### 控制台提示用户输入一个字符串，过滤重复字符。效果如下：

|  |
| --- |
| 请输入一个字符串：abccbefagfdfefea  过滤后的字符串是：abcefgd |

### 控制台提示用户输入一个字符串，判断字符串左右两边是否对称。效果如下：

|  |
| --- |
| 请输入一个字符串：abccba  abccba是对称字符串  请输入一个字符串：abcdecba  abcdecba不是对称字符串 |

### 使用面向对象实现以下功能：

控制台提示用户输入一个年份。使用方法判断该年份是否为闰年。

|  |
| --- |
| 请输入年份：2016  2009不是闰年 |

提示：

（1）能被4整除而不能被100整除，或者，能被400整除 是闰年

（2）定义一类，类中定义一个方法，要求方法能够接受参数，将键盘录入的年份传递进方法中，判断完平年和闰年后，返回结果，闰年返回true，不是闰年返回false;

### 4、定义一个方法判断用户输入的数字是否是水仙花数。

提示："水仙花数"是指一个三位数，其各位数字立方和等于该数本身。比如：153是一个"水仙花数"，因为153=1的三次方加上5的三次方加上3的三次方。

### 5、编写方法计算斐波拉契数列中前10个数字的和

提示：斐波拉契数列形如：1,1,2,3,5,8,13,21,34,.......

### 6、自行车类:

属性:轮子个数,轮子颜色

行为:跑(输出语句模拟:”瞪着跑”,输出语句中要有自行车的属性)

电动车类:

属性:轮子个数,轮子颜色,电池(布尔类型:真代表有电,假代表没电)

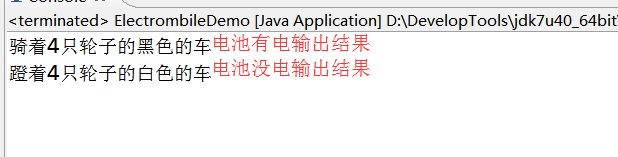
行为:跑(如果电池有电就骑着跑,如果电池没电只能瞪着跑)

注意:骑着跑用输出语句模拟,瞪着跑需要调用自行车类的跑方法

测试类:

创建一个电动车对象,为电动车各项属性赋值,电动车电池有电,调用跑方法

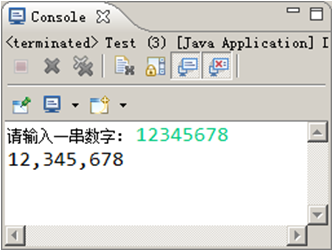
修改电池属性的值,使电动车电池没电,再次调用跑方法



注意:除了测试类都需要有相应的构造方法与getter/setter方法

### 需求说明：使用StringBuffer类的length()和insert()方法实现

将一个数字字符串转换成逗号分隔的数字串，即从右边开始每三个数字用逗号分隔，输出效果如图所示。



### 编程实现迷你计算器功能，

支持“+”“-”“\*”“/”，从控制台输入两个操作数，输出运算结果。如下所示：

|  |
| --- |
| 请输入第一个操作数：  5  请输入第二个操作数：  6  请输入运算符（+ - \* /）：  \*  计算结果：5\*6=30 |

|  |
| --- |
| 请输入第一个操作数：  5  请输入第二个操作数：  a  请输入正确的数字！ |

### 小明左手、右手中分别拿两张纸牌：

黑桃10和红桃8，现在交换手中的牌。用程序模拟这一过程：两个整数分别保存在两个变量中，将这两个变量的值呼唤，并输出互换后的结果。程序运行结果如图：

|  |
| --- |
| 输出互换前手中的纸牌：  左手中的纸牌：10  右手中的纸牌：8  输出互换后手中的纸牌：  左手中的纸牌：8  右手中的纸牌：10 |

### 需求说明：采用面向接口编程思想组装一台计算机

计算机的主要组成部分有：CPU、硬盘、内存：首先，定义CPU的接口CPU，返回CPU品牌和主频；其次，定义内存的接口EMS，返回容量；再次，定义硬盘的接口HardDisk，返回容量；然后，编写各组件厂商分别实现CPU、EMS、和HardDisk接口，编写计算机类，组装计算机并显示相关信息；最后，编写测试类运行如图所示的效果。



