



一. 判断语句

1. 单if语句

```
if(判断条件) {  
    执行语句体;  
}
```

2. 标准的if-else语句

```
if(判断条件) {  
    执行语句体1;  
}  
else {  
    执行语句体2;  
}
```

3. 扩展的if-else语句

```
if(判断条件1) {  
    执行语句体1;  
}  
else if(判断条件2) {  
    执行语句体2;  
}  
...  
else if(判断条件n) {  
    执行语句体n;  
}  
else {  
    执行语句体n+1;  
}
```

4. 标准的if-else语句和三元运算符的互换

在某些简单的应用中，标准的if-else语句是可以和三元运算符互换使用的。凡是可以用三元运算符的均可以使用标准的if-else语句来实现，但是能用标准的if-else语句的不一定能用三元运算符来实现。

二. 选择语句

switch语句的语法格式如下：

```
switch(表达式) {  
  
    case 常量值1:  
        语句体1;  
        break;
```

```

        case 常量值2:
            语句体2;
            break;
        ..

        case 常量值n:
            语句体n;
            break;

        default:
            语句体n+1;
            [break;]    // 最后一个break语句可以省略，但是强烈推荐不要省略
    } //switch语句在此结束

```

#.注意:

a.表达式返回的常量和变量只能是下列数据类型:

- 基本数据类型: byte、short、int、char
- 引用数据类型: String字符串(仅限于JDK7以上)、enum枚举

b.case后面的数值不可以重复,若有重复则编译器会报错

c.case的穿透性:

如果进入到某个case且其后面不写break,将出现穿透现象,也就是直接按顺序穿透下面所有的case和default语句向后执行,直到遇到break,或者整体switch结束。(即匹配到哪一个case就从哪一个位置向下按顺序穿透执行,直到遇到了break或者整体结束为止,所以case后面break不能少)

d.switch语句格式可以很灵活: case、default语句的顺序可以颠倒,而且颠倒后不影响和case的匹配,也不影响default是最后默认的执行顺序

### 三.循环语句

循环结构的基本组成部分:

- 初始化语句: 在循环开始最初执行,而且只执行唯一一次,一般为初始化循环变量。
- 条件判断语句: 如果成立,则循环继续;如果不成立,则退出循环。
- 循环体: 要重复执行的代码,若干行语句。
- 步进语句: 每次循环结束之后都要执行一次,一般为更新循环变量使得条件判断不再成立,以此跳出循环。

1.for循环语句

```

for(初始化语句; 条件判断语句; 步进语句){
    循环体;
}

```

2.while循环语句

(1).标准格式:

```

while (条件判断语句) {
    循环体;
}

```

(2).扩展格式:

```

初始化语句;
while (条件判断语句) {
    循环体;
    步进语句;
}

```

3.do...while循环语句

(1).标准格式:

```

do {
    循环体;
} while (条件判断语句); // 注意此时有个分号不能丢

```

(2).扩展格式:

```

初始化语句;
do {
    循环体;
} while (条件判断语句);

```

```
        步进语句；
    } while (条件判断语句); // 注意此时有个分号不能丢
```

#.注意：

a.如果条件判断语句从来没有满足过，那么for、while循环将会执行0次，但是do...while循环会执行至少一次。

b.三种循环之间可以互相转换。即一个循环问题，我们可以使用三种不同的循环语句来解决。但是他们的应用场景有所差别：

- for循环更适用于计次数的循环问题(已知循环次数的问题)
- while、do...while循环更适用于判断条件的循环问题(循环次数未知的问题)
- do...while循环更适用于先执行一次某段代码再进行循环判断的问题(do...while循环使用较少)

c.循环变量的作用域：

- for循环的循环变量i在小括号当中定义，意味着该循环变量的作用域仅限于for循环内部，只有循环内部才可以使用该变量。(for循环结束，该变量就从内存中消失，能够提高内存的使用效率)
- while、do...while循环的初始化语句本来就在循环外面，所以出来循环之后还可以继续使用。

#### 4.循环控制语句break和continue

(1).break语句

- 可以用在switch语句当中，一旦执行break，整个switch语句立刻结束，起到终止switch的作用
- 可以用在循环体语句当中，一旦执行break，整个循环语句立刻结束，起到跳出循环的作用

(2).continue语句：一旦执行continue，则立刻跳过当前次循环剩余内容，马上开始下一次循环

#### 5.死循环与嵌套循环

(1).死循环：当循环中的条件判断语句永远为true，则会一直循环下去，就叫做死循环。死循环的常用标准格式如下：

```
while (true) {
    循环体；
}
```

(2).嵌套循环：是指一个循环的循环体内包含另一个循环。比如：for循环里面还有一个for循环(for循环的嵌套)、while循环里面还有一个while循环(while循环的嵌套)、for循环里面还有一个while循环(for循环与while循环的嵌套)等等，就是嵌套循环。语法格式如下：(以for循环嵌套为例)

```
for(初始化语句；条件判断语句；步进语句) {
    for(初始化语句；条件判断语句；步进语句) {
        内循环体；
    }
}
```

a.嵌套循环的执行特点是外循环执行一次，内循环执行多次。嵌套循环的循环次数 = 外循环次数 \* 内循环次数。

b.上述所说的都是两个循环嵌套在一起，叫做双重嵌套循环。其实可以有多个循环嵌套在一起，叫做多重嵌套循环。多重嵌套循环总共的循环次数=第一层循环次数 \* 第二层循环次数 \* 第三层循环次数 \* ...