

第三章 流程控制语句



□ 流程控制分类：

➤ 顺序语句

- 从上到下按顺序依次执行

➤ 分支语句

- 根据不同条件，执行不同语句

➤ 循环语句

- 重复执行某些动作

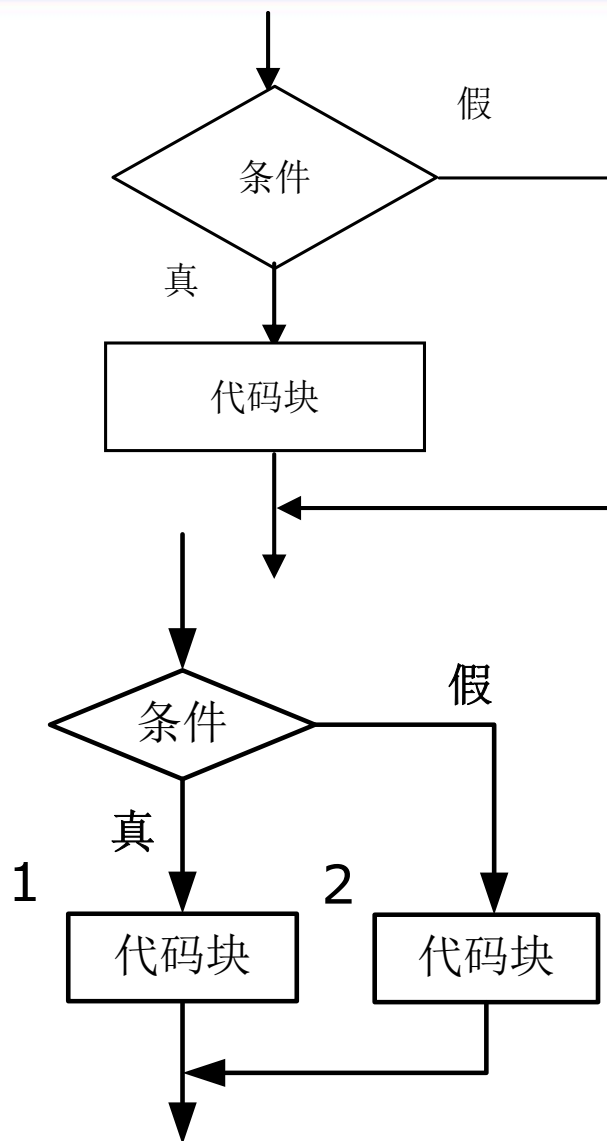
条件语句



语法格式：

```
if (条件) {  
    //代码块  
}
```

```
if (条件) {  
    //代码块1  
} else {  
    //代码块 2  
}
```



