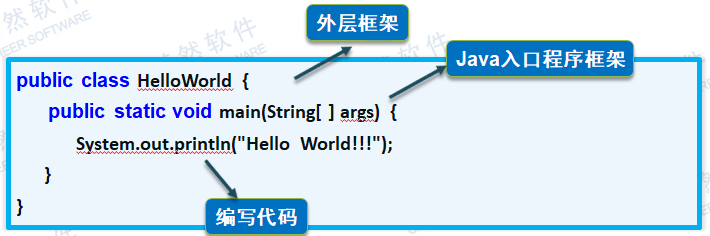
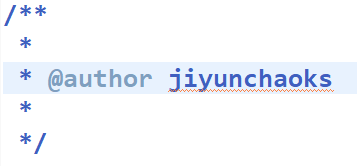
**总结**

**(1)java入门**

1. **程序的概念:** 程序是为实现特定目标或解决特定问题而用计算机语言编写命令序列的集合。为实现预期目的而进行操作的一系列语句和指令
2. **Java语言特点:**面向对象,健壮性,跨平台性
3. **Java环境搭建:** JAVA\_HOME：配置JDK安装路径 Path：配置JDK命令文件的地址，也就是bin目录地址（ %JAVA\_HOME%\bin;） CLASSPATH：配置类库文件地址（.;%JAVA\_HOME%\lib\dt.jar;%JAVA\_HOME%\lib\tools.jar）
4. **开发Java程序的步骤:**编写代码,编译,运行
5. **Java程序的基本框架:**



1. **Java的注释类型:单行注释和多行注释**



1. **常用dos命令:** dir（directory : 列出当前目录下的文件以及文件夹

md（make directory）: 创建目录

rd（remove directory） : 删除目录

cd（change directory）: 进入指定目录

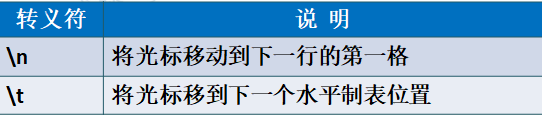
cd.. : 退回到上一级目录

cd \: 退回到根目录

exit : 退出 dos 命令行

echo: 打印输出命令

1. **JVM**是一个虚拟的计算机，具有指令集并使用不同的存储区域。负责执行指令，管理数据、内存、寄存器.
2. **核心机制—垃圾回收:** Java 不需要程序员负责回收无用的内存：它提供一种系统级线程跟踪存储空间的分配情况。并在JVM空闲时，检查并释放那些可被释放的存储空间。
3. **JDK(Java Development KitJava开发工具包)：**JDK是提供给 Java 开发人员使用的，其中包含了 java 的开发工具，也包括了**JRE。**所以安装了JDK，就不用在单独安装JRE了
4. **JRE(Java Runtime Environment Java运行环境) ：**包括Java虚拟机(JVM：Java Virtual Machine)和 Java 程序所需的核心类库等，如果想要运行一个开发好的Java程序，计算机中只需要安装JRE即可。
5. **转义符:**