### 现有电视商品价格竞猜活动

活动的规则：随机出现一个商品名，用户猜测它的价值，如果在规定次数内猜对，便可获得此商品。模拟竞猜活动，运行结果如下所示：

|  |
| --- |
| 请猜测“公主电动车”的价格：1000  再大点！  再猜一次吧：5000  再小点！  再猜一次吧：2500  在小点！  再猜一次吧：2200  4次内没有猜对，下次努力吧！ |

|  |
| --- |
| 参考实现步骤如下：   * 定义类QuessMachine，编写它的initial()方法预定义商品信息，根据产生的随机数字，选定一款竞猜的商品 * 编写QuessMachine类的guess()方法，如果猜测正确，返回“猜对了！”；如果偏大，返回“再小些”；如果偏小，返回“再大些！” * 编写测试类模拟竞猜 |

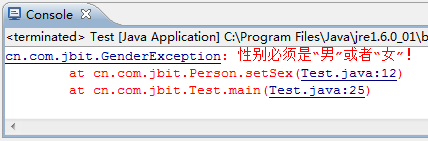
### 12、编码创建一个打印机类Printer

定义抽象方法print()；创建两个子类，即针式打印机类DotMatrixPrinter和喷墨打印机类InkpetPrinter，并在各自类中重写方法print()，编写测试类实现两种打印机；再添加一个激光打印机子类LaserPrinter，重写方法print()，修改测试类实现该打印机打印。

要求使用：抽象类实现

### 13、JDK 中的异常类型不能满足程序的需要时，可以自定义异常类。

请自定义异常GenderException类，继承Throwable类、继承Excepion 或者RuntimeException，并使用GenderException对性别的错误输入进行异常捕捉，实现如下图所示的效果。

****

### 14、使用OOP思想，实现以下需求：

1. 假设用户名和密码均为admin，如果输入正确，控制台输出：欢迎admin登录
2. 如果输入错误，控制台输出：录入错误,您还有2次机会（最多有3次机会）

### 15、使用封装实现企鹅类

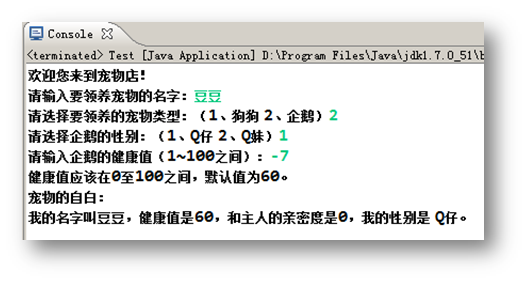
需求说明：按照封装的步骤：

（1）修改属性的可见性；

（2）创建公有的getter/setter方法；

（3）在getter/setter方法中加入属性控制语句；

正确使用封装实现电子宠物系统的企鹅类，把企鹅封装起来，对外提供getter和setter方法，并在方法中对容易误操作的属性进行赋初值和判断：把正确输入健康值和亲密度：保证健康值的有效性（0-100），否则取默认值60；保证亲密度的有效性（0-100），否则取默认值60，该程序的输出效果如图所示。



### 16、需求说明：按照向下转型的注意事项：

（1）<子类型> <引用变量名> = (<子类型> )<父类型的引用变量>；

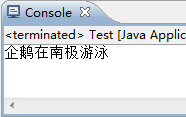
（2）在向下转型的过程中，如果没有转换为真实子类类型，会出现类型转换异常；

为狗狗添加叼飞碟的方法,在测试类中调用该方法。正确使用向下转型，即父类类型转换为子类类型，输出如图所示的效果。该程序中具体的写法为Dog dog=(Dog)pet;



### 17、需求说明：狗具有特有的接飞盘方法，企鹅具有特有的南极游泳方法。

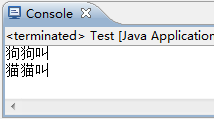
正确使用instanceof判断宠物类型，编写测试类分别调用不同宠物的特有方法输出下图所示企鹅的南极游泳效果。（注意：使用instanceof时，对象的类型必须和instanceof后面的参数所指定的类在继承上有上下级关系）



### 18、需求说明：按照多态的主要应用形式：

1. 使用父类作为方法的形参，是Java中实现和使用多态的主要方式；
2. 使用父类作为方法的返回值，也是Java中实现和使用多态的主要方式；

主人类（Host类）中的赠送动物（donateAnimal()）方法的返回值类型是父类类型。在测试类中调用该方法时根据传入不同的动物类型字符串参数，返回不同的动物对象，并调用动物“叫”的方法。输出如图所示的效果。

