者private这样权限小的修饰词。所以，子类的重写方法的权限，不能低于父类方法的权限）

注意：

* 不能重写static静态方法。(形式上可以写，但本质上不是重写，属于下面要讲的隐藏)
* 重写方法可以改变它的方法修饰符，如final,synchronized,native。不管被重写方法中有无final修饰的参数，重写方法都可以增加、保留、去掉这个参数的 final 修饰符(参数修饰符不属于方法签名)。
* 1.2 重载
* 在同一个类中，有多个方法名相同，参数列表不同（参数个数不同，参数类型不同），与方法的返回值无关，与权限修饰符无关。编译器通过对方法签名的识别即可静态编译出不同的方法。这也是java中重载与重写的区别之一。
* 重载只是一种语言特性，与多态无关，与面向对象也无关。多态是为了实现接口重用。
* Java中方法是可以和类名同名的，和构造方法唯一的区别就是，构造方法没有返回值。
* 1.3 隐藏
* 隐藏与覆盖在形式上极其类似(语法规则)，但有着本质的区别：只有成员变量(不管是不是静态)和静态方法可以被隐藏。

1.4 成员变量

超类 Parent 中有成员变量 A ，子类 Child 定义了与 A 同名的成员变量 B ，子类对象 ChildObj 调用的是自己的成员变量 B。如果把子类对象 ChildObj 转换为超类对象 ParentObj ，ParentObj 调用的是超类的成员变量 A ！

* 隐藏成员变量时，只要同名即可，可以更改变量类型(无论基本类型还是隐藏类型)
* 不能隐藏超类中的 private 成员变量，换句话说，只能隐藏可以访问的成员变量。
* 隐藏超类成员变量 A 时，可以降低或提高子类成员变量B的访问权限，只要A不是 private。
* 隐藏成员变量与是否静态无关！静态变量可以隐藏实例变量，实例变量也可以隐藏静态变量。
* 可以隐藏超类中的final成员变量。

1.5 静态方法

超类 Parent 有静态方法 A ，子类 Child 定义了与 A 相同签名和子集返回类型 的静态方法 B ，子类对象 ChildObj 调用的是自己的静态方法 B 。如果把子类对象 ChildObj 转换为超类对象 ParentObj ，ParentObj 调用的是超类的静态方法 A ！

隐藏后的方法必须仍为静态方法

Java基础详细学习链接: <https://wenku.baidu.com/view/302c725f76a20029bc642d3f.html?sxts=1599315383564&word=undefined&pn=51>

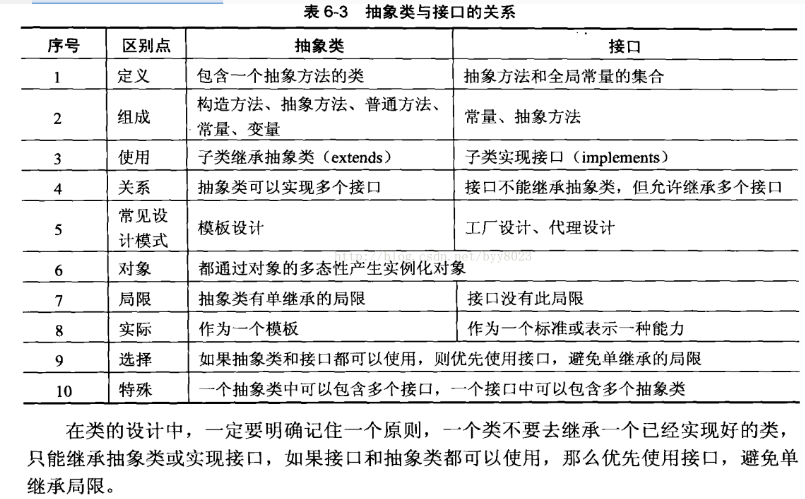
Java基础学习最新总结：https://wenku.baidu.com/view/65d5c739941ea76e58fa04e8.html?sxts=1599316907728&word=undefined

eclipse使用教程

https://zhinan.sogou.com/guide/detail/?id=1610014301

关于接口，抽象类，实体类的关系

接口:把某个功能或者某个模块抽象出来，先定义了他们能干什么，也就是写成抽象方法，即规定了每个方法的方法名，参数列表，返回类型，相对于我们做项目来说，就是告诉下面的程序员，当你要在这个功能模块下开发方法的具体实现的时候，为了统一与规范，并须与接口中已经定义了的方法的参数列表，返回类型一致，即要去实现接口。  
  
抽象类：抽象类是API中的第二层，之所以用到它，是因为下面的实现类如一般只会去实现顶层接口的一部分功能，如果直接去实现接口就必须要实现接口中所有的方法，所以先用抽象类去实现接口的一部分方法（抽象类可以实现一部分，而实体类必须实现全部），最后让实体类去继承抽象类的一部分功能。  
  
抽象类里既有抽象方法又有已实现方法是因为，抽象类中有一类方法是下面的实现类通用的，不会有变化，所以在抽象类里面就实现了，而不通用的方法就写成抽象方法，让实现类各自去实现。  
  
由于API分类更加复杂，一般需要分成更多层，所以有时候接口会继承接口，实现类也可能直接实现接口。



# Java核心类库

Java类库链接：https://blog.csdn.net/duan196\_118/article/details/104111149

泛型链接：<https://www.zhihu.com/question/374453817>

集合链接：<http://c.biancheng.net/view/6824.html>

http://c.biancheng.net/view/1282.html

第2周

1、java高级

JDBC编程（1.5课时）

IO流（1课时）

多线程（1课时）

2、MySql数据库

建库、建表、建约束（0.5课时）

数据的增删改查、高级查询（1课时）

存储过程（1课时）

第3周

Web基础

常用几个html标签（0.5课时）

CSS样式基础知识（1课时）

javaScript、jQuery（3.5课时）

第4周

JavaWeb开发技术

Servlet（4课时）

jsp（1课时）

监听器和过滤器（1课时）

文件上传下载（1课时）

第5~8周

在线投票系统（JavaWeb+MySQl）

第:9~10周

SSM框架

spring（5课时）

mybatis（2课时）

springmvc（5课时）

SSM（Spring+SpringMVC+MyBatis）框架集bai由Spring、SpringMVC、MyBatis三个开源框架整合而成，du常作为数据源较简单的web项目的框架。zhi

其中daospring是一个轻量级的控制反转（IoC）和面向切面（AOP）的容器框架。

SpringMVC分离了控制器、模型对象、分派器以及处理程序对象的角色，这种分离让它们更容易进行定制。

MyBatis是一个支持普通SQL查询，存储过程和高级映射的优秀持久层框架。

第11~16周

项目2：网上商城系统