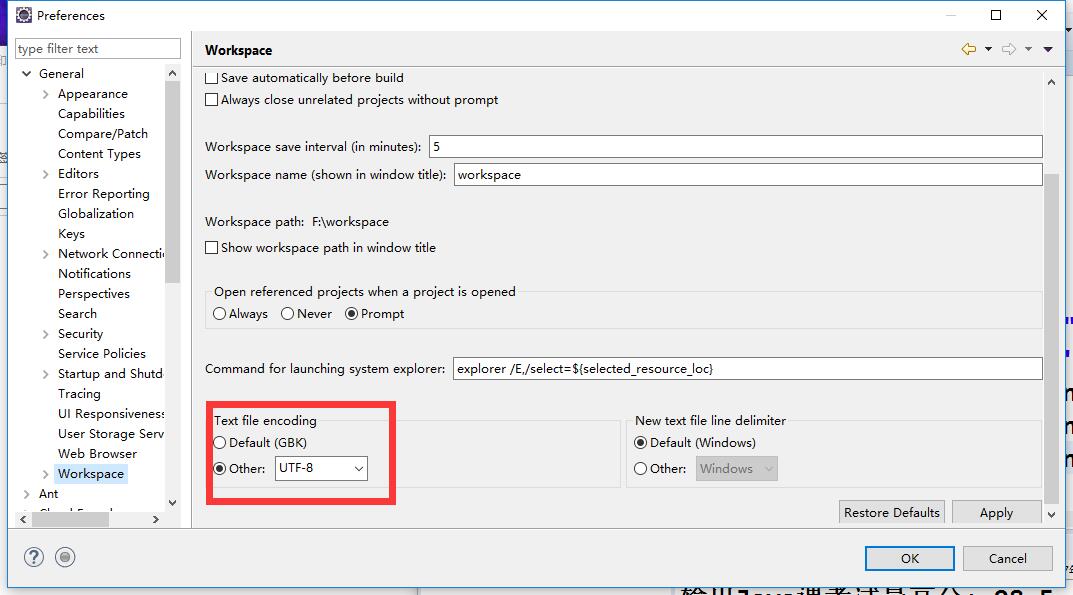
1. **编码规范:**



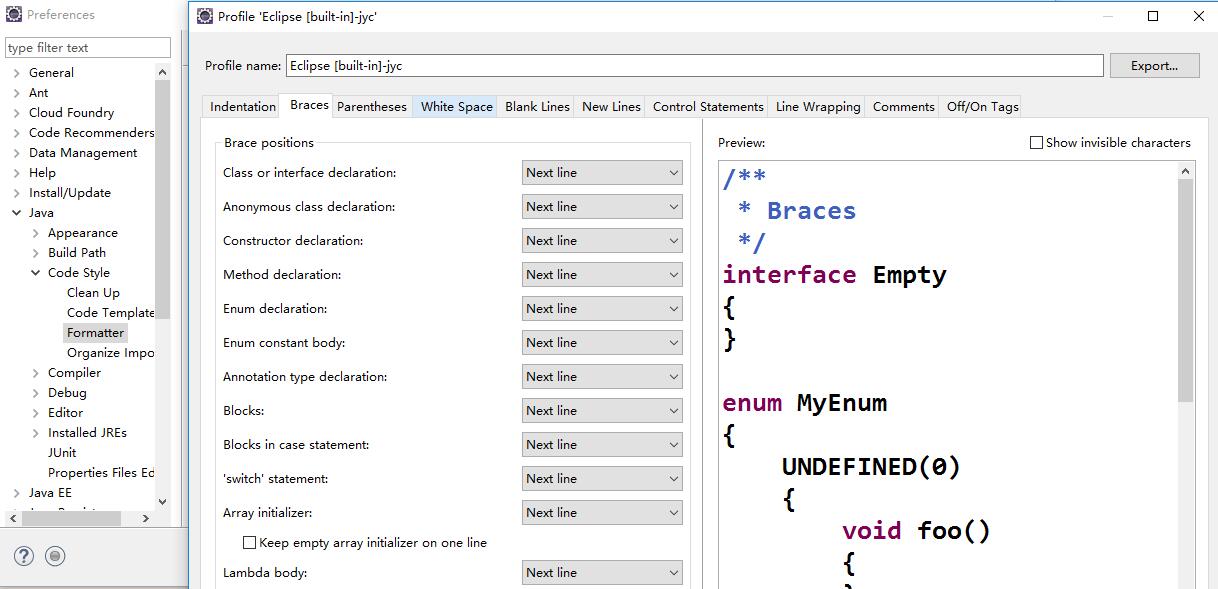
1. **使用Eclipse开发程序步骤:**



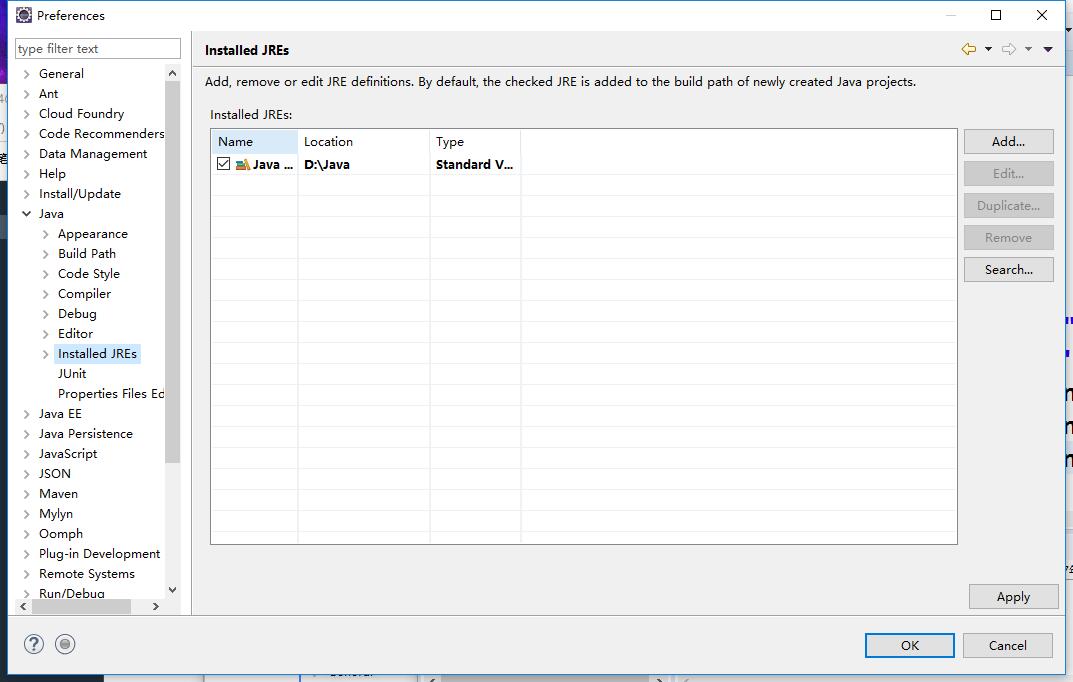
1. **初次运行Eclipse需进行的相关配置:**

****

(改编码)

****

**改代码格式**

****

**改jre位置**

**(2)变量,数据类型,运算符**

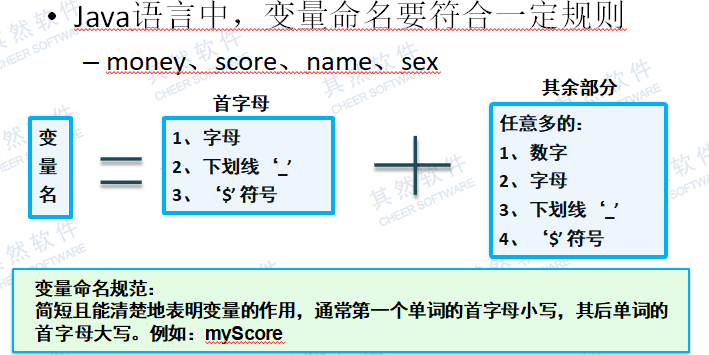
**1) 变量的定义:**在执行过程中.其值被改变的量成为变量

**2) 常用数据类型:整型(byte,short,long,int) 浮点型(float,double) 字符型(char) 布尔型(boolean)**

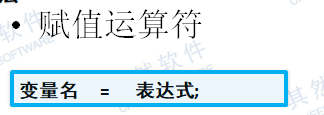
**3) 变量声明和使用:**



**4)变量名命名规则:**



**5)赋值运算符:**



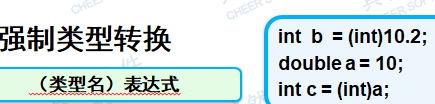
**6)算术运算符:+ - \* / % 自增自减:++a和a++的区别:前者先加后赋值,后者先加,然后赋原值**