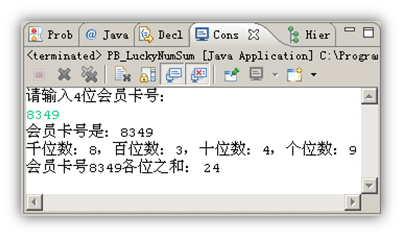


### 19、使用IO技术。实现.class文件的复制。

### 20、分别使用List集合、Set集合保存10个随机数。并在控制台输出。

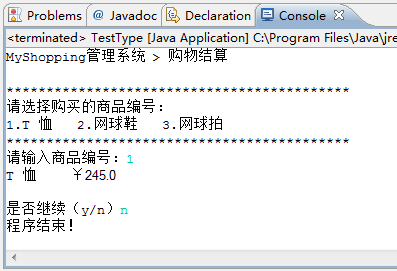
### 21、需求说明：使用Scanner获得键盘输入四位数字的会员卡号

分别使用“/”和“%”运算符分解获得会员卡各位上的数字，最后将各位上数字求和并输出，输出效果如图所示。



### 22、需求说明：正确使用while循环

根据“是否继续”和“商品编号”两个条件，反复查询所选商品的价格，显示效果如下图所示。



### 23、需求说明：冒泡排序时注意以下关键点：

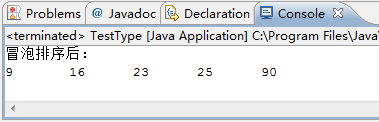
（1）5个数字如何存放：（数组，数组.length = 5）

（2）控制比较多少轮：（外层循环，循环变量 i）

（3）控制每轮比较多少次：（内层循环，循环变量 j）

（4）交换数据；

使用二重循环实现如图所示的冒泡排序。



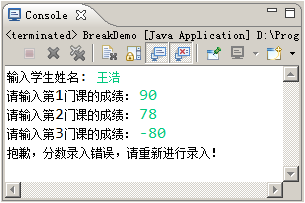
### 24、需求说明：根据3个班各5名学生某门课程的成绩，正确使用二维数组计算

如图所示3个班各自的总成绩。在这个程序中二维数组的第一维是班级，第二维是各班级的学生，与之对应的双重循环中，外层循环i为班级，内层循环j为各班级的学生。（注意：二维数组实际上是一个一维数组，它的每个元素又是一个一维数组）



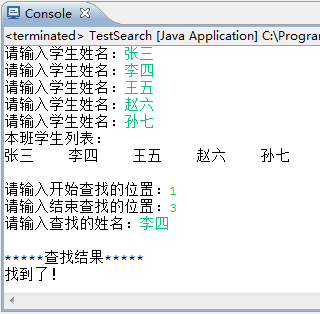
### 25、需求说明：循环录入某学生5门课的成绩并计算平均分，

如果某分数录入为负，使用break语句停止录入，并提示录入错误，实现效果如下图所示。



### 26、需求说明：正确使用带参方法实现学员信息管理

增加学员姓名；在保存了多个学生姓名的数组中，指定查找区间，查找某个学生姓名并显示是否查找成功，程序输出结果如图所示。



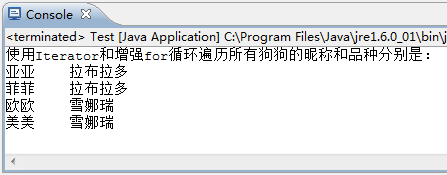
### 27、需求说明：按照以下实现的步骤，使用Iterator和增强型for循环遍历Map集合，输出效果如下图所示：

（1）创建多个狗狗对象；

（2）创建Map集合对象并把多个狗狗对象放入其中；

（3）通过Iterator依次输出集合中所有狗狗的信息；

（4）使用增强型for循环输出集合中所有狗狗的信息；



### 28、动物类:

属性:动物名字,毛颜色,腿个数；

行为:吃饭。

猫类：

属性：动物名字,毛的颜色,腿的个数

行为：猫吃鱼

猫特有行为：抓老鼠catchMouse

狗类:

属性: 动物名字,毛的颜色,腿的个数

行为:狗啃骨头

狗特有行为：看家lookHome

测试类:

用多态的形式,调用猫和狗的共同行为,

利用向下转型调用猫和狗的特有行为

打印结果:



### 29、需求说明：

已知三个类分别是员工类,软件开发工程师类,讲师类,三个类中的成员如下

1. 员工类:

属性:

姓名,年龄

行为:

吃饭方法(方法中打印:人都要吃饭)

2. 讲师类:是员工类的一种

属性:

含有和员工类相同的属性:姓名,年龄,薪资

行为:

教课(方法中打印:"xx岁的xx在教java",

注意:教课方法中打印的年龄和姓名需要获取,不能直接打印这句话:"xx岁的xx在教java")

3.软件开发工程师类:是员工类的一种

属性:

含有和员工类相同的属性:姓名,年龄

行为:

工作的方法：(方法中打印:"xx岁的xx在努力敲代码",

注意:方法中打印的年龄和姓名需要获取,不能直接打印这句话:"xx岁的xx在努力敲代码")

要求如下:

a.按照上面的描述定义出员工类,讲师类,软件开发工程师类

b.利用继承来简化代码

c.所有的属性必须用私有修饰,利用Eclipse生成每个属性对应的get和set方法

d.利用Eclipse生成每个类的空参构造函数

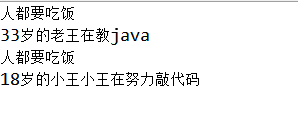
利用Eclipse生成每个类的含有满参构造函数

在讲师类和学生类的满参构造方法中必须用上super语句

e.定义一个测试类名称为TestDemo,在测试类中

利用讲师类的满参构造函数,创建一个老师对象,这个老师的姓名为老王,年龄为33,调用吃饭方法和教课方法

利用软件开发工程师类的满参构造函数,创建一个工程师对象,这个工程师的姓名为小王,年龄为18,调用吃饭方法和工作方法

最终控制台打印结果  


### 30、需求说明：学士后学员应聘至外企工作，每个学员都会有一个英文名称，对应该学员对象。

按照以下实现的步骤实现通过英文名称，获得该学员对象的详细信息：

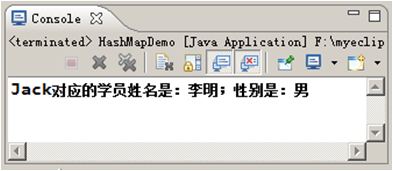
（1）创建学员对象，学员属性包括姓名、性别；

（2）创建保存“键-值对”的集合对象；

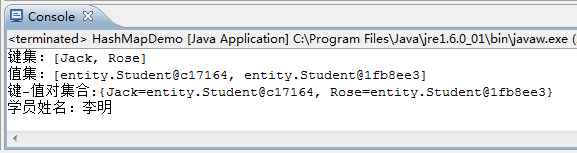
（3）把英文名称与学员对象按照“键-值对”的方式存储在HashMap中，这里以“Jack”举例；

（4）判断是否存在”Jack”这个键；

（5）如果存在，根据键获取如图所示的相应的值；



修改程序使用泛型HashMap<K,V>存储学员数据，使用containsKey()方法判断值的存在，如果存在按照下图效果进行输出。



### 需求说明：

已知用户购买的商品名称如下:

铅笔

本子

橡皮擦

铅笔

铅笔

本子

本子

本子

分析：

1. 请使用一个集合(使用泛型)存储用户购买的上述商品名称
   1. List<商品类>
   2. 商品类：商品名称 全参构造方法 equals()

b.利用迭代器方式遍历出上述集合

c.利用HashMap统计出每件商品对应的购买数量

Map<商品,Integer>

d.利用增强for循环遍历上述的HashMap

|  |
| --- |
| 商品名称 数量  铅笔 3  本子 4  橡皮擦 1 |

### 32、需求说明：

根据需求完成以下功能

a.已知在当前工程的根目录下有一个1.txt,2.txt中的内容为:(可以在1.txt中手动输入以下内容)

"abc"

"def"  
"aaabaaa"

"aba"

"pop"

b.利用 字符数组,FileReader,FileWriter将1.txt拷贝到当前工程下的2.txt中,异常不需要处理,抛出即可(throws)

c.利用 BufferedReader和BufferedWriter将1.txt拷贝到当前工程下的3.txt中,并处理异常

d.打印出字符串中包含"a"字符串的原始字符串

e.统计出包含"a"的字符串的个数

最终控制台打印结果:

