一个&、|和两个&&、||有什么区别？

1）&|位与、位或

&&||逻辑与、逻辑或

2）位与和位或可以用来计算数值的二进制位数运算

也可以计算true和false的运算

System.out.println(true|false);

逻辑与和逻辑或只能进行true和false的运算。

System.out.println(10&&3);//错误的

3）逻辑运算符/短路运算符

逻辑运算符具有短路功能的。

短路：直接提前结束，不执行后面的判断过程。

当第一个表达式的结果可以决定整个计算结果的时候，

不会执行第二个表达式的判断过程。

&&：如果第一个表达式结果为false，则跳过后面所有的判断，直接

返回false

||：如果第一个表达式结果为true，则跳过后面所有判断，直接返回true

1.求出1-100之间的偶数之和和奇数之和（使用for循环和while循环）。

代码：

publicstaticvoidtest1(){

intnum1=0;

intnum2=0;

for(inti=1;i<=100;i++){

if(i%2==0){

num1+=i;

}else{

num2+=i;

}

}

System.out.println("偶数之和为"+num1);

System.out.println("奇数之和为"+num2);

}

2.创建一个阶乘应用程序

功能：一个数X的阶乘（通常记作X!）等于X\*(X-1)\*(X-2)\*.....\*1。例如4!等

于4×3×2×1=24。

代码：

publicstaticvoidtest2(intx){

System.out.print(x+"!"+"的阶乘是：");

intnum=1;

for(inti=x;i>=1;i--){

num\*=i;

if(i!=1){

System.out.print(i+"\*");

}else{

System.out.print(i);

}

}

System.out.print("="+num);

}

3.题目：有1、2、3、4个数字，能组成多少个互不相同且无重复数字的三位数？

都是多少？

程序分析：可填在百位、十位、个位的数字都是1、2、3、4。组成所有的排列

后再去掉不满足条件的排列。

代码：

publicstaticvoidtest3(){

System.out.print("互不重复的三位数字有：");

intnum=0;

for(inti=1;i<=4;i++){

for(intj=1;j<=4;j++){

for(intk=1;k<=4;k++){

if(i!=j&&i!=k&&j!=k){

num++;

intnumber=i\*100+j\*10+k;

System.out.print(number+"\t");

}

}

}

}

System.out.println("共有："+num+"个");

}

4.打印出所有的“水仙花数”，所谓“水仙花数”是指一个三位数，其各位

数字立方和等于该数本身。例如：153是一个“水仙花数”，因为153=1的三次

方＋5的三次方＋3的三次方。

代码：

publicstaticvoidtest4(){

inta=0;

intb=0;

intc=0;

intnum=0;

for(inti=100;i<1000;i++){

a=i/100;

b=i/10%10;

c=i%10;

if(i==Math.pow(a,3)+Math.pow(b,3)+Math.pow(c,3)){

num++;

System.out.println(i);

}

}

System.out.println("水仙花数共有："+num);

}

5.打印出四种形式的九九乘法表。

提示：

1、System.out.println()的功能为输出+换行

System.out.print()的功能为输出

2、在适当的位置可以使用'\t'进行对齐操作

形式1：

1\*1=1

1\*2=22\*2=4

1\*3=32\*3=63\*3=9

1\*4=42\*4=83\*4=124\*4=16

1\*5=52\*5=103\*5=154\*5=205\*5=25

1\*6=62\*6=123\*6=184\*6=245\*6=306\*6=36

1\*7=72\*7=143\*7=214\*7=285\*7=356\*7=427\*7=49

1\*8=82\*8=163\*8=244\*8=325\*8=406\*8=487\*8=568\*8=64

1\*9=92\*9=183\*9=274\*9=365\*9=456\*9=547\*9=638\*9=72

9\*9=81

代码：

publicstaticvoidtest5(){

for(inti=1;i<10;i++){

for(intj=1;j<=i;j++){

System.out.print(i+"\*"+j+"="+i\*j+"\t");

}

System.out.println();

}

}

形式2：

1\*9=92\*9=183\*9=274\*9=365\*9=456\*9=547\*9=638\*9=72

9\*9=81

1\*8=82\*8=163\*8=244\*8=325\*8=406\*8=487\*8=568\*8=64

1\*7=72\*7=143\*7=214\*7=285\*7=356\*7=427\*7=49

1\*6=62\*6=123\*6=184\*6=245\*6=306\*6=36

1\*5=52\*5=103\*5=154\*5=205\*5=25

1\*4=42\*4=83\*4=124\*4=16

1\*3=32\*3=63\*3=9

1\*2=22\*2=4

1\*1=1

代码：

publicstaticvoidtest6(){

for(inti=9;i>0;i--){

for(intj=1;j<=i;j++){

System.out.print(j+"\*"+i+"="+i\*j+"\t");

}

System.out.println();

}

}

形式3：

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=72\*7=143\*7=214\*7=285\*7=356\*7=427\*7=49

1\*8=82\*8=163\*8=244\*8=325\*8=406\*8=487\*8=568\*8=64

1\*9=92\*9=183\*9=274\*9=365\*9=456\*9=547\*9=638\*9=729\*9=81

代码：

publicstaticvoidtest7(){

for(inti=1;i<=9;i++){

for(intk=1;k<=9-i;k++){

System.out.print("\t");

}

for(intj=1;j<=i;j++){

if(i\*j>=10){

System.out.print(j+"\*"+i+"="+i\*j+"\t");

}else{

System.out.print(j+"\*"+i+"="+i\*j+"\t");

}

}

System.out.println();

}

}

形式4：

1\*9=92\*9=183\*9=274\*9=365\*9=456\*9=547\*9=638\*9=729\*9=81

1\*8=82\*8=163\*8=244\*8=325\*8=406\*8=487\*8=568\*8=64

1\*7=72\*7=143\*7=214\*7=285\*7=356\*7=427\*7=49

1\*6=62\*6=123\*6=184\*6=245\*6=306\*6=36

1\*5=52\*5=103\*5=154\*5=205\*5=25

1\*4=42\*4=83\*4=124\*4=16

1\*3=32\*3=63\*3=9

1\*2=22\*2=4

1\*1=1

代码：

publicstaticvoidtest8(){

for(inti=9;i>=0;i--){

for(intk=1;k<=9-i;k++){

System.out.print("\t");

}

for(intj=1;j<=i;j++){

if(i\*j>=10){

System.out.print(j+"\*"+i+"="+i\*j+"\t");

}else{

System.out.print(j+"\*"+i+"="+i\*j+"\t");

}

}

System.out.println();

}

}

6.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(质数)

代码：

publicstaticvoidtest9(){

intnum=0;

intnum1=0;

for(inti=101;i<=200;i++){

for(intj=2;j<=i/2;j++){

if(i%j==0){

num++;

}

}

if(num==0){

num1++;

System.out.println(i);

}

num=0;

}

System.out.println("素数共有："+num1);

}

7.求1000以内的完全数

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。

例如，6=1+2＋328=1＋2＋4＋7＋14496=1+2+4+8＋16+31+62＋124，先计

算所选取的整数a(a的取值1~1000)的因数，将各因数累加于m，若m等于a，

则可确认a为完全数。

代码：

publicstaticvoidtest10(){

inttemp=0;

intnum=0;

for(inti=2;i<=1000;i++){

for(intj=1;j<i;j++){

if(i%j==0){

temp+=j;

}

}

if(i==temp){

num++;

System.out.println(i);

}

temp=0;

}

System.out.println("1000以内完全数共有："+num+"个");

}