**7)比较运算符:> < == >= <= !=**

**8)逻辑运算符:&& || ! 返回值:true false**

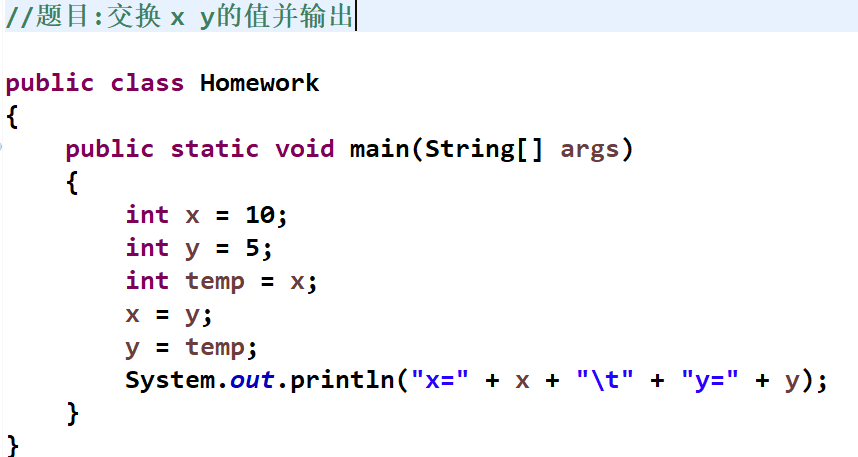
**9)数据类型转换:**

**1.自动类型转换:byte<short<int<long<float<double**

**2.强制类型转换:** 



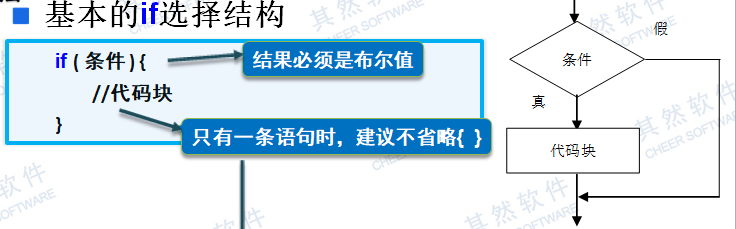
**经典例题:**



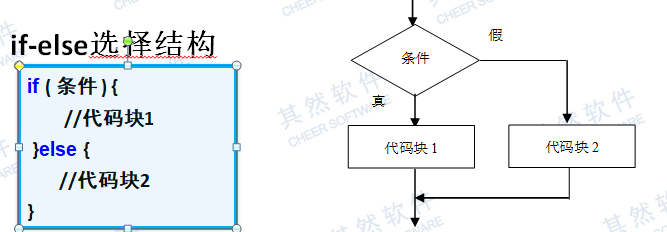
**(3)流程控制**

**1)分类:if语句 if … else语句 if … else if多分支语句 switch开关语句**

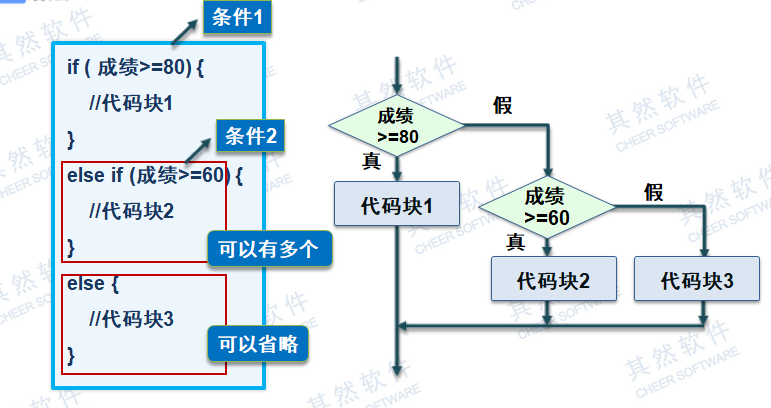