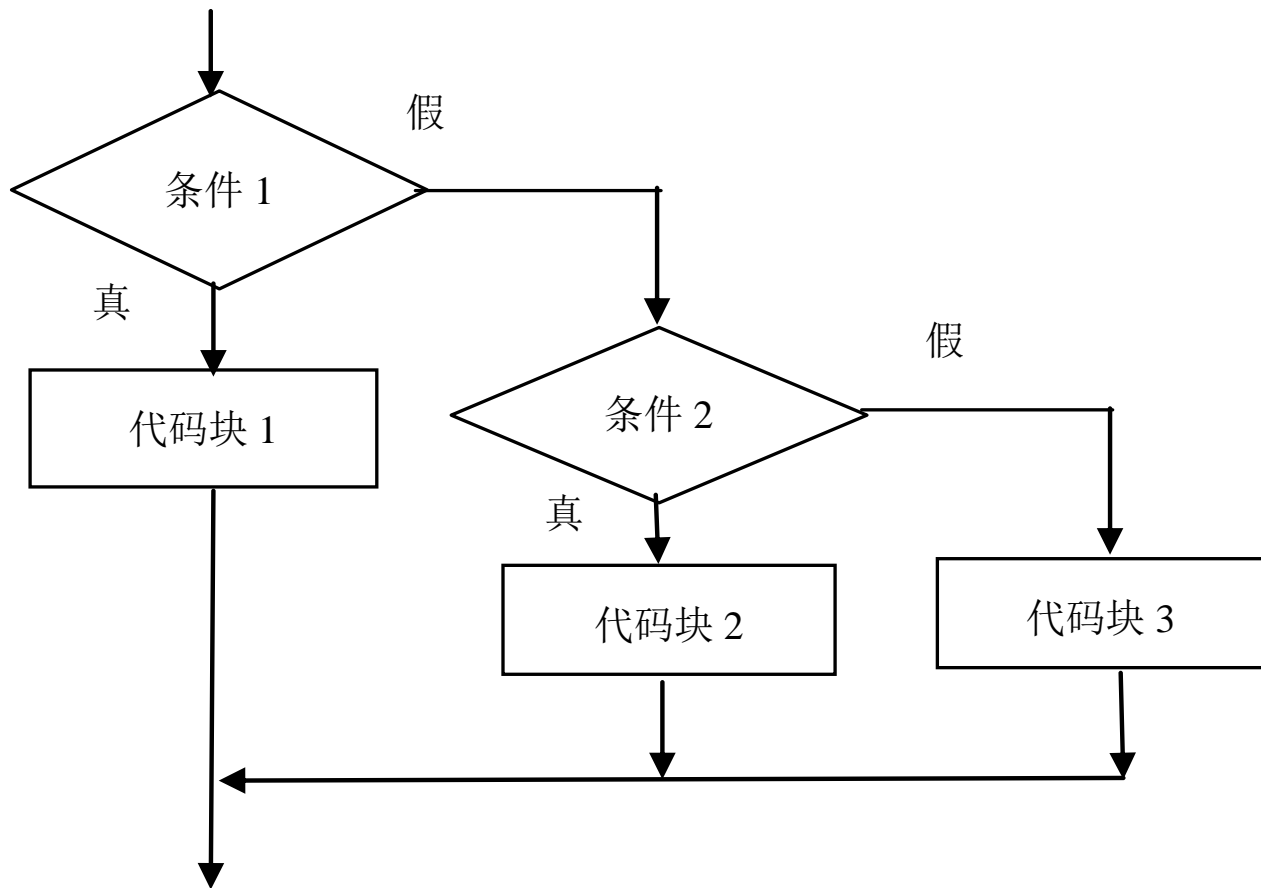
什么是多重if选择结构



语法：

多重if选择结构

```
if ( 条件 1 ) {  
    //代码块1  
}else if ( 条件2 ) {  
    //代码块2  
}else {  
    //代码块3  
}
```



多分支选择语句



- **switch**语句
- 与处理多条件的if语句功能相同

byte、
int、
short、
char
String
枚举

```
switch(表达式){  
    case 值1 :  
        语句;  
        break;  
  
    case 值2 :  
        语句;  
        break;  
  
    .....  
    default:  
        语句;  
        break;  
}
```



```
if(条件){
```

```
    单分支if
```

```
    双分支if-else
```

```
    多分支if 或 switch
```

```
}else{
```

```
    单分支if
```

```
    双分支if-else
```

```
    多分支if 或 switch
```

```
}
```

```
switch(表达式){
```

```
case 常量值:
```

```
    单分支if
```

```
    双分支if-else
```

```
    多分支if 或 switch
```

```
    break;
```

```
}
```

什么是嵌套if选择结构.



