**西北工业大学实验报告**

课程名称: 高级语言程序设计实验

实验课次周次: 第三次（第十一周）

实验地点: 实验大楼B309-1

学生姓名：

学 号:

实验时间: 2024-05-11

提交时间: 2024-05-17

|  |
| --- |
| **成绩评定：（满分为100分）**  **指导教师评语：**  **指导教师签名：** |

目 录

[1 UML实验三 类图 2](#_Toc29242)

[1.1 实验内容 2](#_Toc4994)

[1.2 UML实验结果 6](#_Toc3936)

[2 UML实验四 对象图 8](#_Toc277)

[2.1 实验内容 8](#_Toc11966)

[2.2 UML实验结果 10](#_Toc17750)

[2.3 实验内容： 12](#_Toc27393)

[3 UML实验五 时序图与协作图 13](#_Toc9496)

[3.1 实验内容 13](#_Toc32575)

[3.2 UML结果文件说明 15](#_Toc31349)

[3.3 UML实验结果 16](#_Toc17152)

[4 UML实验六 用例图 17](#_Toc30027)

[4.1 实验内容 17](#_Toc19286)

[4.2 UML结果文件说明 19](#_Toc4195)

[4.3 UML实验结果 20](#_Toc477)

[5 实验总结 21](#_Toc26879)

[5.1 实验中遇到的问题以及解决方法 21](#_Toc18590)

[5.2 实验体会 22](#_Toc21408)

# UML实验三 类图

## 实验内容

### 1.简述类的定义，以及类的三要素

解：类是用于描述具有相同属性和行为的一组对象的模板或蓝图。类定义了对象的结构和行为，并可以用于创建多个具体的对象实例。

类的三要素：1. 属性：类的属性是描述对象特征的数据。它们存储在对象中，并用于表示对象的状态。2. 方法：类的方法是定义在类中的函数，用于描述对象的行为。方法定义了对象可以执行的操作和行为。3. 构造函数：类的构造函数是一种特殊的方法，在创建对象实例时被调用。构造函数用于初始化对象的属性和执行必要的设置。

### **2.**类的属性和方法的可见性有哪些？UML中如何表示？

解：公共：公共属性和方法可以被类的外部代码访问和调用。它们在类的内部和外部都可见。私有：私有属性和方法只能在类的内部访问和调用，对外部代码不可见。私有成员通常用于封装和隐藏类的内部实现细节。保护：保护属性和方法可以在类的内部访问和调用，以及派生类（子类）中访问和调用。默认/包内可见：默认可见性指没有显式指定可见性修饰符时的情况。默认可见性下，属性和方法可以在同一包内的其他类中访问和调用，对于不在同一包内的类不可见。

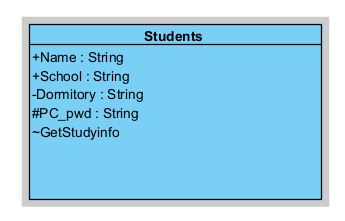
UML中使用符号来表示：

-公共属性和方法用加号 "+" 表示。

-私有属性和方法用减号 "-" 表示。

-保护属性和方法用井号 "#" 表示。

-默认/包内可见的属性和方法用波浪号"~" 表示。类如：



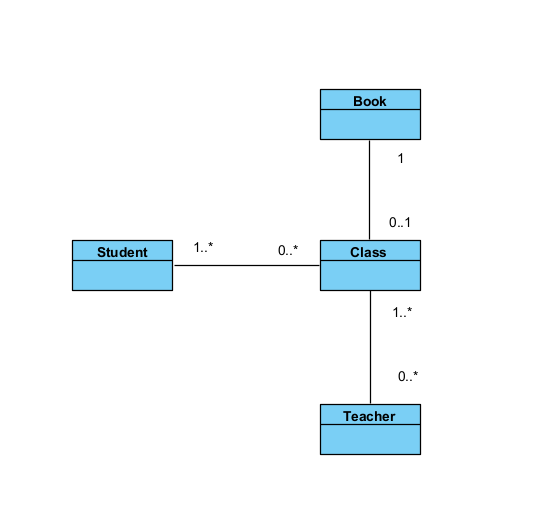
在这个student类中，声明了公共属性姓名和学院，私有属性宿舍，保护属性和方法个人密码以及默认包内可见的获取学籍信息的方法GetStduyinfo。

