

软件体系结构 设计模式实验报告

学生姓名：_____

所在学院：_____

学生学号：_____

学生班级：_____

指导教师：_____

完成日期：_____

一、实验目的

熟练利用 PowerDesigner 和任意一种面向对象编程语言实现几种常见的设计模式，包括组合模式、外观模式、代理模式、观看者模式和策略模式，明白得每一种设计模式的模式动机，把握模式结构，学习如何利用代码实现这些模式，并学会分析这些模式的利用成效。

二、实验内容

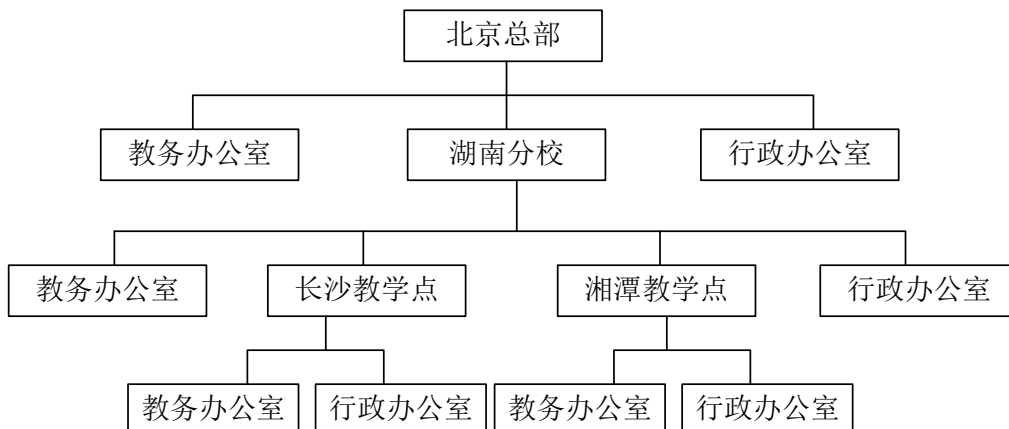
使用 PowerDesigner 和任意一种面向对象编程语言实现组合模式、外观模式、代理模式、观看者模式和策略模式，包括依如实例绘制模式结构图、编写模式实例实现代码，运行并测试模式实例代码。

(1) 组合模式

利用组合模式设计一个杀毒软件(AntiVirus)的框架，该软件既能够对某个文件夹(Folder)杀毒，也能够对某个指定的文件(File)进行杀毒，文件种类包括文本文件 TextFile、图片文件 ImageFile、视频文件 VideoFile。绘制类图并编程模拟实现。