嵌套if选择结构

```
if (条件1) {
```

```
    if (条件2) {
```

```
        代码块1
```

```
    } else {
```

```
        代码块2
```

```
    }
```

```
} else {
```

```
    代码块3
```

```
}
```

内层选择结构



□ 什么时候使用循环？

□ 循环的分类

while

do-while

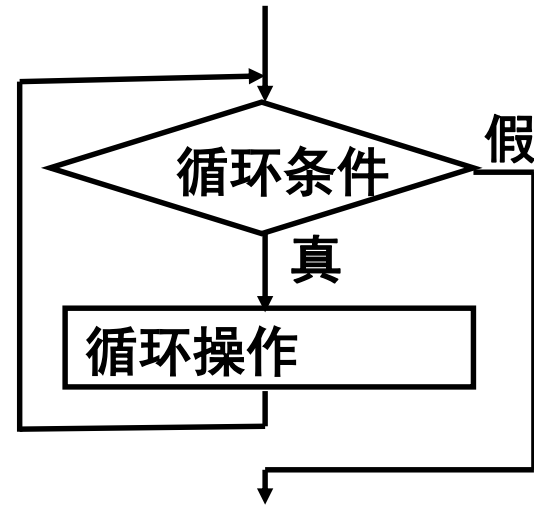
for

什么是while循环



语法：

```
while ( 循环条件 ) {  
  
    循环操作  
  
}
```



什么是do-while循环

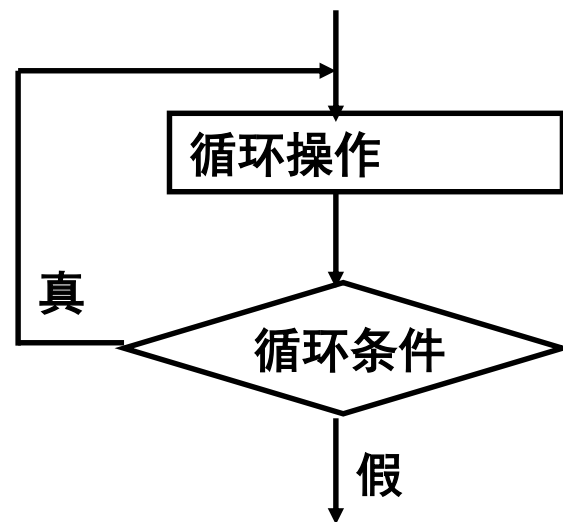


语法：

```
do {
```

循环操作

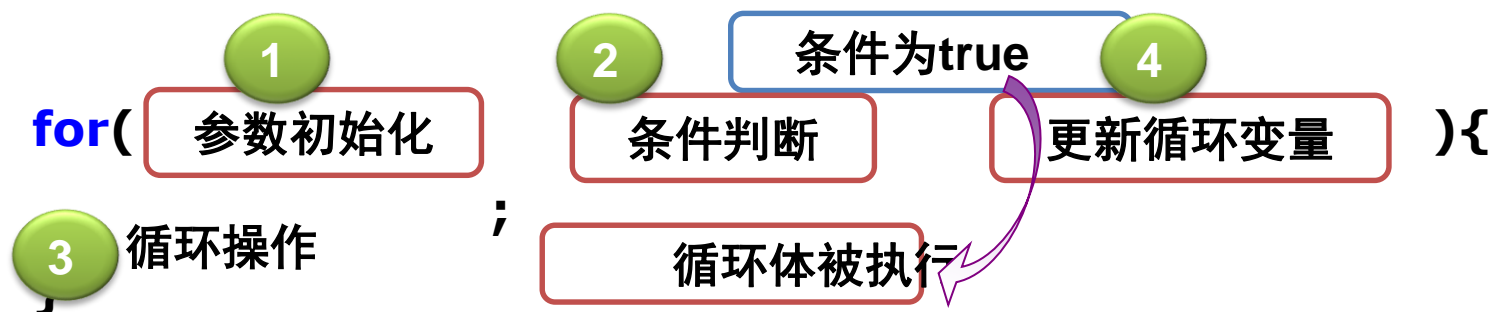
```
} while ( 循环条件 );
```



for循环



语法:

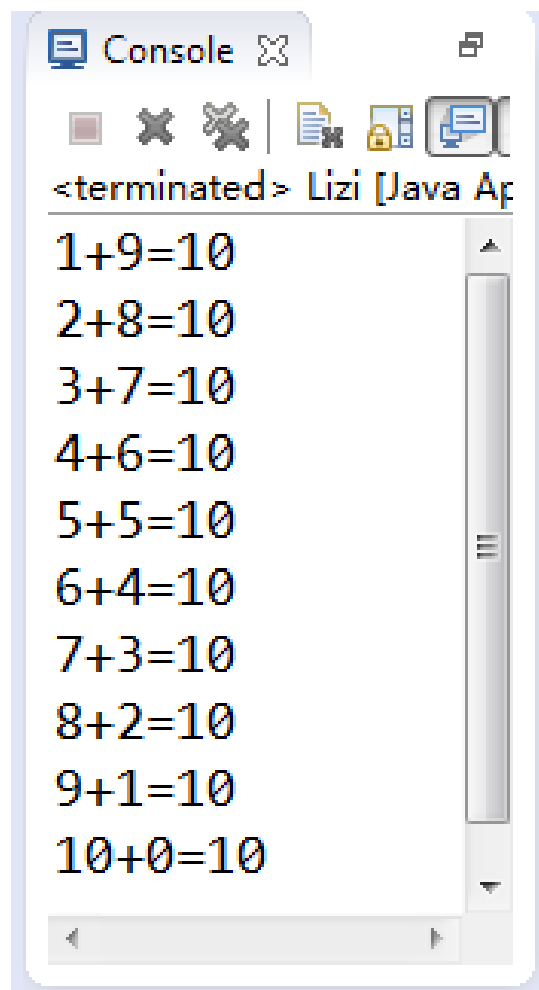


```
for ( int i = 0 ; < 100 ; i++ ){  
    System.out.println("好好学习! ");  
}
```

for-逗号表达式



示例：

A screenshot of a Java IDE's console window. The window has a title bar that says "Console" with a close button. Below the title bar is a toolbar with icons for running, debugging, and other actions. The main area of the console shows the output of a program. It starts with "<terminated> Lizi [Java Ap" and then lists ten lines of arithmetic calculations: "1+9=10", "2+8=10", "3+7=10", "4+6=10", "5+5=10", "6+4=10", "7+3=10", "8+2=10", "9+1=10", and "10+0=10". The console window has a scrollbar on the right side.

```
<terminated> Lizi [Java Ap
1+9=10
2+8=10
3+7=10
4+6=10
5+5=10
6+4=10
7+3=10
8+2=10
9+1=10
10+0=10
```



区别

while 循环:

```
while(<条件>){  
    //循环体  
}
```

do-while 循环:

```
do{  
    //循环体  
} while(<条件>);
```

for 循环:

```
for(初始化;条件;迭代){  
    //循环体  
}
```

什么是break语句



break : 改变程序控制流

```
while(...) {
```

```
.....
```

```
.....
```

```
.....
```

```
break;
```

```
.....
```

```
.....
```

```
.....
```

```
}
```

跳出整个循环



什么是continue语句



□ continue 作用

- continue语句用来结束本次循环
- 用在循环语句中

```
while(...) {
```

```
.....
```

```
.....
```

```
.....
```

```
continue;
```

```
.....
```

```
.....
```

```
}
```

继续下一次循环

示例

```
for(int i = 0; i<10;i++){
```

```
    跑400米;
```

```
    if (不口渴) {
```

```
        continue;
```

```
    }
```

```
    接过水壶，喝水;
```

```
}
```



□ 语法：

标签名:

```
for(){  
    for(){  
        break 标签名; [continue 标签名;]  
    }  
}
```


二重循环



```
□ for( ){  
    for( ){ }  
}
```

```
□ while( ){  
    while( ){ }  
}
```

```
□ while(){  
    do{ }while();  
}
```

```
□ for( ){  
    while( ){ }  
}
```

```
□ while( ){  
    for( ){ }  
}
```

```
□ for(){  
    do{ }while();  
}
```