# 《面向对象与多线程综合实验》指导书

本课程旨在使学生了解软件开发的一般过程，深入理解面向对象语言的基本概念和基本原理，理解和掌握继承与多态、异常处理、输入输出流、GUI设计、JDBC数据库操作、网络编程、多线程等技术；熟练掌握在Java语言环境下，上述技术的具体实现方法，并深入领会Java程序设计实用开发技术。

为达到上述目标，本课程计划让学生以迭代方式逐步编程开发一个小型档案管理系统，实现上述7个部分的训练，从而培养学生分析和解决问题的能力。（基于授课对象未曾学习过网络、数据库，为简化起见）该小型档案系统要求如下：

1. 系统基于C/S模式，包括客户端子系统，服务器端子系统
2. 用户需登录，验证口令通过后才能使用系统。用户分为系统管理人员、档案录入人员，档案浏览人员
3. 系统中相关用户信息、档案属性存放于关系数据库中，档案本身以文件形式存放于服务器相关目录中
4. 客户端和服务器端基于Socket实现通信，服务器端需实现多线程功能，可同时处理、响应多个客户端的数据请求
5. 用户信息管理功能，普通用户登录后可对自己基本信息(用户名、密码、角色)的查询、修改（只能修改密码，不可修改用户名、角色）。系统管理人员除上述功能外可增加、删除用户、修改用户的相关属性。
6. 档案数据录入功能，档案录入人员可输入新的档案文件信息，并将档案文件上传至服务器
7. 档案数据浏览功能，普通用户可查询相应的档案文件信息，对相关档案文件可以下载保存

## 封装、继承与多态

### 1.1 知识要点

（1）封装

封装(Encapsulation)是类的三大特征之一，即将类的状态信息隐藏在类内部，不允许外部程序直接访问，必须通过该类提供的方法来实现对隐藏信息的操作和访问。简而言之就是隐藏对象的属性和实现细节，仅向外提供访问接口。

封装的好处主要有：隐藏类的实现细节；让使用者只能通过程序员规定的方法来访问数据；可以方便的加入存取控制语句，限制不合理操作。

封装的具体步骤：修改属性的可见性来限制对属性的访问；为每个属性创建一对赋值(setter)方法和取值(getter)方法，用于对这些属性的存取；在赋值方法中，加入对属性的存取控制、有效性检查等。

（2）继承

继承(Inheritance)是Java中实现代码重用的重要手段，它是一种软件复用形式。继承是使用已存在的类的定义作为基础建立新类的技术，新类可以吸收现有类的成员，并可以增加新的功能或修改原有的功能。这种技术使得复用以前的代码非常容易，能够大大缩短开发周期，降低开发费用。

继承表达的是is-a的关系，或者说是一种特殊和一般的关系。Java中只支持单继承，即每个类只能有一个父类。在Java中所有的类都直接或间接的继承了java.lang.Object类，即Object类是所有java类的祖先。

在Java中，子类可以从父类中继承的有：继承public和protected修饰的属性和方法，不管子类和父类是否在同一个包里；继承默认权限修饰符修饰的属性和方法,但子类和父类必须在同一个包里。子类无法继承父类的有：无法继承private修饰的属性和方法；无法继承父类的构造方法。

（3）多态

多态(Polymorphism) 即多种状态，是指计算机程序运行时，系统可依据对象所属类，引发对应类的方法，从而有不同的行为。简单来说，所谓多态意指相同的消息给予不同的对象会引发不同的动作。

多态又可分为设计时多态和运行时多态。重载又被称为设计时多态，而对于覆盖或继承的方法，Java运行时系统根据调用该方法的实例的类型来决定选择调用哪个方法则被称为运行时多态。多态的优点可归纳为五个方面：可替换性、可扩充性、接口性、灵活性和简化性。

在Java中实现多态的三个条件：继承的存在(继承是多态的基础,没有继承就没有多态)；子类重写父类的方法(多态下调用子类重写的方法)；父类引用变量指向子类对象(子类到父类的类型转换)。

特别注意：子类转换成父类时的规则：将一个父类的引用指向一个子类的对象，称为向上转型，自动进行类型转换。此时通过父类引用调用的方法是子类覆盖或继承父类的方法，不是父类的方法。此时通过父类引用变量无法调用子类特有的方法。如果父类要调用子类的特有方法就得将一个指向子类对象的父类引用赋给一个子类的引用，称为向下转型,此时必须进行强制类型转换。

### 实验目的

掌握类的定义、对象的创建、对象的属性的引用和方法的调用；熟悉Java中的继承机制，方法的重载与覆盖；掌握多态、抽象类、接口的使用。

### 1.3 实验内容

编写一个程序，实现档案管理系统中的用户管理模块。要求模块中实现用户的模拟登录过程。通过用户输入，获取用户名和口令；与事先记录在程序中的用户信息进行对比，通过口令验证后才能使用系统。用户分为系统管理人员、档案录入人员，档案浏览人员三类，相关类图如下所示。

（1）要求在用户类中实现封装，并构造setter、getter方法实现属性的访问。

（2）通过继承方式加以实现上述类。

（3）通过多态，实现用户菜单项的展示，根据用户角色不同，系统自动调用对应showMenu()方法。

（4）在未讲数据库之前，系统中已存在用户的信息放置在Hashtable中。提供DataProcessing类实现数据的查找、插入、更新和删除。（此类可提供给学生直接使用）

（5）在未讲I/O之前，系统中的文件操作的方法可虚化，只用打印语句即可。

## 2. 异常处理

## 3. 输入输出流

## 4. GUI设计

## 5. JDBC数据库操作

## 6. 网络编程

## 7. 多线程