(2) 组合模式

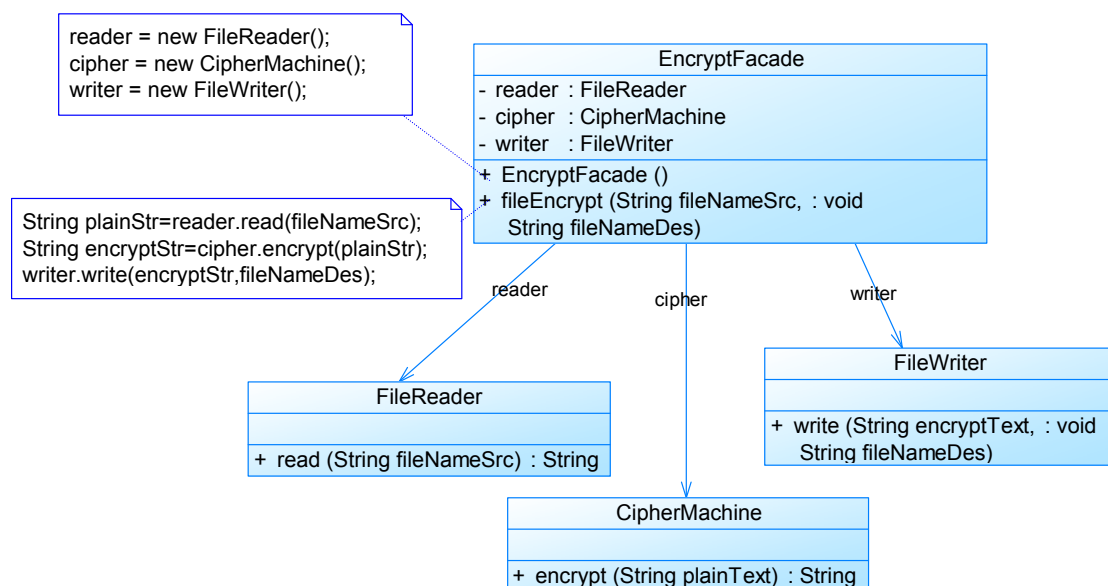
某教育机构组织结构如以下图所示：



在该教育机构的 OA 系统中能够给各级办公室下发公函，现采纳组合模式设计该机构的组织结构，绘制相应的类图并编程模拟实现，在客户端代码中模拟下发公函。（注：能够概念一个办公室类为抽象叶子构件类，再将教务办公室和行政办公室作为其子类；能够概念一个教学机构类为抽象容器构件类，将总部、分校和教学点作为其子类。）

(3) 外观模式

某系统需要提供一个文件加密模块，加密流程包括三个操作，别离是读取源文件、加密、保留加密以后的文件。读取文件和保留文件利用流来实现，这三个操作相对独立，其业务代码封装在三个不同的类中。此刻需要提供一个统一的加密外观类，用户能够直接利用该加密外观类完成文件的读取、加密和保留三个操作，而不需要与每一个类进行交互，利用外观模式设计该加密模块，要求编程模拟实现。参考类图如下：



(4) 外观模式

在电脑主机(Mainframe)中，只需要按下主机的开机按钮(on())，即可挪用其他硬件设备和软件的启动方式，如内存(Memory)的自检(check())，CPU 的运行(run())，硬盘(HardDisk)的读取(read())，操作系统(OS)的载入(load())等，若是某一进程发生错误那么电脑启动失败。利用外观模式模拟该进程，绘制类图并编程模拟实现。

(5) 代理模式

应用软件所提供的桌面快捷方式是快速启动应用程序的代理，桌面快捷方式一样利用一张小图片来表示(Picture)，通过挪用快捷方式的 run()方式将挪用应用软件(Application)的 run()方式。利用代理模式模拟该进程，绘制类图并编程模拟实现。

(6) 代理模式

在某应用软件中需要记录业务方式的挪用日记，在不修改现有业务类的基础上为每一个类提供一个日记记录代理类，在代理类中输出日记，如在业务方式 method()挪用之前输出“方式 method()被挪用，调历时刻为 2020-10-10 10:10:10”，挪用以后若是没有抛异常那么输出“方式 method()成功挪用”，不然输出“方式 method()挪用失败”。在代理类中挪用真实业务类的业务方式，利用代理模式设计该日记记录功能的结构，绘制类图并编程模拟实现。

(7) 观看者模式

某在线股票软件需要提供如下功能：当股票购买者所购买的某支股票价钱转变幅度达到 5%时，系统将自动发送通知（包括新价钱）给购买该股票的股民。现利用观看者模式设计该系统，绘制类图并编

程模拟实现。

(8) 观看者模式

某联机射击游戏能够实时显示队友和仇敌的存活信息，若是有队友或仇敌阵亡，所有在线游戏玩家将收到相应的消息，现提供一个统一的中央角色操纵类(CenterController)来实现消息传递机制，在中央角色操纵器中概念一个集合用于存储所有的用户信息，若是某用户角色(Player)阵亡，那么挪用 CenterController 的通知方式 notify()，该方式将遍历用户信息集合，挪用每一个 Player 的 display()方式显示阵亡信息，若是是队友阵亡，提示信息为：队友****阵亡，我方队员尚有**人；若是是仇敌阵亡，提示信息为：仇敌****阵亡，敌方队员尚有**人。在挪用 notify()方式通知其他用户的同时，阵亡的角色对象将从用户信息集合中删除。

现利用观看者模式设计该消息传递模块，绘制类图并编程模拟实现。(提示：CenterController 充当观看目标，Player 充当观看者，Player 的 die()方式执行时将挪用 CenterController 的 notify()方式，在 notify()方式中挪用其他 Player 对象的 display()方式，并挪用 detach()方式删除阵亡的 Player 对象；能够概念两个角色集合，一个用于存储甲方队员信息，一个用于乙方队员信息；所有方式的参数依如实际需要确信)

(9) 策略模式

设计一个网上书店，该系统中所有的运算机类图书(ComputerBook)每本都有 10%的折扣，所有的语言类图书(LanguageBook)