### **3.**已知三个类A.B和C.其中类A由类B的一个实类和类C的1个或多个实类构成.请画出能够正确表示类A,B和C之间关系的UML类图.

### 解： 如图：

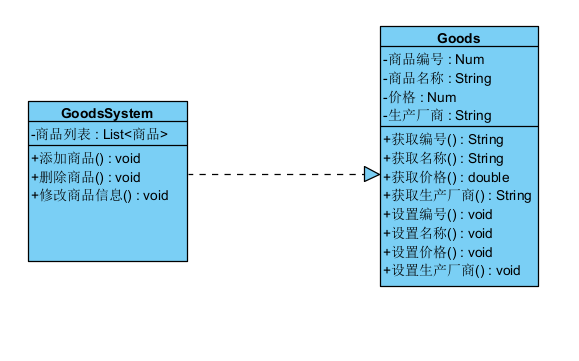
### **4.**根据以下描述画出类图，并注明多重性关系：一个学生可以选修多门课程，也可能没有任何课程；一门课程可以被多个学生选修；一个老师可以教多门课程或者不教课；每门课程至少有一个老师，也可以有多个老师任教；每门课程可以有0或1本教材，每本教材只能用于一门课程。

解：如图所示：



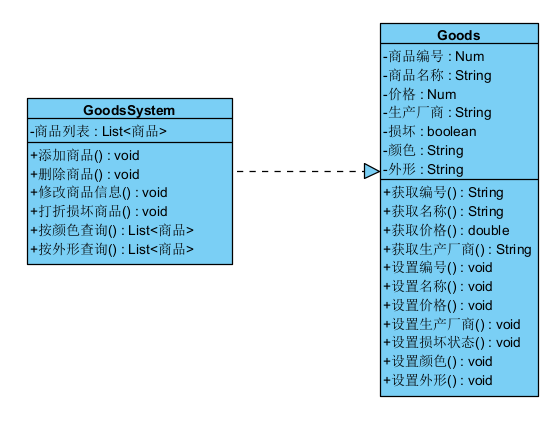
### **5.**现有一系统需要对商品进行管理，包括添加，删除商品，修改商品信息三项功能，画出系统类图。（商品信息包括商品编号，商品名称，价格，生产厂商等）

解：如图：



### **6.**如果现在系统需求发生变化，需要能够对损坏商品进行打折，以及可以按照商品的颜色和外形进行查询，则系统类图应该如何修改？

解：如图：



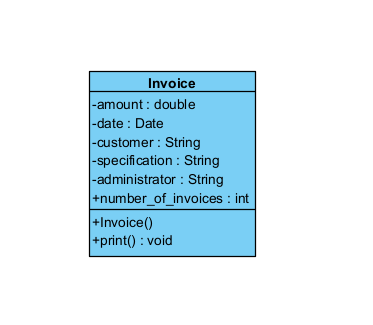
-在商品类中添加了损坏状态、颜色和外形等私有属性。

-在商品管理系统类中添加了打折损坏商品的方法，用于对损坏的商品进行打折操作。

-在商品管理系统类中添加了按颜色查询和按外形查询的方法，用于根据商品的颜色和外形进行查询操作。

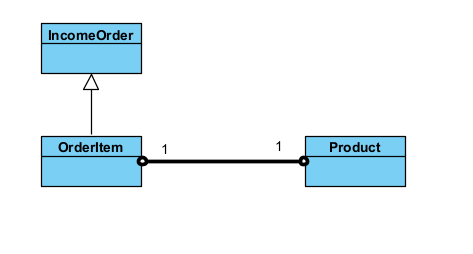
### 7d.根据下面的代码画出Invoice类的类图，要求标明各属性的类型和可见性以及类方法。

