类之间的关系

依赖 users– a

是一种最明显的、最常见的关系。如果一个类的方法操作另一个类的对象，我们就说一个类依赖于另一个类。

尽可能地将相互依赖的类减至最少。如果类A不知道B的存在，它就不会关心B的任何改变（这意味着B的改变不会导致A产生任何bug）。让类之间的耦合度最小。

聚合 has – a

是一种具体且易于理解的关系。一个Order对象包含一些Item对象。聚合关系意味着类A的对象包含类B的对象

继承 is – a

是一种用于表示特殊与一般关系的。一般而言，如果类A扩展类B，类A不但包含从类B继承的方法，还会拥有一些额外的功能。

一个对象变量并没有实际包含一个对象，而仅仅引用一个对象

GregorianCalendar Date类区别

java.util. GregorianCalendar 1.1

GregorianCalendar()

构造一个日历对象，用来表示默认地区、默认时区和当前时间

GregorianCalendar(int year,int month,int day)

GregorianCalendar(int year,int month,int day,int hour,int minutes,int seconds)

用给定的日期和时间构造一个Gregorian日历对象。

参数： year 该日期所在的年份

month 该日期所在的月份，以0位基准，0表示一月

day 该月份中的日期

hour 小时(0到23之间)

minutes 分钟(0到59之间)

seconds 秒(0到59之间)

int get(int field)

返回给定域的值

参数： field 可以是下述选项之一：

void set(int field,int value)  
设置特定域的值

参数 filed get接受的常量之一

value 新值

void set(int year,int month,int day)

void set(int year,int month,int day,int hour,int minutes,int seconds)

将日期域和时间域设置为新值

参数： year 该日期所在的年份

month 该日期所在的月份，以0位基准，0表示一月

day 该月份中的日期

hour 小时(0到23之间)

minutes 分钟(0到59之间)

seconds 秒(0到59之间)

void add(int field,int amount)

这是一个可以对日期信息实施算术运算的方法。对给定的时间域增加指定数量的时间。

例如，可以通过调用c.add(Calendar.DAY\_OF\_MONTH.7)将当前的日历日期加上7

参数： field 需要修改的域(可以使用get方法文档中给出的一个常量)

amount 域被改变数量(可以为负值)

int getFirstDayOfWeek()

获得当前用户所在地区，一个星期中的第一天。例如，在美国一个星期中的第一天是Calendar.SUNDAY.

void setTime(Date time)

将日历设置为指定的时间点

参数：time 时间点

Date getTime()

获得这个日历对象当前值所表达的时间点

java.text.DateFormatSymbols 1.1

String[] getShortWeekdays()

String[] getShortMonths()

String[] getWeekdays()

String[] getMonths()

获得当前地区的星期几或月份的名称。利用Calendar的星期和月份常量作为数组索引值。

构造器与类同名

每个类可以有一个以上的构造器

构造器可以有0个、1个、或多个参数

构造器没有返回值

构造器总是伴随着new操作一起调用

每一个方法中，关键字this表示隐式参数。

位于方法名后面括号中的数值，这是一个显式参数。

一个私有的数据域

一个公有的域访问器方法

一个公有的域更改器方法

可以改变内部实现，除了该类的方法外，不会影响其他代码

更改器方法可以执行错误检查，然而直接对域进行赋值将不会进行这些处理