每本都有 2 元的折扣，小说类图书(NovelBook)每 100 元有 10 元的折扣。现利用策略模式来设计该系统，绘制类图并编程模拟实现。

(10) 策略模式

某系统需要对重要数据（如用户密码）进行加密，并提供了几种加密方案（如凯撒加密、DES 加密等），对该加密模块进行设计，使得用户能够动态选择加密方式。要求绘制类图并编程模拟实现。

三、实验要求

1. 正确无误绘制组合模式、外观模式、代理模式、观看者模式和策略模式实例的模式结构图；
2. 利用任意一种面向对象编程语言实现组合模式、外观模式、代理模式、观看者模式和策略模式实例，代码运行正确无误。

四、实验步骤

1. 熟悉 PowerDesigner15 的经常使用功能；
2. 利用 PowerDesigner 绘制组合模式实例结构图并用面向对象编程语言实现该模式实例；
3. 利用 PowerDesigner 绘制外观模式实例结构图并用面向对象编程语言实现该模式实例；
4. 利用 PowerDesigner 绘制代理模式实例结构图并用面向对象编程语言实现该模式实例；
5. 利用 PowerDesigner 绘制观看者模式实例结构图并用面向对

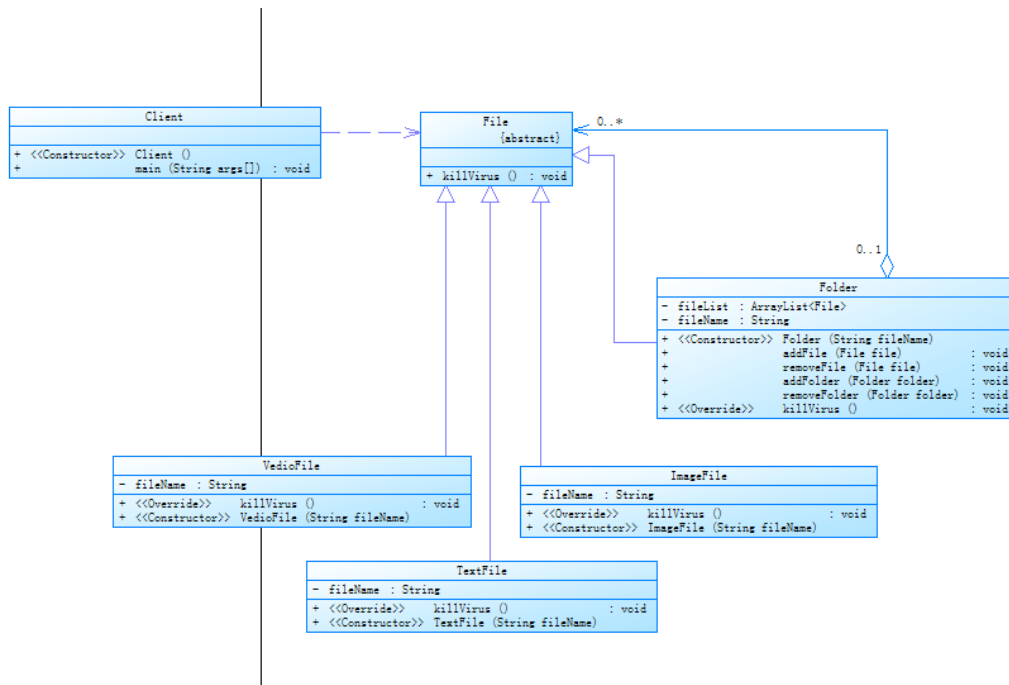
象编程语言实现该模式实例；

6. 利用 PowerDesigner 绘制策略模式结构图并用面向对象编程语言实现该模式。

五、实验结果

(1) 组合模式 1:

类图:



结果:

```

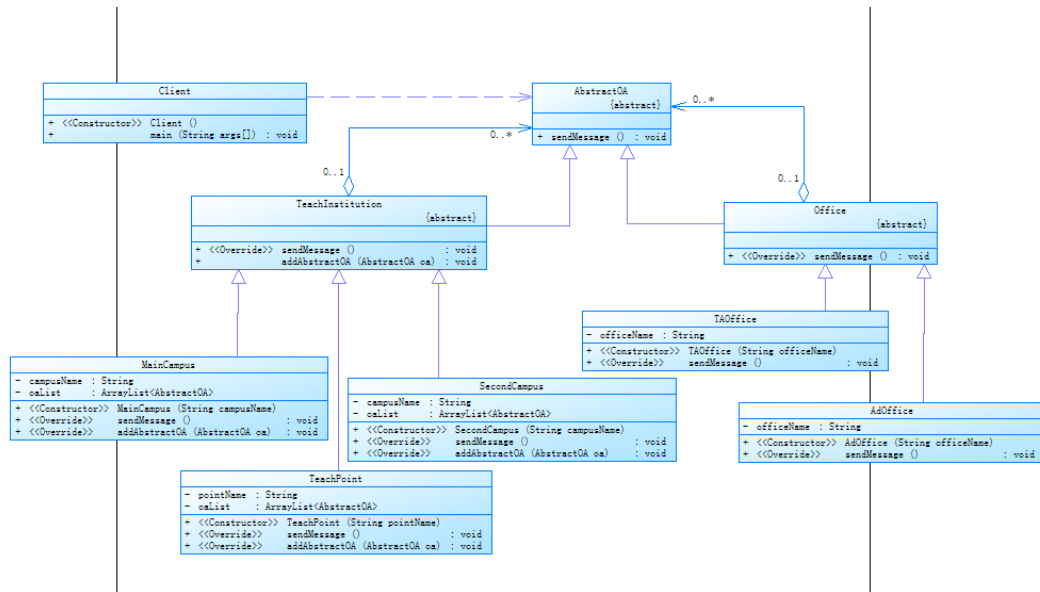
public Client() {
    Folder folder = new Folder("总文件夹");
    File file1 = new ImageFile("图片文件1");
    File file2 = new TextFile("文本文件2");
    File file3 = new VedioFile("视频文件3");
    folder.addFile(file1);
    folder.addFile(file2);
    folder.addFile(file3);
    Folder folder1 = new Folder("子文件夹1");
    File file4 = new ImageFile("图片文件4");
    File file5 = new TextFile("文本文件5");
    File file6 = new VedioFile("视频文件6");
    folder1.addFile(file4);
    folder1.addFile(file5);
    folder1.addFile(file6);
    folder.addFolder(folder1);
    Folder folder2 = new Folder("子文件夹2");
    File file7 = new ImageFile("图片文件7");
    File file8 = new TextFile("文本文件8");
    File file9 = new VedioFile("视频文件9");
    folder1.addFile(file7);
    folder1.addFile(file8);
    folder1.addFile(file9);
    folder1.addFolder(folder2);
    folder.killVirus();
}

```



(2) 组合模式 2:

类图:



结果:

```

public Client(){
    TeachInstitution zongbu = new MainCampus ("北京总部");

    TeachInstitution hunanfenxiao = new SecondCampus ("湖南分校");
    Office hunanof1 = new AdOffice ("湖南行政办公室");
    Office hunanof2 = new TAOffice ("湖南教务办公室");

    Office changshaf1 = new AdOffice ("长沙行政办公室");
    Office changshaof2 = new TAOffice ("长沙教务办公室");
    hunanfenxiao.addAbstractOA (changshaf1);
    hunanfenxiao.addAbstractOA (changshaof2);

    TeachInstitution cspoint = new TeachPoint ("长沙教学点");
    TeachInstitution xtpoint = new TeachPoint ("湘潭教学点");
    hunanfenxiao.addAbstractOA (cspoint);
    hunanfenxiao.addAbstractOA (xtpoint);

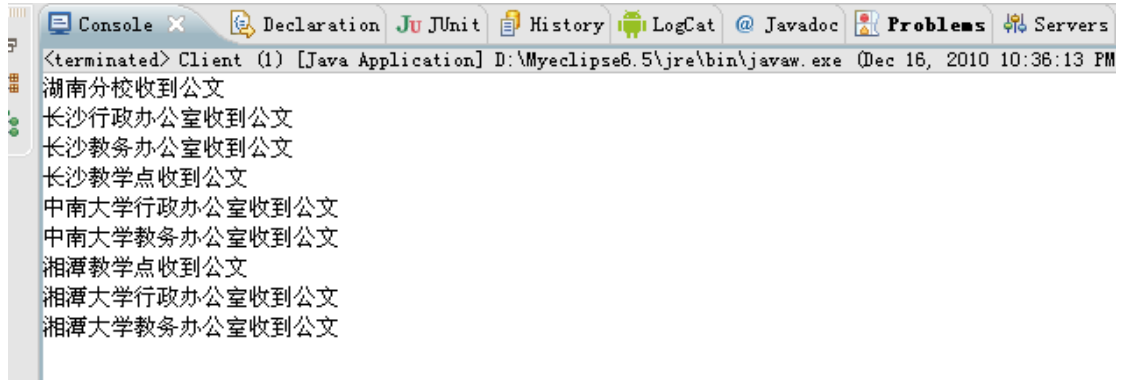
    Office csuof1 = new AdOffice ("中南大学行政办公室");
    Office csuof2 = new TAOffice ("中南大学教务办公室");
    cspoint.addAbstractOA (csuof1);
    cspoint.addAbstractOA (csuof2);

    Office xtdxof1 = new AdOffice ("湘潭大学行政办公室");
    Office xtdxof2 = new TAOffice ("湘潭大学教务办公室");
    xtpoint.addAbstractOA (xtdxof1);
    xtpoint.addAbstractOA (xtdxof2);

    zongbu.addAbstractOA (hunanfenxiao);
    zongbu.addAbstractOA (hunanof1);
    zongbu.addAbstractOA (hunanof2);

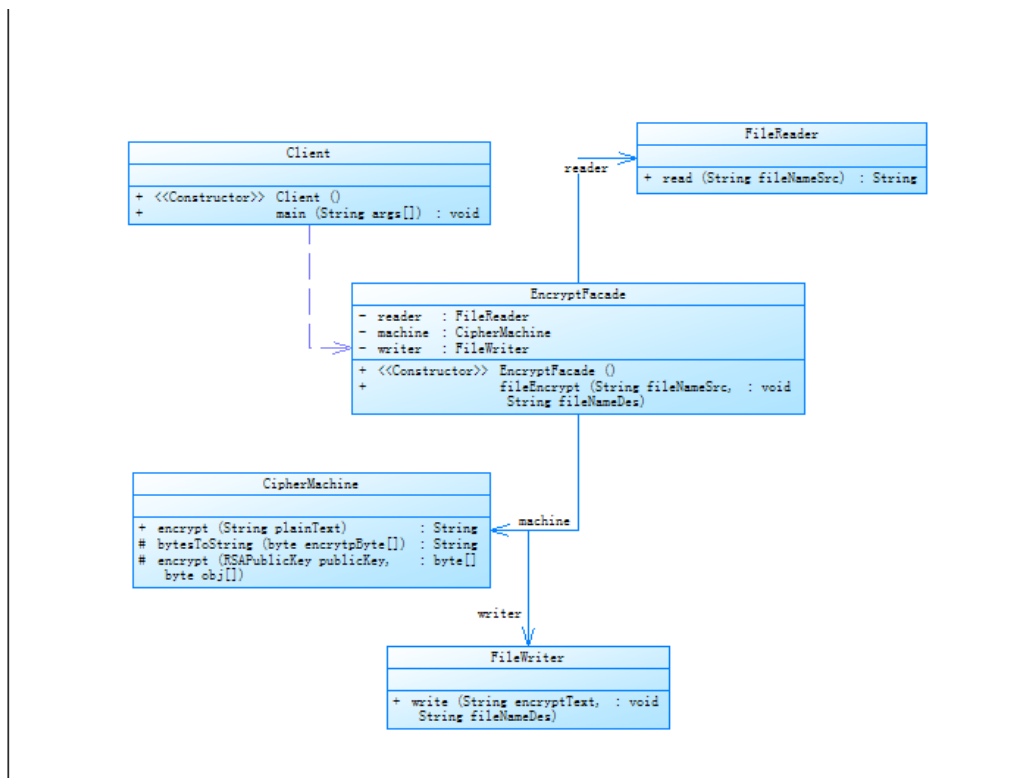
    hunanfenxiao.sendMessage ();
}

```



(3) 外观模式 1

类图:



结果:

```

public Client() {
    EncryptFacade facade = new EncryptFacade();
    facade.fileEncrypt("D:/plp.txt", "D:/test.txt");
}
  
```

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。

如要下载或阅读全文，请访问：

<https://d.book118.com/937135166031006106>