解：如图



### 8.下图是一个仓库管理系统的类模型局部，其中IncomeOrder是指入库单，OrderItem是指入库中的每一项，Product则是产品信息。请指出模型中的错误，说明原因并改正类图。

解：个入库单是由多个入库单项（OrderItem）组成的，因此不可能只涉及一个产品，合理的方式应该是入每个入库单项（OrderItem）与产品一对一关联。



## UML实验结果

**[分析与讨论]**

### 1.讨论类图在建模过程中的重要作用。

解：1. 可视化系统结构和关系：类图提供了一种直观的方式来可视化系统的结构和组成部分之间的关系。通过类图，可以清晰地看到类、属性和方法之间的关系，以及它们之间的连接和依赖关系。

2. 定义类的属性和方法： 类图用于定义类的属性和方法，从而确定了系统中的对象的行为和状态。通过类图，可以明确指定每个类应该具有的属性和方法，以及它们的可见性和数据类型。

3. 标识和管理对象之间的关系：类图不仅显示了类的属性和方法，还显示了类之间的关系。这包括关联、继承、聚合、组合等关系。通过类图，可以清楚地标识和管理对象之间的关系，以及它们之间的依赖和交互方式。。

4.支持系统设计和架构决策：类图是系统设计和架构决策的重要工具。通过绘制类图，可以更好地理解系统的结构和功能，并进行重要的设计决策，例如确定类的职责和功能划分、选择合适的设计模式、优化系统性能等。

5.文档和代码生成：类图可以用作生成文档和代码的基础。通过类图，可以生成类和对象的文档，包括属性和方法的说明和文档注释。

### 2.总结在设计绘制类图的过程中应注意哪些问题。

解：准确理解需求，在设计类图之前，确保对系统需求有清晰、准确的理解。选择适当的粒度，在设计类图时，需要选择适当的粒度来表示系统的组成部分。定义清晰的类关系，类之间的关系应该明确而清晰。确保正确使用关联、继承、聚合、组合等关系，以准确表达类之间的依赖和交互方式。考虑可扩展性和重用性，在设计类图时，应考虑系统的可扩展性和重用性。使用适当的设计模式和设计原则来促进模块化和可扩展性，使系统能够灵活地适应变化和扩展。

# UML实验四 对象图

## 实验内容

1. 什么是对象，对象的三大特征是什么？

解：对象图是面向对象建模中的一种图形表示，用于展示系统中对象及其之间的关系和状态。

1.状态：在对象图中，每个对象都有一个状态框，表示对象的当前状态。状态框中通常列出了对象的属性及其对应的值。

2.行为： 对象图中可以使用操作符（如加号或减号）表示对象的行为，即对象能够执行的操作。

3.标识：在对象图中，每个对象都有一个唯一的标识符，通常以对象的名称或引用来表示。通过标识符，可以区分不同的对象，并在对象之间建立关联。对象之间的关联可以通过箭头或线条来表示。

### 2.UML中对象的表示方法有哪些？

解：类的实例 ： 使用类的名称加上下划线和斜杠的形式表示对象。例如，

对象名称： 使用对象的名称来表示对象。对象名称通常是一个独特的标识符，可以用于引用和识别对象

对象图符： 使用类似于类图中的状态框（表示对象的属性）和类名（表示对象的类型）的符号表示对象。

对象图： 对象图是一种特殊类型的UML图，用于展示对象及其之间的关系和状态。对象图中的对象使用对象图符来表示，可以直观地展示对象的属性、类型和关系。

### 3.在下图中最上面的对象的名称是什么？该图表示的意思是什么？请绘制出与其相应的类图

解：答:Country，Province是类，而china、fujian和sichuan 分别是这两

个类中的对象。

1. 说明对象图的适用场景以及它的优缺点

解：

使用场景：

1.系统设计与分析：对象图可以帮助系统设计人员和分析人员更好地理解和描述系统的结构和行为。它们可以用于展示对象及其之间的关系、属性和状态，使得系统的设计和分析更加直观和可视化。

2需求理解和沟通：对象图可以作为需求文档的补充，帮助团队成员更好地理解系统的需求。通过对象图，可以明确对象之间的关系和行为，有助于准确地传达需求，并促进团队成员之间的沟通和协作。

