

# 操作符

## 1. 代码观察

---

观察以下代码

```
1  int a = 6--;  
2  System.out.println(a);
```

问题：上面代码是否有错误，如果没有，程序结果是多少？

## 2. 代码分析

---

分析以下代码的运行结果 【重点复习原码 反码 补码】

```
1  System.out.println(5&9);  
2  System.out.println(5|9);  
3  System.out.println(5^9);  
4  System.out.println(~-5);
```

```
5 System.out.println(5<<2);
6 System.out.println(-5<<2);
7 System.out.println(-5>>2);
8 System.out.println(-5>>>2);
9 System.out.println(5>>2);
10 System.out.println(5>>34);
11 System.out.println(97=='a');
12 System.out.println(5.0==5);
13 System.out.println(4>5 ^ 'c'>'a');
14 System.out.println((int)(char)(byte)-1);
```

## 3. 指定位置设置为1

---

问题描述：

将参数v的第n位置为1，然后返回该结果值。

代码结构：

```
1 public class Test3_SetOne {
2     // 方法声明如下:
3     public static int setBit(int v,int n){
4         // 提示: 使用 | 及 <<
5
6         // 补全代码
7
8     }
9     public static void main(String[] args) {
10         int r = setBit(10,3);
11         System.out.println("r: " + r);
12     }
13 }
```

## 4. 指定位置设置为0

---

问题描述：

将参数v的第n位置为0，然后返回该结果值。

代码结构：

```
1 public class Test4_SetZero {
2     // 方法声明如下:
3     public static int setZero(int v,int n){
4         // 提示: 使用 ~ 及 <<
5
6         // 补全代码
7
8     }
9     public static void main(String[] args) {
10         int r = setZero(10,2);
11         System.out.println("r: " + r);
12     }
13 }
```

# 流程控制

## 5. 代码分析

---

分析下面几段代码，写出输出结果，最后运行程序进行验证

1)

```
1  int b=5;
2  if(b>4)
3      System.out.println(b);
4  else
5      b--;
6  System.out.println(b);
```

2)

```
1  {
2      int b=5;
3      if(b>4)b--; System.out.println("b大于4");
4      else {
5          System.out.println("b不大于4");
6      }
7  }
```

3)

```
1  int age=45;
2  if(age>20) {
3      System.out.println("年轻人");
4  }else if(age>40){
5      System.out.println("中年人");
6  }else if(age>60){
7      System.out.println("老年人");
8  }
```

4)

```
1  int a=80;
2  switch(a) {
3      case 90:{
4          System.out.println("优秀");
5      }
6      case 80:{
7          System.out.println("一般");
8      }
9      case 60:{
10         System.out.println("及格");
11         break;
12     }
13     case 50:{
14         System.out.println("基础弱");
15     }
16 }
```

5)

```
1  int count=0;
2  while(count>0);
3  {
4      System.out.println(count);
5  }
```

6)

```
1  for (int i = 0; i < 5; i++) {
2      System.out.print(i);
3      i*=0.1;
4  }
5  System.out.println("循环结束");
```

7)

```
1  for(int i=0;i<3;i++) {
2      System.out.println(i);
3      if(i==1) {
4          continue;
5      }
6      System.out.println("continue后的输出语句");
7  }
```

8)

```
1  for (int i = 0; i < 3; i++) {
2      System.out.println(i);
3      if (i == 1) {
4          return;
5      }
6      System.out.println("return后的输出语句");
7  }
```

## 6. 九九乘法表

---

### 问题描述：

编写一个Java程序，实现如下效果的九九乘法表

```
1*1=1
1*2=2   2*2=4
1*3=3   2*3=6   3*3=9
1*4=4   2*4=8   3*4=12  4*4=16
1*5=5   2*5=10  3*5=15  4*5=20  5*5=25
1*6=6   2*6=12  3*6=18  4*6=24  5*6=30  6*6=36
1*7=7   2*7=14  3*7=21  4*7=28  5*7=35  6*7=42  7*7=49
1*8=8   2*8=16  3*8=24  4*8=32  5*8=40  6*8=48  7*8=56  8*8=64
1*9=9   2*9=18  3*9=27  4*9=36  5*9=45  6*9=54  7*9=63  8*9=72  9*9=81
```

### 代码结构：

```
1  public class Test6_MultiplicationTable {
2      public static void main(String[] args) {
3          // 请在此处补全代码
4      }
```

## 7. 素数问题

---



## 问题描述：

请编写一个Java程序，判断101-200之间有多少个素数，并输出所有素数

只能被1和它本身整除的自然数为素数 如101、5等

## 代码结构：

```
1 public class Test7_PrimeNumber {  
2     public static void main(String[] args) {  
3         // 请在此处补全代码  
4     }
```

# 8. 月份计算

---

## 问题描述：

请编写一个Java程序，实现从键盘录入月份，判断该月份属于哪个季节

### 【要求】

- 如果月份不再1-12之间，则报错：输入月份有误

- 匹配月份，输出对应的季节 12-2冬 3-5春 6-8夏 9-11秋
- 使用 `java.util.Scanner` 类完成

代码结构：

```
1  import java.util.Scanner;
2
3  public class Test8_Season {
4      public static void main(String[] args) {
5          Scanner sc = new Scanner(System.in);
6          System.out.print("请输入一个月份: ");
7          // 请继续向下补全代码
8
9      }
10 }
```

## 9. 天数计算

---

问题描述：

请编写一个Java程序，实现从键盘中输入年份 月份 日期，输出这一天是该年的第几天

**【要求】**

- 输入的年份、月份、日期均为数字，不足10的用0x表示，如：1月1日为01 01
- 输入的年份、月份、日期之间用空格隔开，如：2023 04 27

### 【示例】

输入：2023 04 27

输出：117

代码结构：

```
1 public class Test9_DayOfYear {  
2     public static void main(String[] args) {  
3         // 请在此处补全代码，学会使用Scanner类  
4     }  
5 }
```

## 10. 完全数问题

---

问题描述：

请编写一个Java程序，实现输入一个数，判断其是否为完全数

若一个自然数，恰好与除去它本身以外的一切因数的和相等，这种数叫做完全数。

如：

6是完全数，因为， $6 = 1 + 2 + 3$ ;

28是完全数，因为， $28 = 1 + 2 + 4 + 7 + 14$ ;

### 【示例1】

输入：6

输出：6是完全数

### 【示例2】

输入：11

输出：11不是完全数

代码结构：

```
1 public class Test10_PerfectNumber {
2     public static void main(String[] args) {
3         // 补全代码
4     }
5 }
```

## 11. 图形输出

---

问题描述：

请编写一个Java程序，实现下面图形的输出

```
1      *
2     ***
3    *****
4   *********
5    *****
6     ***
7      *
```

【提示】

- 先输出上半部分【先输出空格 再输出\*】
- 再输出下半部分

代码结构：

```
1 public class Test11_PrintDiamond {  
2     public static void main(String[] args) {  
3         // 补全代码  
4     }  
5 }
```

## 12. 分解质因数

---

问题描述：

请编写一个Java程序，实现输入一个15以内的正整数，对该正整数进行分解质因数，输出该整数与质因数的等式。

【示例1】

输入：3

输出：3 = 3

【示例2】

输入：20

输出：输入数字不符合要求

### 【示例3】

输入：8

输出：8 = 2 \* 2 \* 2

代码结构：

```
1  public class Test12_PrimeFactorization {  
2      public static void main(String[] args) {  
3          // 补全代码  
4      }  
5  }
```