

今天的主要掌握的内容是:

运算符和流程控制语句,另外再学习一个新的程序:键盘录入数据.

其中中:运算符和流程控制是要掌握的,键盘录入程序只要能比着抄出来就可以了.

运算符-----

运算符的分类

\* 算术运算符, \* 赋值运算符, \* 比较(关系或条件)运算符, \* 逻辑运算符, \* 位运算符, \* 三目(元)运算符

\* 逻辑运算符

\* & :并且,算出所有的boolean结果,得到值,只有全是true才能得到true

\* | :或者,算出所有的boolean结果,得到值,只有全是false才能得到false

\* ^:异或:只有不一样才能得到true

\* ! :非,就是和这个值反着来.....这个只针对一个boolean值,其它的都是两个boolean值

\* &&:短路并且,结果同&,与&的区别是这个只要运算到能确定true/false了,就不再进行运算了

\* || :短路或者,结果同|,与|的区别是这个只要运算到能确定true/false了,就不再进行运算了

\* 位运算符(这些都是针对二进制数来说的)

\* &:有0则0

\* |:有1则1

\* ^:相同则0,不同则1

\* ~:按位取反

\* <<:左移 左边最高位丢弃,右边补齐0(正负数都适合)

\* >>:右移 最高位是0,左边补齐0;最高为是1,左边补齐1(正数适合,负数不适合)

\* >>>:无符号右移 无论最高位是0还是1,左边补齐0(正数适合,负数不适合,负数右移会变成正数)

最有效率的算出2 \* 8的结果(左移三位)

把0看成false,把1看成true,然后进行判断

\* 三目(元)运算符

boolean值? 值1 : 值2;

键盘录入-----

```
import java.util.Scanner; //第一步:导包
class ReadKeyboard {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in); //第二步:创建键盘录入对象
        System.out.println("请输入第一个整数:");
        int x = sc.nextInt(); //第三步将键盘录入的数据存储在x中
        System.out.println(x); //读到的数据就可以使用了.....
    }
}
//能抄会写
```

流程控制语句-----

\* 顺序结构 \* 选择结构 \* 循环结构

\* 顺序结构

从上到下依次执行

\* 选择结构

\* if语句

#意思是如果对就执行里面的语句,否则就不执行

```
if(boolean 值){
    //TODO 语句
}
```

#意思是如果对,就执行语句1,否则就执行语句2

```
if(boolean 值){
    //TODO 语句1
}
```

```
}else{  
    //TODO 语句2  
}
```

#意思是如果对,就执行语句1,否则就进行下一个判断,如果对执行语句2,否则就进行下一个判断.....如果都不对,就进else执行语句n+1

```
if(boolean 值){  
    //TODO 语句1  
}else if(boolean 值){  
    //TODO 语句2  
}else if(boolean 值){  
    //.....  
    //TODO 语句n  
}else{  
    //TODO 语句n+1  
}
```

注:如果if里面只有一句是可以省略括号的,但我们在开发中并不建议这样做.

\* switch语句

```
switch(值){  
    case 值1:  
        //TODO 如果值等于值1,执行的语句.....  
        break;  
    case 值2:  
        //TODO 如果值等于值2,执行的语句....  
        break;  
    .....  
    default:  
        //都不匹配的时候执行的语句  
}
```

注意:1.break的用法,匹配就象switch拿到钥匙,里面的语句是顺序执行的,如果上面就拿到了钥匙,而没有break,下面的是不进行匹配判断的

2.default语句是在其它case都判断完了再进行的,进行时仍然是顺序执行,所以default一定要放在最后.