3.软件开发过程中的设计和实现：对象图可以在软件开发过程中用于设计和实现系统。它们可以作为设计阶段的产物，帮助开发人员更好地规划和组织代码结构。同时，对象图也可以用于生成代码或与代码进行对比，确保设计的一致性和正确性。

优点：

1.可视化表示：对象图提供了一种直观和可视化的方式来展示对象及其之间的关系和状态。通过图形表示，人们更容易理解和分析系统的结构和行为。

2.清晰的对象关系：对象图能够清晰地展示对象之间的关系，包括关联、聚合、组合等，有助于理解对象之间的协作和通信方式。

3.易于理解和沟通：对象图使用了简洁而直观的符号和表示方式，使得系统的设计和需求更容易理解和沟通。它们能够促进团队成员之间的交流和理解。

缺点：

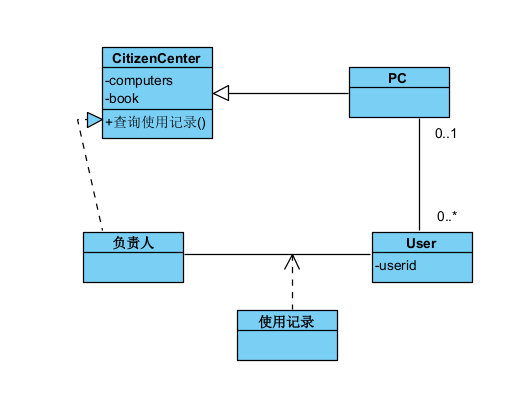
1.局限性：对象图只能展示某一时刻的对象状态和关系，无法完整地表示对象的动态行为和变化过程。为了完整地描述系统的行为，可能需要结合其他UML图形或技术。

2.复杂性管理：对于大型系统，对象图可能变得复杂且庞大，难以管理和维护。设计人员需要注意控制图的规模，并采取适当的分层和组织方式。

3.过度设计风险：对象图可能导致过度设计的风险，即过度关注对象之间的关系和细节，而忽视了系统整体的设计和架构。设计人员需要权衡和平衡各个方面，避免过度设计。

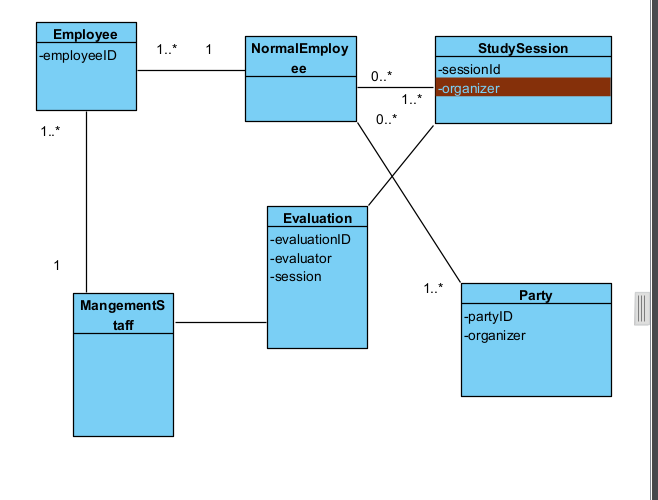
## UML实验结果

实验一：



CitizenCenter类与PersonalComputer类和UsageRecord类之间存在聚合关系。CitizenCenter类管理多台个人电脑和账簿，PersonalComputer类和UsageRecord类与CitizenCenter类相关联。通过这些类和关系，可以实现市民中心对个人电脑使用的管理和记录。

实验二：



**[分析与讨论]**

1.思考类与对象之间的关系是怎样的？各举几个实例，说明哪些是类，哪些是对象。

解：

1. 思考类与对象之间的关系

类 (Class) 是一种抽象的概念,它定义了一类事物的共同特征和行为。而对象 (Object) 则是类的具体实例化,是拥有类定义的属性和方法的实体。例如:类 - 猫；对象 - 小白、小黑、小花；

