

# Java流程控制语句1

---

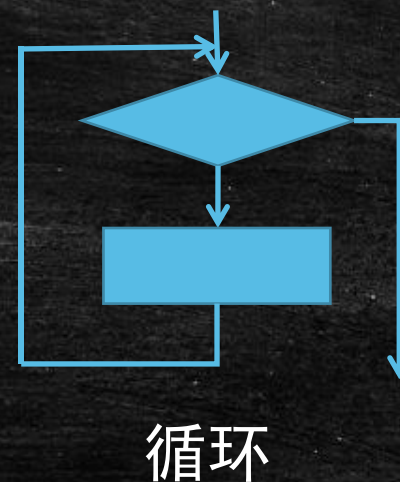
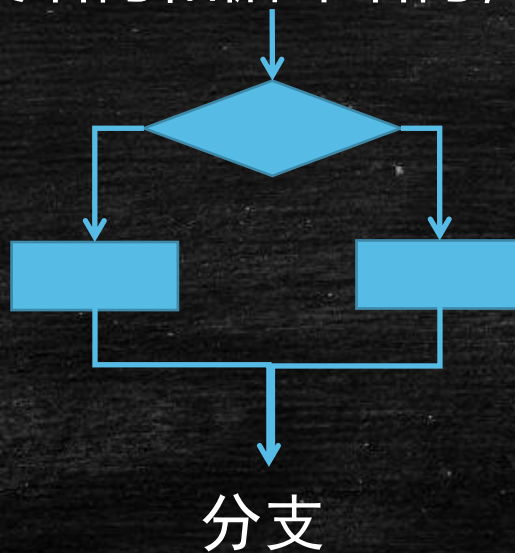
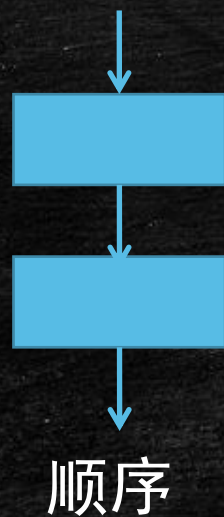
- What?Why?How?





# 本章概述

- 流程控制语句是用来控制程序中各语句执行顺序的语句，可以把语句组合成能完成一定功能的小逻辑模块。
- 其流程控制方式采用结构化程序设计中规定的三种基本流程结构，即：顺序结构、分支结构和循环结构，如下图所示：





# 本章概述

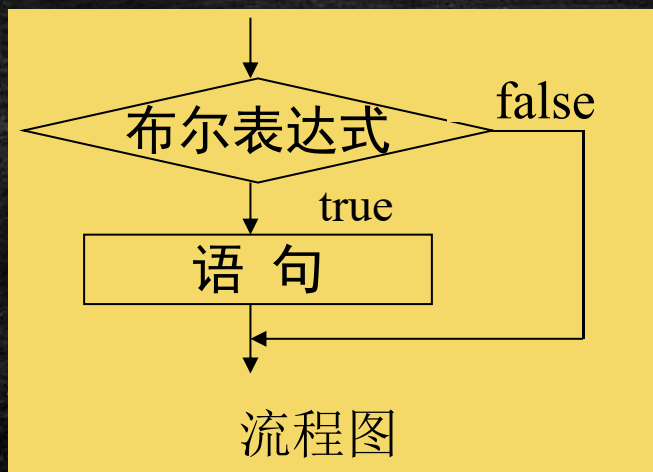
- 顺序结构
- 选择结构
  - if-else switch
- 循环结构
  - while do-while for
- 跳转
  - break continue return
- 多重循环
- 方法
  - 定义、调用、重载
- 递归算法





# if单分支选择结构

- if语句对条件表达式进行一次测试，若测试为真，则执行下面的语句，否则跳过该语句



附：

Math类的使用：

```
int i = (int) (6 * Math.random());  
//产生：[0,5]  
//如何产生：10-15随机数？
```



# if 语句

```
double i = 6 * Math.random();
double j = 6 * Math.random();
double k = 6 * Math.random();
int count = (int) (i + j + k);
if(count > 15 ) {
    System.out.println("今天手气不错");
}
if(count >= 10 && count <= 15) { //错误写法: 10<count<15
    System.out.println("今天手气很一般");
}
if(count < 10) {
    System.out.println("今天手气不怎么样");
}
System.out.println("得了" + count + "分"); 要求必须是布尔表达式
```

条件结果必须是布尔值

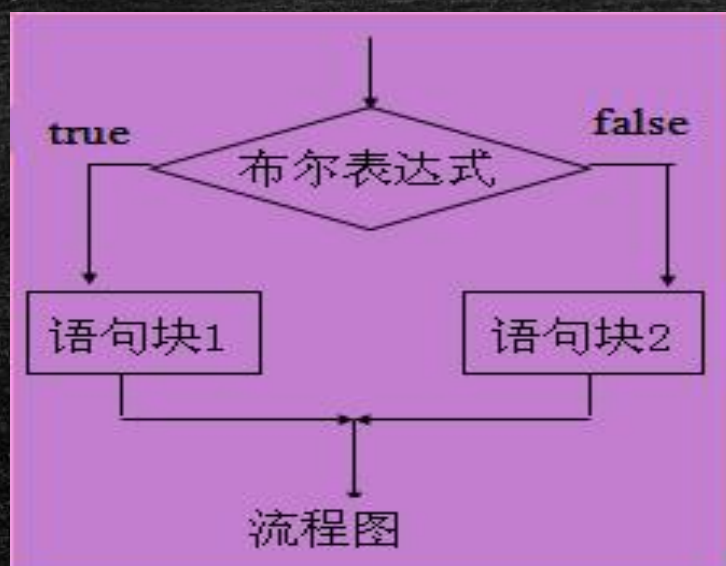
建议都加上花括号。 如果不加花括号, 则只对第一句话有效!





# if-else双分支选择结构

- 当条件表达式为真时，执行语句块1，否则，执行语句块2。也就是else部分



```
double r = 4 * Math.random();
double area = Math.PI * Math.pow(r, 2);
double circle = 2 * Math.PI * r;
System.out.println("半径为: " + r);
System.out.println("面积为: " + area);
System.out.println("周长为: " + circle);
if(area >= circle) {
    System.out.println("面积大于等于
    周长");
} else {
    System.out.println("周长大于面积");
}
```





# 上机练习1—会员信息录入

```
D:\>java AddCust
欢迎登录马士兵教育会员系统
添加会员信息
请输入会员号<4位整数>:
123
请输入会员生日(月/日<用两位整数表示>):
12/12
请输入积分:
12
客户号123是无效会员号
录入信息失败
```

```
D:\>java AddCust
欢迎登录马士兵教育会员系统
添加会员信息
请输入会员号<4位整数>:
1234
请输入会员生日(月/日<用两位整数表示>):
12/23
请输入积分:
45
已录入的会员信息是:
1234    12/23    45
```

需求说明:

录入会员信息  
判断录入的会员号是否  
合法





## 上机练习2—幸运抽奖

- 需求说明：
  - 抽奖规则：会员号的百位数字等于产生的随机数字即为幸运会员

```
D:\>java GoodLuck
欢迎登录马士兵教育抽奖系统
请输入4位会员号码:
1234
1234, 谢谢支持
```

```
D:\>java GoodLuck
欢迎登录马士兵教育抽奖系统
请输入4位会员号码:
1234
恭喜中奖
```





# 课堂练习

- 会员登录，如果用户名是 '青'，并且密码 123，
  - 输入欢迎您青，
  - 否则输出，对不起，您不是会员
- 
- 小孩儿搬桌子：年龄大于7岁，可以搬动桌子，大于5岁
  - 性别为'男'也可以搬动桌子，
  - 否则不可以搬动桌子，你还太小了





# If-else if-else多分支选择结构

- if(布尔表达式1) {
  - 语句块1;
- } else if(布尔表达式2) {
  - 语句块2;
- }.....
- else if(布尔表达式n){
  - 语句块n;
- } else {
  - 语句块n+1;
- }
- 逐条if语句进行判断
  - 条件匹配, 进入语句体
  - 否则对if语句继续匹配

```
int age = (int) (100 * Math.random());
System.out.print("年龄是" + age + ", 属于");
if (age < 15) {
    System.out.println("儿童, 喜欢玩!");
} else if (age < 25) {
    System.out.println("青年, 要学习!");
} else if (age < 45) {
    System.out.println("中年, 要工作!");
} else if (age < 65) {
    System.out.println("中老年, 要补钙!");
} else if (age < 85) {
    System.out.println("老年, 多运动!");
} else {
    System.out.println("老寿星, 古来稀!");
}
```





