1..%:取余数,取模以后结果的符号，取决于被模数的符号。

int m1 = 13;

int n1 = 5;

System.out.println(m1 % n1);//3

int m2 = -13;

int n2 = 5;

System.out.println(m2 % n2);//-3

int m3 = 13;

int n3 = -5;

System.out.println(m3 % n3);//3

int m4 = -13;

int n4 = -5;

System.out.println(m4 % n4);//-3

2..条件表达式的运算结果一定是boolean类型。

3..一旦执行某个条件表达式返回true，则进入其执行语句执行。执行结束，则退出当前if-else结构。

如果所有的条件表达式都返回false,则执行else中的条件表达式。

4..一、表达式之间关系的情况

1如果两个条件表达式之间没有交集：哪个条件表达式声明在上，哪个条件表达式声明在下面都可以

2如果两个条件表达式之间有交集：具体问题具体分析，看谁适合写在上面。

3如果两个条件表达式之间是包含关系：将范围小的声明在范围大的上面。

二、其他

如果执行语句只有一行，则可以省略其所在的一对{}。但是建议大家不要省略

1..random()返回[0.0,1.0)范围的一个double型的随机数

2..如何获取一个两位数，

即[10,99] [0.0,1.0) --> [0.0,90) --> [0,89] --> [10,99]

int v2 = (int)(Math.random() \* 90 ) + 10;

System.out.println(v2);

3..如何获取指定范围的一个随机整数： 即[a ,b]:(int)(Math.random() \* (b - a + 1)) + a

2.. 根据switch中表达式的值，依次匹配case中的值。一旦满足某个case的值，则进入其执行语句执行。执行完以后，有可能继续执行其后的其他case结构与default结构。直到遇到break关键字或程序最后终止。

3.. switch后的表达式，其结果为一个变量。此变量只能是如下类型：byte\short\char\int\枚举类(jdk5.0)\字符串(jdk7.0)

5.. switch case中:default类似于if-else中else结构。

default的位置是灵活的。

default是可选的。类似的，if-else中的else结构也是可选的。

6..在switch-case中不一定要使用break