2.思考在具体场景中应该如何区分类与对象。

解：

1) 类描述的是一类事物的共同特征,而对象则是类的具体实例。

2) 类是抽象的,而对象是具体的。

3) 类是模板,对象是根据类创建出来的实体。

4) 类是静态的,而对象是动态的,可以根据需要创建和销毁。

3.找一段源程序，并绘制出与其相应的对象图。

解：

public class Person {

private String name;

private int age;

public Person(String name, int age) {

this.name = name;

this.age = age;

}

public String getName() {

return name;

}

public int getAge() {

return age;

}

}

public class Main {

public static void main(String[] args) {

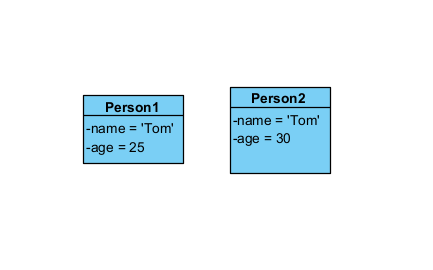
Person person1 = new Person("Tom", 25);

Person person2 = new Person("Alice", 30);

}

}

对应的对象图如下:

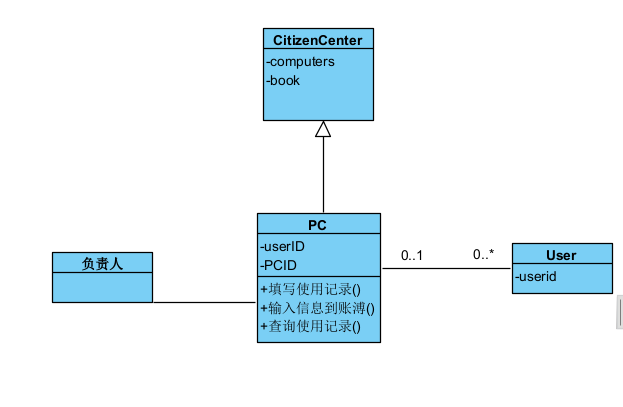


在这个例子中,Person类定义了人的姓名和年龄两个属性,以及对应的 getter 方法。在 Main类的main()方法中,创建了两个Person对象,分别是 person1 和 person2。这两个对象就是 Person 类的实例。

## 实验内容：

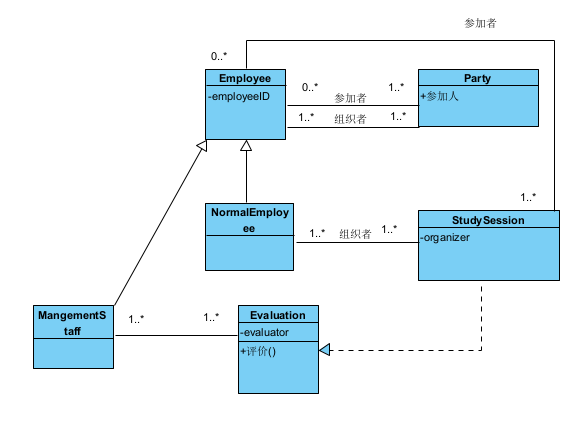
实验一：

解：



实验二：

解：



# UML实验五 时序图与协作图

## 实验内容

### **1.**UML中的时序图与协作图之间各有什么区别和联系？

解：

区别：

- 时序图关注对象之间消息的顺序和时序，强调时间和顺序关系。它展示了对象之间的交互顺序，以及消息是如何在对象之间传递的。

- 协作图关注对象之间的结构和协作关系，强调对象之间的聚合和关联关系。它展示了对象之间的协作结构，以及对象之间的关联和通信方式。

联系：

- 时序图和协作图都可以用于描述对象之间的交互行为，但强调的方面略有不同。

- 时序图和协作图可以相互转换，因为它们描述的是同一个系统或场景，只是从不同的视角进行展示。

### 2.在顺序图和协作图中，应如何表示“循环”，“判断”等结构？

解：循环：在顺序图中，可以使用循环片段来表示循环结构，使用loop关键字标识。在协作图中，可以使用自关联来表示对象与自身之间的循环协作。

判断：在顺序图中，可以使用分支片段来表示判断结构，使用alt关键字标识不同的分支条件。在协作图中，可以使用分支线来表示不同的分支条件。

并发： 在顺序图中，可以使用并发片段来表示并发结构，使用par关键字标识并发执行的消息。在协作图中，可以使用重叠的对象来表示对象之间的并发执行。

**3.什么是消息？消息在交互中扮演什么角色？**

解：消息是交互图中对象之间传递的信息。

角色：

触发行为： 消息触发接收方执行相应的行为或操作。

通信机制： 消息是对象之间进行通信和交互的一种机制，用于传递信息和请求。

传递信息： 消息可以携带数据、参数或信号，用于在对象之间传递信息和数据。

建立关系： 消息的发送和接收可以建立对象之间的关联和关系。

**4.交互图中有哪几种消息？它们之间具体有何分别？应如何表示？**

解：

同步消息： 发送消息后，发送方将等待接收方完成相应的操作或返回结果，然后继续执行。

异步消息： 发送消息后，发送方不会等待接收方的响应，而是继续执行后续操作。

返回消息： 接收方完成相应的操作后，返回结果给发送方。

自消息： 对象向自身发送消息，表示对象内部的交互或自我触发的行为。

在交互图中，可以使用箭头来表示消息的方向和流动，并使用标签或序号来标识消息的类型和内容。

5.假设有一系统得协作图局部如图所示，print方法的功能是将传入的参数的值打印在屏幕上，fact方法是计算阶乘，如果test的初值为1，那么最终将打印出什么？（写出计算过程）

解：

初始值: test = 1

第1次循环: test = 1 + 1 = 2

第2次循环: test = 2 + 2 = 4

第3次循环: test = 4 + 24 = 28

最终打印值: 28

## UML结果文件说明

**[分析与讨论]**

### 1.讨论并总结在建模过程中运用交互图的重要作用及意义

**解：**

1.展示系统中对象之间的动态交互行为:

交互图能清楚地描述系统中各个对象之间的消息传递、时序关系等动态交互过程,有助于理解系统的运行机制。

2.支持系统行为的分析和设计:

交互图为系统行为的分析提供了有价值的信息,有助于发现系统中的错误、性能瓶颈等问题。

3.增强系统可理解性和可维护性:

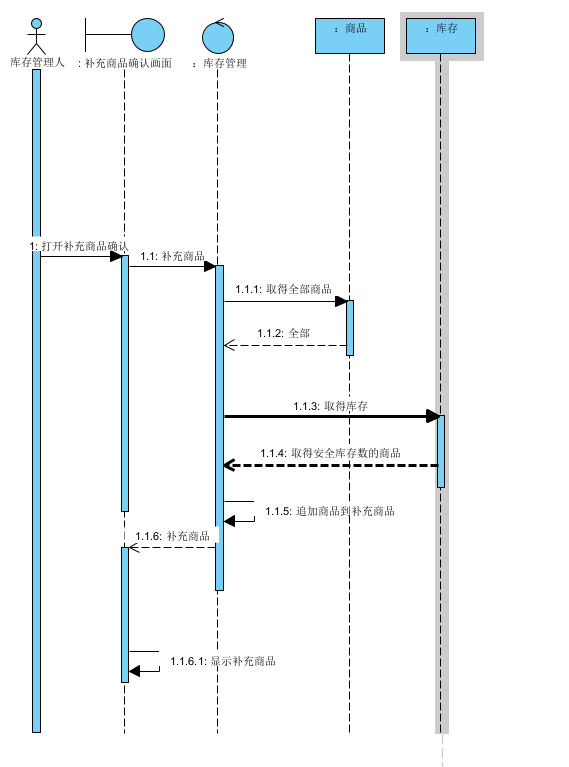
交互图作为系统行为的可视化表示,可以帮助利益相关方更好地理解和沟通系统的工作机制。

4.支持系统建模的迭代和验证:

在系统建模的不同阶段,交互图可以反复修订和完善,有助于验证模型的正确性和完整性。

## UML实验结果

实验：下面是库存补充对象商品的确认处理的说明。参考该说明和实体部分的类图，请画出顺序图。



该顺序图显示了库存补充的整个过程：库存管理人打开补充商品确认画面，然后取得补充商品一览表，在库存管理类中取得全部商品，在库存类中取得库存数和安全库存数，再向补充商品一览表中追加商品。

# UML实验六 用例图

## 实验内容

1.什么叫做状态？状态由哪五个部分组成？

解：状态是对象在某一特定时刻的内部条件或外部表现。状态由以下五个部分组成:

状态名称：描述对象当前所处的内部条件。

状态属性：描述对象在该状态下的特定属性。

状态行为：描述对象在该状态下可执行的特定行为。

进入/退出行为：描述对象进入或退出该状态时需要执行的特定动作。

内部转换：描述对象在该状态下发生的内部事件引起的状态变化。

2.什么是转换？UML中转换的5要素是什么？

解：转换是一个对象从一个状态转换到另一个状态的过程。UML中转换包含以下5个要素:源状态：转换的起点状态。目标状态：转换的终点状态。

触发事件：引发转换的事件。监护条件：决定是否允许转换的条件。

动作：转换时执行的动作。

3.触发事件是什么？可以具体分为哪些事件？

解：触发事件是引发转换的外部刺激或内部条件变化。常见的触发事件包括:

时间事件：在某个时间点发生。

信号事件：接收到某个外部信号。

呼叫事件：调用某个操作。

条件事件：满足某个条件。

错误事件：发生错误。

4.改变事件和监护条件有何异同？

解：改变事件和监护条件的异同如下:

相同点:都是决定转换是否发生的条件。

不同点:

1.改变事件是触发转换的外部事件或内部状态变化。

2.监护条件是决定是否允许转换发生的逻辑条件。

3.改变事件是转换的诱因,监护条件是转换的前提。

5.什么叫做转换？转换可以分为哪几种？

解：转换是对象从一个状态转换到另一个状态的过程。转换可分为以下几种:

外部转换：由外部事件引发的状态变化。

内部转换：由内部事件引发的状态变化。

自发转换：无需任何事件或条件就自动发生的状态变化。

6.状态和对象的关系是怎样的？ 下面描述中哪些是对象，哪些是状态？

支票已付

汽车停在那儿

小王正在工作

解：状态描述了对象在某一特定时刻的内部条件或外部表现。对象是具有状态、行为和标识的实体。

在给出的描述中:

"支票已付"是状态,描述了支票的付款状态。

"汽车停在那儿"是状态,描述了汽车的停车状态。

"小王正在工作"是状态,描述了小王的工作状态。

其中"支票"、"汽车"和"小王"是对象,分别具有自己的状态。

### **7.**说出下面状态的状态名和各种转换、事件。

解： Lighting 状态的生命周期和行为

进入状态时执行 turnOn。

在状态期间执行 blinkFiveTimes。

在特定事件发生时（例如 eventPowerOff 或 eventSelfTest），执行相应的动作（例如 powerSupplySelf 或 defer）。

退出状态时执行 turnOff。

### **8.说出下面状态图所表达的信息**。

解：这是一个烧水壶状态图：  
初始状态Off：如果没有水执行turnOn会自己切换回turnOff；

如果有水会切换到On烧水

在On状态下：水开了会turnOff，一直运行会烧坏。

### 9.说出下面状态图所表达的信息，并指出蓝色部分代表的含义。

解：这是一个课程评价流程：

开始第一步注册课程，首先需要完成实验一：蓝色部分代表：实验一完成成功退出lab界面后进入实验2；实验2完成后第一个流程结束；

第二个流程通过【lab】进入团队项目，项目完成后结束；

第三个流程进行考试，考试及格后结束得到课程通过，总流程结束；如果考试不及格会显示课程不通过需要重新进行补考。

## UML结果文件说明

**[分析与讨论]**

### 1.讨论思考状态图在UML建模中可以起到什么样的作用

(1) 描述系统对象在不同状态下的行为和转换。状态图能够清晰地展示对象在各种情况下的状态变迁过程,有助于理解系统的动态特性。

(2) 捕捉系统的复杂动态行为。对于存在复杂状态变化的系统,状态图能够更好地描述和分析其内部状态转换的逻辑。

(3) 指导设计和实现。状态图可以为系统的设计和实现提供蓝图,帮助开发人员更好地理解系统的行为并指导具体的编码工作。

(4) 支持需求分析。状态图能够帮助分析师更好地理解和定义系统的行为需求。

### 2.思考什么样的情况下适合引入状态图进行建模。

(1) 系统存在复杂的状态变化和转换逻辑。如果系统的对象在不同情况下会经历多种状态转换,使用状态图能够更好地描述和分析这些动态行为。

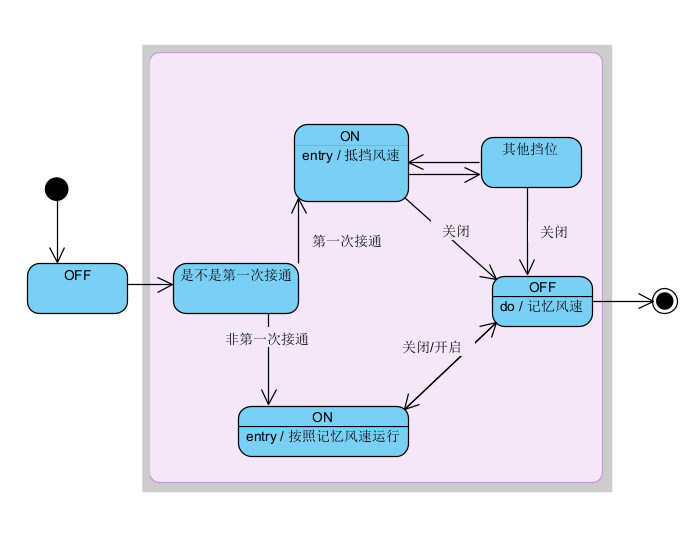
(2) 系统需要处理事件驱动的交互。状态图擅长描述事件触发下的状态变迁过程,适合于建模事件驱动型系统。

(3) 系统具有较强的实时性和响应性需求。状态图能够清晰地表达系统在不同状态下的响应行为,有助于满足实时性需求。

(4) 系统需要实现复杂的业务逻辑。状态图能够帮助分析师更好地理解和设计系统的核心业务逻辑。

## UML实验结果

实验：画出状态图。某电扇产品通过ON/OFF开关来控制电源接入，风速有高、中、低三档。开关置为ON时电扇开启，置为OFF时电扇停止。第一次接通电源默认以低档风速运转，风扇停止时记忆当时的风速，再次开启风扇时直接按照上次的风速运转。

解：如图  


# 实验总结

## 实验中遇到的问题以及解决方法

1.问题：实现关系不在类图的一级面板里

解决方法：与同学交流软件使用心得，在泛化-实现关系中找到

2.问题：流程图相关知识点不清楚

解决方法：询问老师助教解答

3.问题：对象图的关联创建标注不懂

解决方法：自己在CSDN上学习有关知识点，答疑解惑

## 实验体会

通过这4次实验练习,我对UML建模有了更全面和深入的理解。掌握了UML类图、对象图、时序图和协作图等常用建模工具的基本概念和表示方法,并学会如何根据系统需求选择合适的图形进行建模。在绘制UML图时,需要注意各种类之间的关系,以及如何使用合适的符号和多重性来表示,这对于准确描述系统的静态结构很重要。对象图则侧重于描述系统某一时刻的状态,有助于理解对象的属性和行为。而时序图和协作图则重点关注对象之间的交互动态,有利于分析系统的行为。通过实践,我深刻认识到UML建模在系统分析和设计中的重要意义。它不仅能帮助开发人员更好地理解系统需求,提高设计的可复用性和可维护性,而且也为团队成员之间的沟通和协作提供了一个共享的语言和视角。

总的来说,这些实验让我掌握了UML建模的基本方法和技能,为今后的软件工程实践奠定了良好的基础。我将继续深入学习和应用UML技术,提高自己的建模能力,为未来的软件项目贡献自己的力量。