**2)if语句选择结构:**



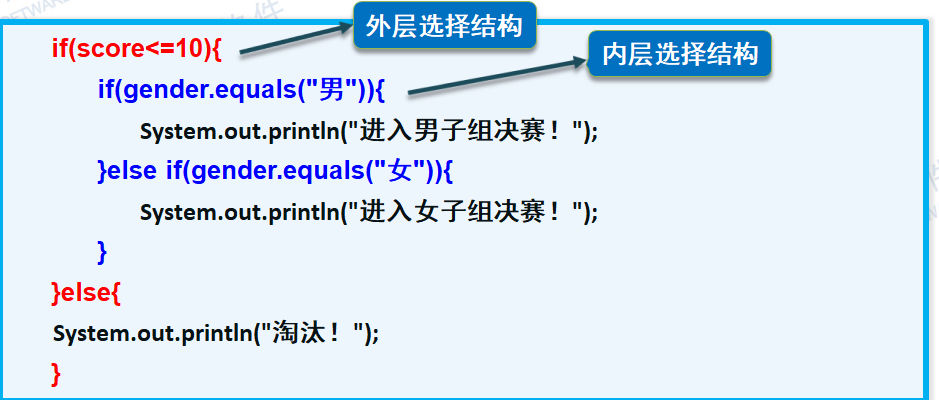
**3) if … else选择结构:**



**4) if … else if多分支选择结构:**



**5) 嵌套if选择结构:**

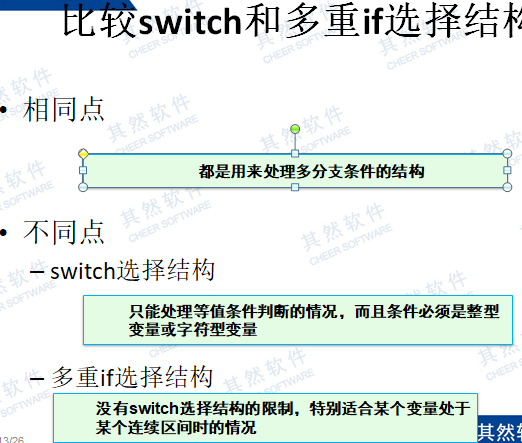


**书写规范: 1.为了使 if 结构更加清晰、避免执行错误，应该把每个 if 或 else 包含的代码块都用大括号括起来 2. 相匹配的一对 if 和 else 应该左对齐 3. 内层的 if 结构相对于外层的 if 结构要有一定的缩进**

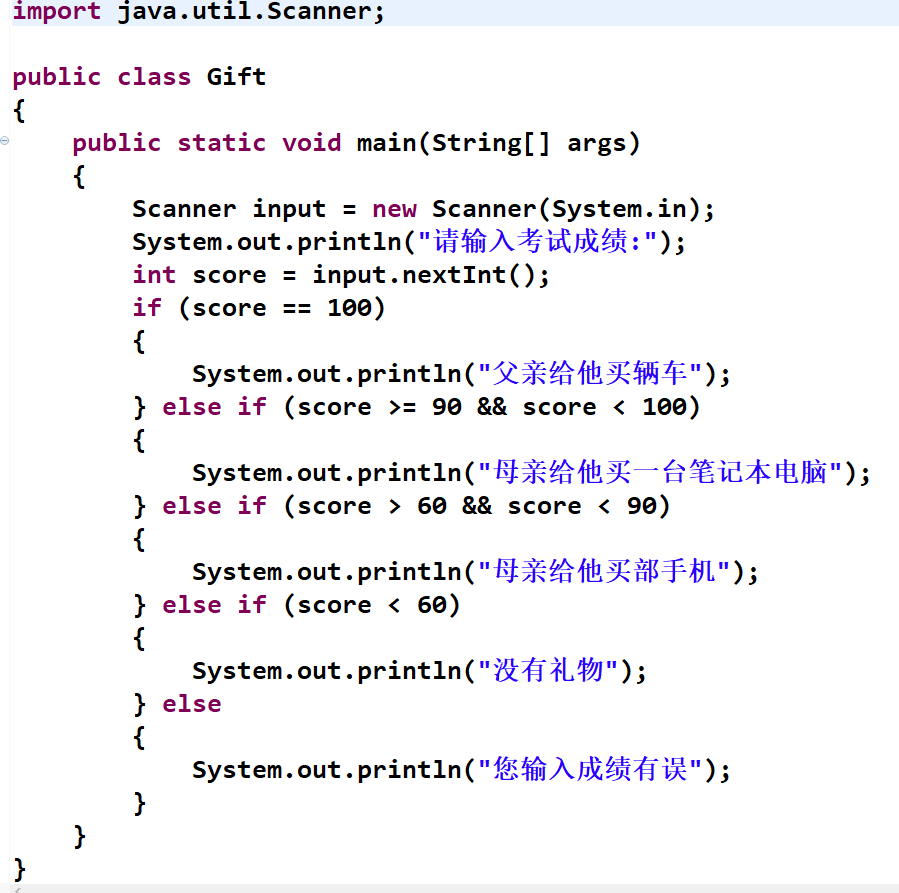
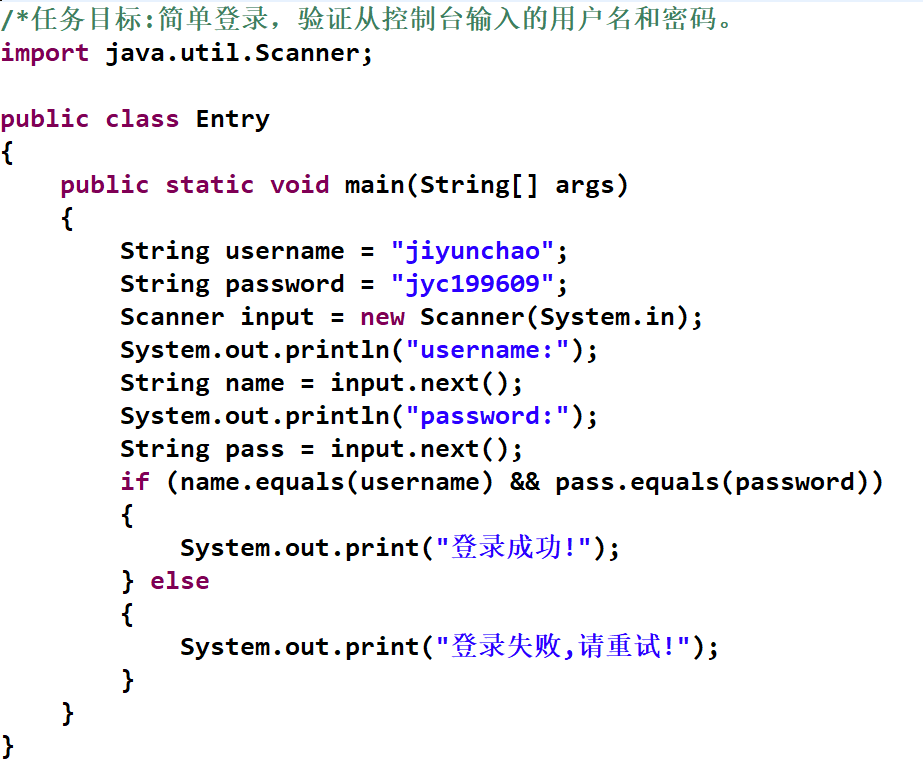
**6)switch开关语句结构:**