# If-else if-else多选择结构

- 对学员的结业考试成绩评测
  - 成绩 $\geq 90$  : 优秀
  - 成绩 $\geq 80$  : 良好
  - 成绩 $\geq 60$  : 中等
  - 成绩 $< 60$  : 差

```
int score = 70; //考试成绩
if ( score >= 90 ) {
    System.out.println("优秀");
} else if (score >= 80 ) {
    System.out.println("良好");
} else if (score >= 60 ) {
    System.out.println("中等");
} else {
    System.out.println("差");
}
```





## 上机练习3—多重if买车问题

- 我想买车，买什么车决定于我在银行有多少存款
  - 如果我的存款超过500万，我就买凯迪拉克
  - 否则，如果我的存款超过100万，我就买帕萨特
  - 否则，如果我的存款超过50万，我就买依兰特
  - 否则，如果我的存款超过10万，我就买奥托
  - 否则，我买捷安特





## 上机练习4—多重if计算会员折扣

- 需求说明：
  - 会员购物时，根据积分的不同享受不同的折扣
  - 计算会员购物时获得的折扣

会员积分x	折 扣
$x < 2000$	9折
$2000 \leq x < 4000$	8折
$4000 \leq x < 8000$	7折
$x \geq 8000$	6折





# 为什么使用嵌套if选择结构

- 学校举行运动会，百米赛跑跑入20秒内的学生有资格进决赛，根据性别分别进入男子组和女子组
  - 要判断是否能够进入决赛
  - 在确定进入决赛的情况下，还要判断是进入男子组，还是进入女子组





# 为什么使用嵌套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  
}
```





# 上机练习5—折扣计算

- 需求说明：
  - 普通顾客购物满100元打9折；会员购物打8折；会员购物满200元打7.5折





# switch多分支选择结构

- 根据表达式值的不同执行许多不同的操作

```
▪ switch (表达式) {  
  - case 值1 :  
  - 语句序列;  
  - [break];  
  - case 值2:  
  - 语句序列;  
  - [break] ;  
  - ... ..  
  - [default:  
  - 默认语句 ;]  
▪ }
```

1.switch语句会根据表达式的值从相匹配的执行，一直执行到break标签处开始ak语句处或者是switch语句的末尾。与任一case值不匹配，则进入default语句(如果有的话)

2.只能处理等值条件判断的情况，且表达式必须为byte, short, int或char类型，不能是String或double,float.1.7之后可以使用string

3.常量值必须是与表达式类型兼容的特定的一个常量

4.不允许有重复的case值

5.default子句为可选





# switch多值选择结构

```
char c = 'a';
int rand =(int) (26*Math.random());
char c2 = (char)(c+rand);
System.out.print(c2 + ": ");
switch (c2) {
case 'a':
case 'e':
case 'i':
case 'o':
case 'u':
    System.out.println("元音"); break;
case 'y':
case 'w':
    System.out.println("半元音"); break;
default:
    System.out.println("辅音");
}
```





# 比较switch和多重if选择结构

- 相同点

都是用来处理多分支条件的结构

- 不同点

- switch选择结构

只能处理等值条件判断的情况，而且条件必须是整型变量或字符型变量或者字符串（jdk1.7之后）

- 多重if选择结构

没有switch选择结构的限制，特别适合某个变量处于某个连续区间时的情况





# 课后作业1—快速拨号

- 张三为他的手机设定了自动拨号
  - 按1：拨爸爸的号
  - 按2：拨妈妈的号
  - 按3：拨爷爷的号
  - 按4：拨奶奶的号





# 课后作业2—迷你计算器

---





## 课后作业3—机票打折

- 淡旺季机票的价格，原价机票价格为5000元，
- 淡季头等舱打5折，经济舱打4折
- 旺季头等舱打9折，经济舱打8折
- **要求**
- **编写程序实现：**
- 输入任意的月份与舱位来计算机票的价格
- 1代表头等舱，2代表经济舱
- 4-10月为旺季，其他月份为淡季

