# 简单if：

语法：**if(条件判断表达式){**

**执行语句**

**}**

如果条件判断的表达式返回值是true则可以执行if内部的语句，否则如果是false就不可以执行。

条件表达式的结果一定是true或者false

|  |
| --- |
| public static void main(String[] args){  //定义一个分数的变量  int score = 66;    if(score >= 60){  System.out.println("恭喜了，及格了");  System.out.println("恭喜了，及格了");  System.out.println("恭喜了，及格了");      }  System.out.println("程序结束");    } |

# if…else

**语法：if(条件判断表达式){**

**执行语句**

**}else{**

**执行语句**

**}**

|  |
| --- |
| public static void main(String[] args){  //定义一个分数的变量  int score = 56;    if(score >= 60){  System.out.println("恭喜了，及格了");  }else{  System.out.println("抱歉，你没及格");  }  System.out.println("程序结束");    } |

# 多重if

多重 if….else if…else

**语法：if(条件判断表达式){**

**执行语句**

**}else if(条件表达式1){ //score < 60 && score > 30**

**执行语句**

**}else if(条件表达式2){**

**…..**

**}else{**

**}**

注意：多重if当遇到第一个满足的表达式条件时执行当前的if语句，就不会再向下去执行

|  |
| --- |
| class Demo5{    /\*\*  示例：打印每个员工的工资等级，如果大于3000就是D级  ，如果大于5000是C级，如果大于8000是B级，如果大于10000是A级。    D C B A      3000-5000 5000-8000 8000-10000 10000-  \*/  public static void main(String[] args){  //定义一个工资的变量  int salary = 12000;    if(salary > 10000){  System.out.println("A");  }else if(salary > 8000){  System.out.println("B");  }else if(salary > 5000){  System.out.println("C");  }else if(salary > 3000){  System.out.println("D");  }    System.out.println("程序结束");    }  } |

# Switch分支判断

swtich的表达式只能是byte, short, char, int, String

switch（表达式）{

case 常量1：

执行语句1；

break；

case 常量2：

执行语句2；

break；

…..

default:

执行语句 ；

break；

}

|  |
| --- |
| class Demo7{    /\*\*  根据工资级别打印不同的☆    工资的等级  1 2 3 4    ☆ ☆☆ ☆☆☆ ☆☆☆☆  \*/  public static void main(String[] args){  //定义一个等级的变量  int grade = 2;  //switch的表达式的类型和常量的类型要匹配并且都是byte, short, char, int, String类型  switch(grade){  //如果grade的值和 1相等的话就会执行System.out.println("☆");  case 1:  System.out.println("☆");  break; //跳出switch  case 2:  System.out.println("☆☆");  //break; //跳出switch, 如果没有break就会穿透执行，后面的case即使不匹配也会执行，知道遇到一个break才会跳出switch  case 3:  System.out.println("☆☆☆");  //break; //跳出switch  case 4:  System.out.println("☆☆☆☆");  break; //跳出switch  default:  System.out.println("无级别");  break;  }    System.out.println("程序结束");    }  } |

# 三元运算符

语法：逻辑表达式2？表达式3：表达式4

？前面的逻辑表达式2如果运算的结果是true，整个表达式的结果就是表达式3

？前面的逻辑表达式2如果运算的结果是false，整个表达式的结果就是表达式4

注意：三元运算符适合两个值的判断

|  |
| --- |
| class Demo10{    /\*\*  给定一个变量 1 男， 2, 女    \*/  public static void main(String[] args){  //定义一个变量  int gender = 1;  /\*  if(gender == 1){  System.out.println("男");  }else{  System.out.println("女");  }  \*/  //三元运算适合来代替if else的形式  //逻辑表达式2？表达式3：表达式4 ,三元运算符的结果是由表达式的结果的类型决定， 表达式三和表达式四结果类型要一致  char c = gender == 1?'男':'女';  System.out.println(c);    String man = "我是男";  String result = gender == 1? man:"我是女";  System.out.println(result);    //如果分数大于等于90A， 如果大于等于60B， 小于60C  int score = 99;  /\*  if(score >= 90){  System.out.println("A");  }else if(score >= 60){  System.out.println("B");  }else{  System.out.println("C");  }  \*/  String result1 = score >= 90? "A":(score >= 60 ? "B":"C");  System.out.println(result1);  }  } |

# 循环

## While

语法：while(表达式){

循环体

}

表达式结果如果是true那么就执行循环体，如果是false就结束循环

开始

判断表达式

False

结束

True 再次做判断

循环体

|  |
| --- |
| class Demo12{    /\*\*    示例2：打印1到100之间的所有的偶数和    \*/  public static void main(String[] args){  //定义一个计数器  int i = 1;  //定义一个存储结果的变量  int total = 0;    while(i <= 100){  //判断是否是偶数  if(i % 2 == 0){  total += i;  }    //计数器的累加  i++;  }  System.out.println("结果是："+total);    }  } |

## do…while 循环

语法：

do{

循环体

}while(表达式)

开始

循环体

True

判断表达式

False

结束

|  |
| --- |
| class Demo15{    /\*\*    需求：打印100到1000之间的所有夹心饼数。  如：191,838  分析：  循环条件：判断计数器是大于等于100小于1000  循环操作：更新计数器，判断是否是夹心饼干    \*/  public static void main(String[] args){    //定义一个计数器  int i = 100;    do{  //获得各位  int g = i % 10;  //获得百位  int b = i / 100;    if(g == b){  System.out.println(i);  }  //计数器的累加  i++;  }while(i < 1000);      }  } |

## for循环

**for循环**

语法：for(表达式1; 表达式2; 表达式3){

循环体

}

表达式1：计数器的初始化，它只初始化一次

表达式2：循环条件的判断，多次执行

表达式3：修改计数器，多次执行

详细见代码