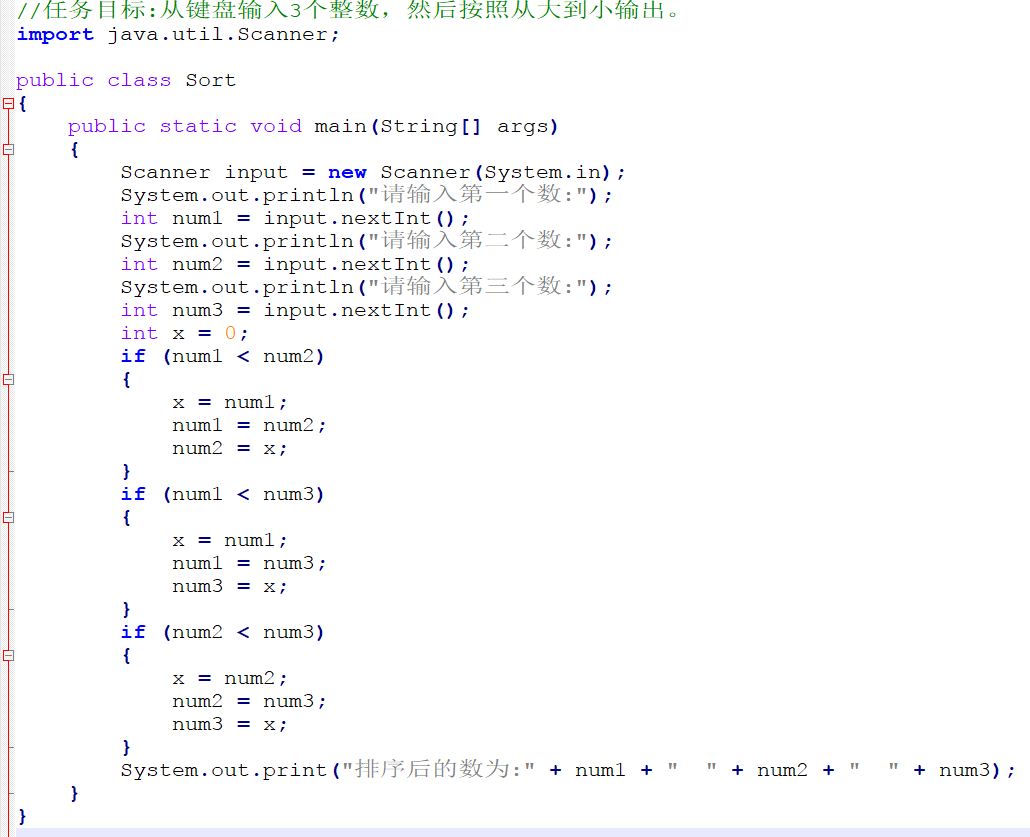
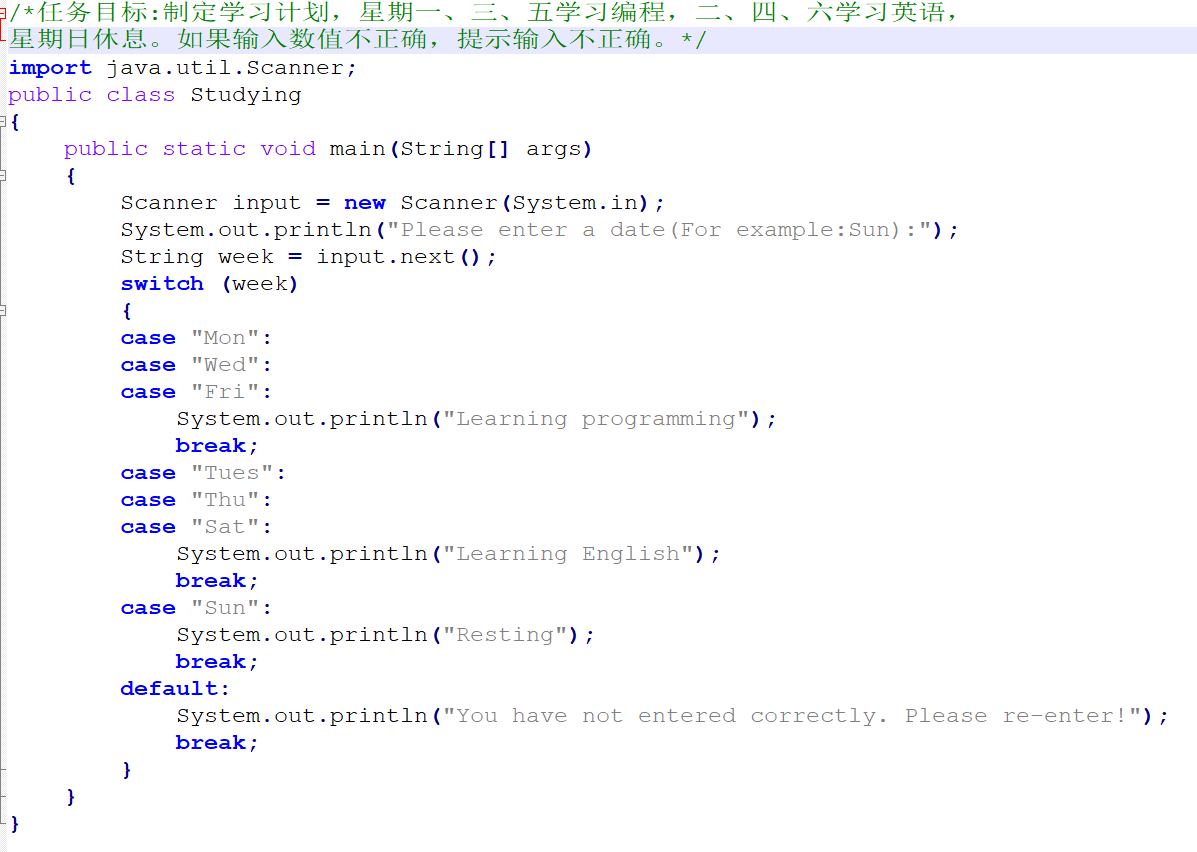


**7)与多重if语句比较:**



**经典例题:**

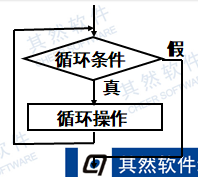
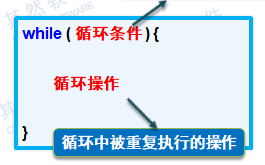




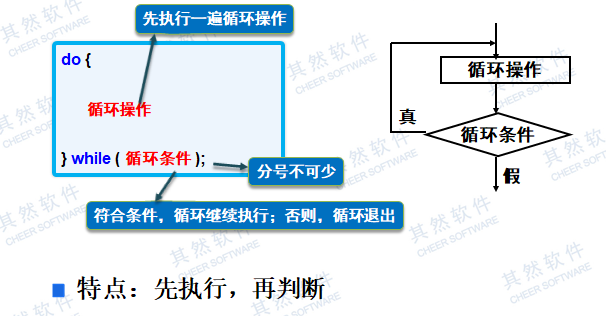
**(4)循环结构**

**1)分类:while循环 do … while循环 for循环**

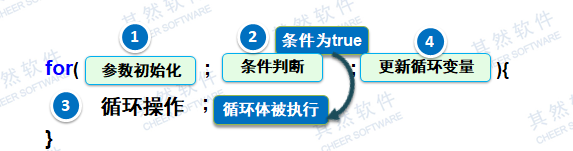
**2)while循环格式:**



**3)do … 循环格式:**



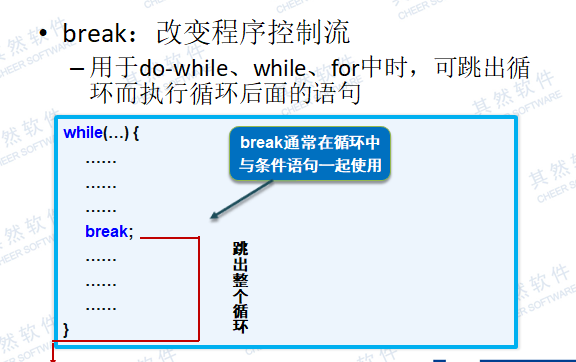
**4)for循环格式:**



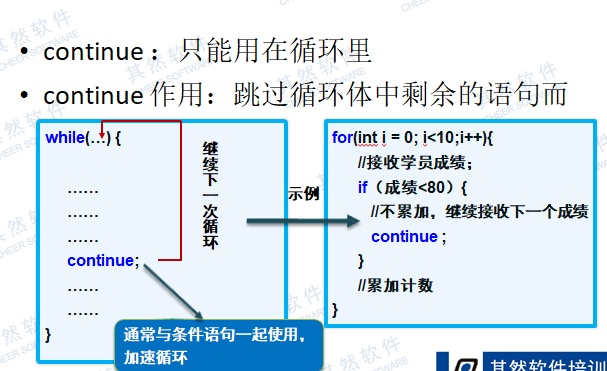
**5)while与do … while区别:**



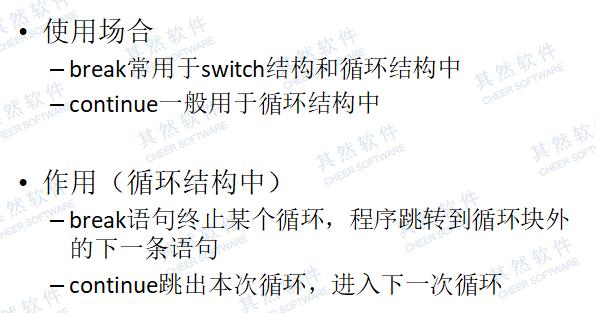
**6) Break语句:**



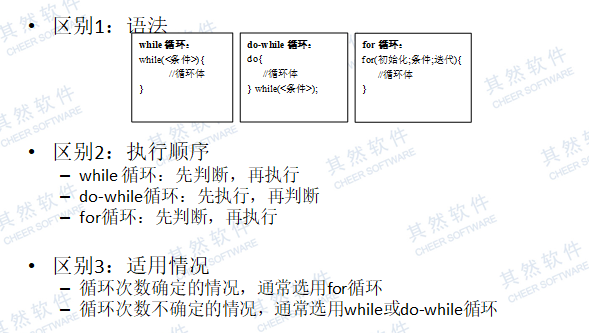
**7) continue语句:**



**8)break与continue对比:**

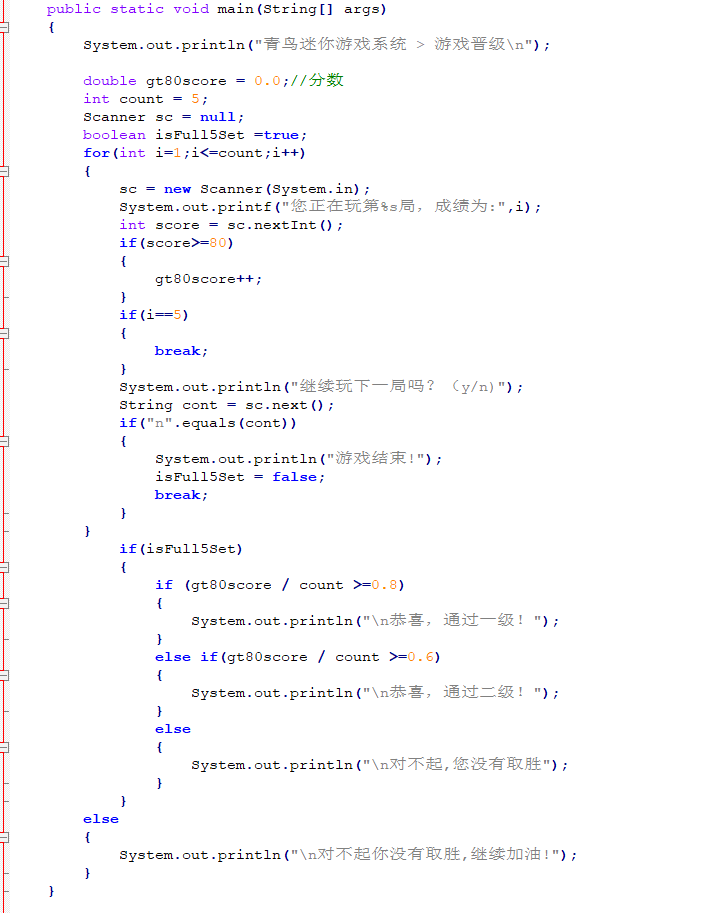


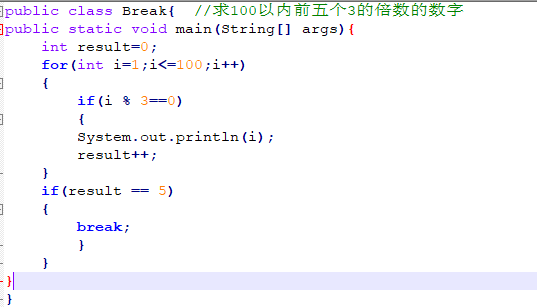
**10.循环总结:**

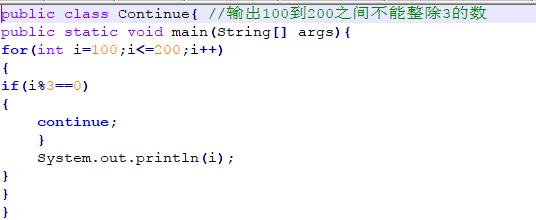


**9经典例题:**



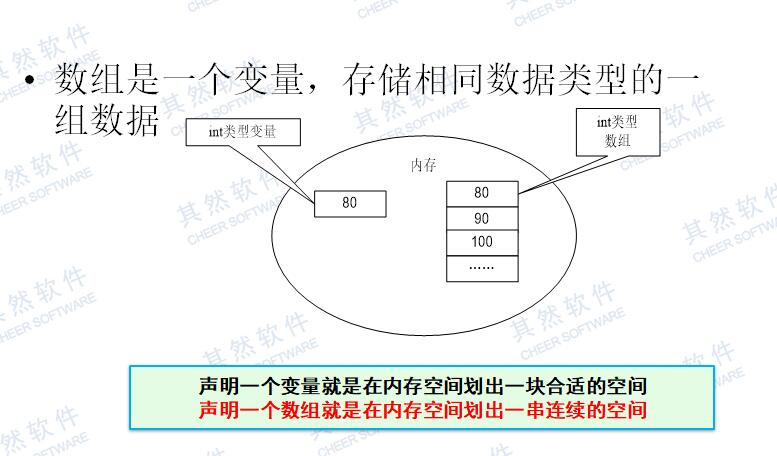
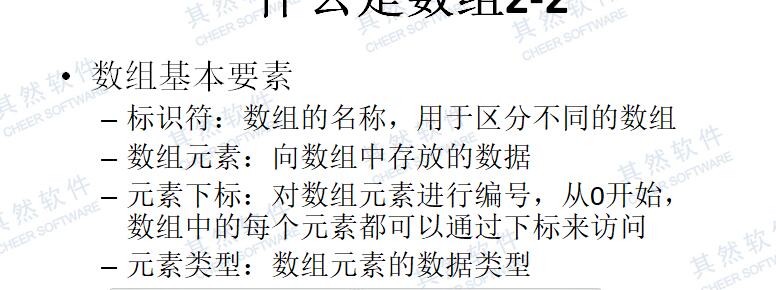




