1、环境变量:一般是指在操作系统中用来指定操作系统的一些参数，如：临时文件夹的位置和系统文件夹位置等。

2、第一个程序：

public class Helloworld{ //class 后面的是类名，首字母必须大写

public static void main{ //程序入口，只能有一个

system.out.println(“hello world”);

}

}；

3、系统输出语句；

4、**变量**

**定义：变量是一个数据存储空间的表示**

为什么要有变量：根据不同的数据类型将数据存储到不同的内存空间

-- 计算 数据（0~9）、字符|字符串、布尔值、数据类型：《重点》

四类八种基本数据类型

第一类：整型byte short int long

第二类：浮点型float double

第三类：逻辑型boolean(其值只有ture false)

第四类：字符型char 如：’男’或者’女’

基本数据类型自动转换

Int->long->double

Byte->short

小可转大，大转小会失去精度！！！

强转：在变量前加上括号，括号中的类型就是要强制转换成的类型，例如：double a = 3.562;

Int b= (int) c;

运行后b的值为3

人脑：计算区域、存储区域

电脑：CPU、内存

存储计算后的结果数据（中间数据）

有效范围、有效期：作用域；

输入的数据，也是映射到大脑皮层；

变量：在内存中开僻出的一个空间（区域），用来 存储、暂时的；该区域可以重复存放不同数据；

方法1：

第一步：数据类型 变量名 ；//申明变量；

第二步：变量名 = 数据 ；//赋值语句；赋值符号；

如：int a;

a=10;

方法2：

边声明边赋值：数据类型 变量名=数据；

如：String name= “zhangsan”;

**变量命名规范**:

**1.必须以字母、下划线“\_”或者“$”符号开头**

**2.变量名可以包含数字，但不能以数字开头 如：不可以“\*”开头**

**3.不可以有特殊字符 如：“+”“%”等**

**4.不能使用关键字，如：int、public 、class等**

1. 运算符

算术运算符： + - \* / %（取余|取模） ；

比较运算符|关系运算符： > 、<、 ==、>=、 <=、 !=；

逻辑运算符： &&(and) 、||(or)、 !(not)；

**优先级：算术运算符》比较运算符》逻辑运算符**

1. **条件控制语句**

如果 （条件满足） {

做什么事件//执行语句；代码块

}

否则{

做其它事件

}

Int i =scanner.nextInt();

If(i < 5){  
 代码块1

}else{

代码块2// i>=5

}

# 多重条件控制语句：

If (条件表达式1){

//代码块1

}else if(条件表达式2){

//代码块2

}else{

//代码块3

}

~~If (条件表达式1){~~

~~}~~

~~if(！条件表达式1 && 条件表达式2){~~

~~}~~

这种方式：代码复杂；计算运行效率低，不建议采用；

# 嵌套语句

条件表达式的代码块中（大括号中），再放一个条件表达式；

If(条件表达式1){

If(条件表达式2){

}else{

}

else{

}

书写规范：相匹配的if和else应该左对齐，内层的if结构要有一定的缩进。

# Switch语句

Switch(待比较的表达式)

只能做等值比较；

Case 关键字；

Case 比较值 **：{**

**}**

Break 关键字；

中断、跳出当前switch语句；

Default 关键字；

1. 循环语句；

* While循环语句
* 初始化语句、比较语句、变量叠代；

While(比较语句){

//循环体；需要重复执行的代码；

//添加退出机制（步长）

//或者用break退出;

}

* Do While

//先执行一次，再判断是否继续循环

// while: 先判断是否继续循环，满足条件才执行

Dowhile最后的分号“；”不要漏掉；

**循环语句，也可以嵌套；**

* For循环

把数据初始化、比较语句、叠代语句，放在一起；

For(数据初始化 **;** 比较语句 **;** 叠代语句){

//循环的内容

}

Break:中断或者退出循环体;

Continue ： 本次循环中，关键字后面的不执行，直接转到下次循环;

**Break 和continue，只在当前循环中有效；**

**Return：结束当前方法的执行并退出，返回调用该方法的语句处。**

* **数组**

方便管理大量数据（变量）

数据类型 [] 数组名；//申明；

两种初始化

**基本初始化**

* New 数据类型 [数组长度]

数组中，没有任何数据值，只默认值 ；

带数据的初始化

* New 数据类型[]{数据1，数据2}；

此种情况，不用指定长度，使用后面数据个数作为长度；

给数组赋值

**数组名[索引|下标] = 数据**

动车001 ，第8号车厢；

声明方式如：double [] array ={1,2,3,4 };

double [] 数组名=new double[数组长度]；

使用数组：

**数组名[索引|下标]**

**如：array[0]**

***二维数组:实际上是一维数组，它的每个元素又是一个一维数组。***

***语法：数据类型 数组名[][]=new 数据类型[数组长度a][数组长度b]***

条件

人与人交流：自然的语言

人与机交流：程序语言；

计算机内部：计算机语言；01