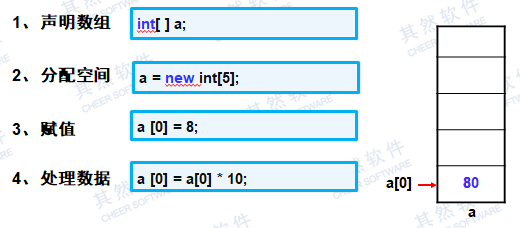


**(5)数组**

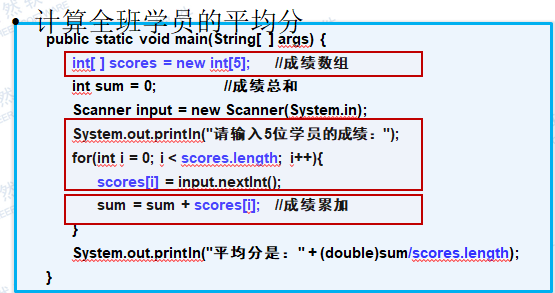
**1).什么是数组?**

****

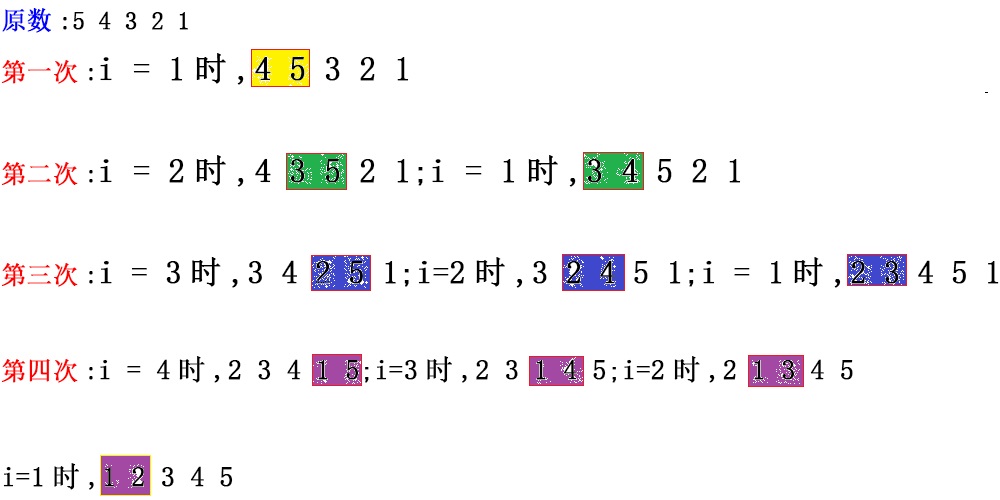
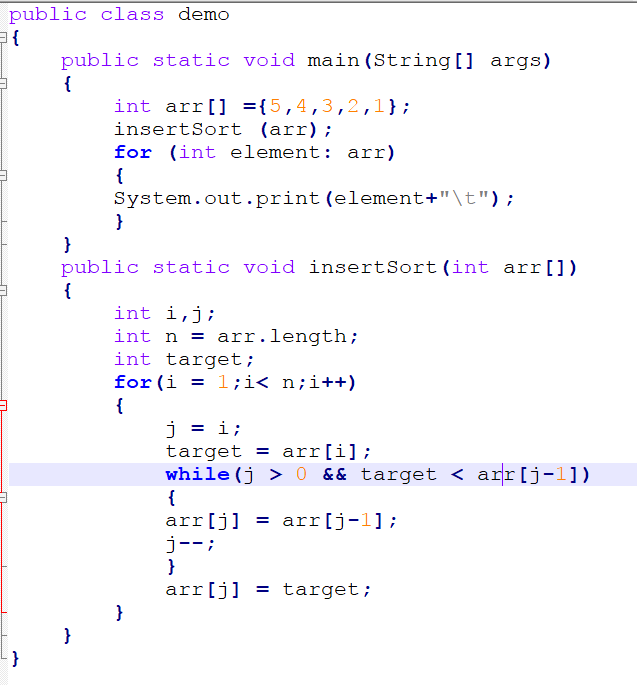
**2.数组的定义:**



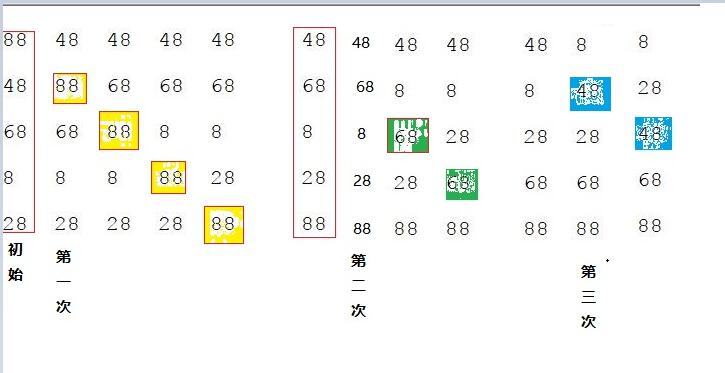
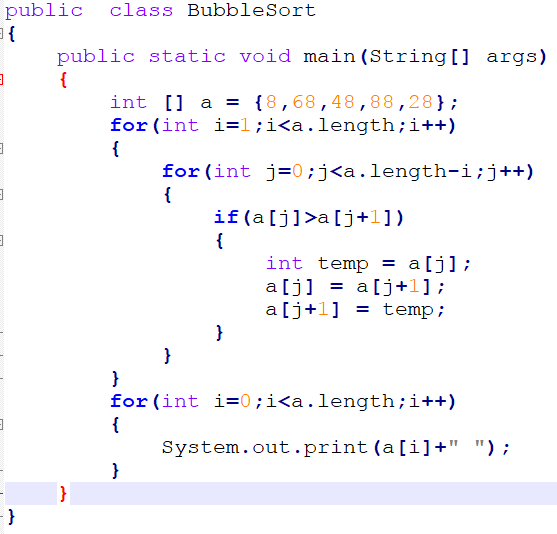
**3). 使用数组解决实际问题:**



**4).插入排序法:**

****

**5)冒泡排序法:**

****

**(6)字符串**

**1).字符串声明:**String(字符串类型)+str(字符串变量名)

**2).字符串创建:**1.String(char a[]) 用一个字符数组创建String对象 2.String(char a[],int offset ,int length),提取数组的部分创建一个字符对象,offset是截取字符的位置,longth()为字符长度 3.Sting(char[] value) 创建一个新的String对象,表示字符数组的有元素连接的结果

**3).字符串方法:** 获得字符串的长度：length();比较字符串：equals();连接字符串：concat();提取字符串：substring();搜索字符串：indexOf();拆分字符串：split();获得指定索引位置的字符:charAt();去除空格:trim();字符串替换:relace();判断字符串开始结尾:startWIth();按字典顺序比较两个字符串:compare To();大小写转换:toLOwerCase() toUpperCase();

4). **StringBuffer类方法**: 转换成String类型：toString();连接字符串：append();插入字符串：insert()

**5)经典例题:**

