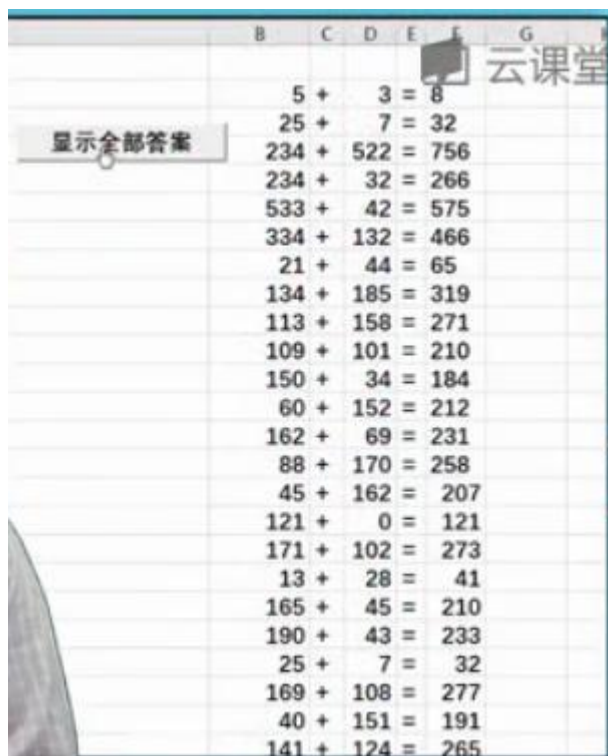


## 基础篇 第十一回

### 单变量难解二维表，双循环突破小周天

前面的几节课中，我们介绍了循环结构的一般用法，我们可以一次的处理成千上万行的数据。

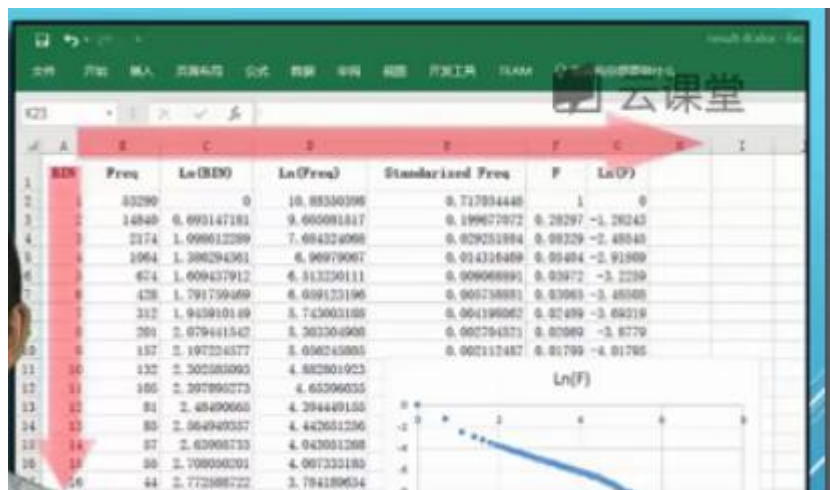


	B	C	D	E	F	G
		5 +	3 =	8		
		25 +	7 =	32		
显示全部答案		234 +	522 =	756		
		234 +	32 =	266		
		533 +	42 =	575		
		334 +	132 =	466		
		21 +	44 =	65		
		134 +	185 =	319		
		113 +	158 =	271		
		109 +	101 =	210		
		150 +	34 =	184		
		60 +	152 =	212		
		162 +	69 =	231		
		88 +	170 =	258		
		45 +	162 =	207		
		121 +	0 =	121		
		171 +	102 =	273		
		13 +	28 =	41		
		165 +	45 =	210		
		190 +	43 =	233		
		25 +	7 =	32		
		169 +	108 =	277		
		40 +	151 =	191		
		141 +	124 =	265		

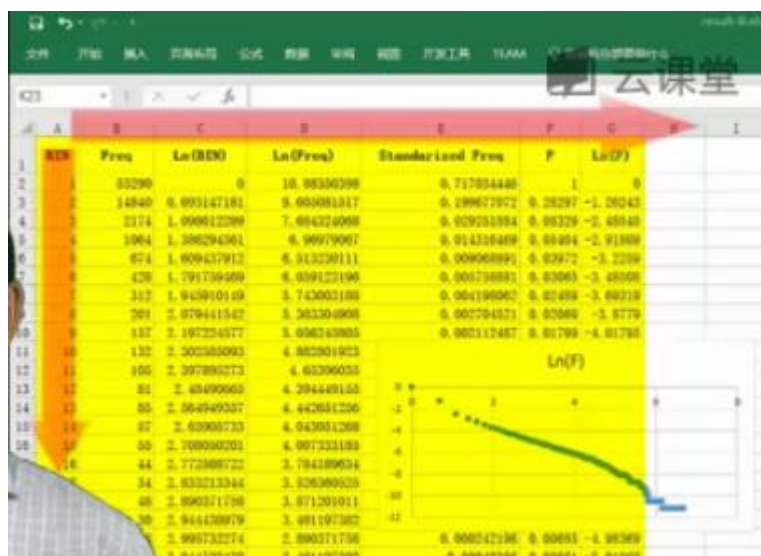
不过大家也注意到了在我们前面的例子中，我们所处理的单元格的都是集中在同一个列上，



而现实中 excel 是一个二维表格，有很多行，也有很多列。



那么怎样轻松的将我们的单元格中的所有数据一次性的，不管哪一行，那一列都处理掉呢？



这就是我们要介绍给大家的重要而又实用的技巧：**嵌套的循环结构**。什么叫做嵌套的循环结构，我们从一个例子开始。大家看这是一个材料清单：

	配料	淀粉	白糖	水	黄油	花生	鸡蛋	苏打	椰蓉	牛奶
1	产品									
2	切片面包	50	10	200	5	10	10	1	10	20
3	羊角面包	60	5	300	4	17	8	2	20	5
4	甜甜圈	40	20	100	7	0	10	3	15	20
5	松饼	60	20	200	10	1	20	2	5	30
6	马卡龙	50	25	150	5	10	15	1	10	30

计量单位： 千克

转换为磅 (LB)

里面列举了我们生产中需要的每一种需要的材料。对这些东西的重量我们用千克来计量，现在如果处于与外商合作的需要，我们需要把这些重量单位转换成英制的磅。那么我们按下按钮之后应该怎么执行代码呢？首先和之前一样，定义变量*i*代表这一行的行号：

```
Option Explicit

Sub changeUnit()

    Dim i

    i = 2

    Do While Cells(i, 1) <> ""

        i = i + 1
    Loop

    Cells(i, 9) = "磅"

End Sub
```

接下来就是进入循环之后，怎么执行我们的代码。比如我们进入了第*i*行，我们要处理第*i*行的那个单元格呢？

```
Option Explicit

Sub changeUnit()

    Dim i

    i = 2

    Do While Cells(i, 1) <> ""

        i = i + 1
    Loop

    Cells(i, 9) = "磅"

End Sub
```

产品	配料	白糖	水	黄油	花生	鸡蛋	苏打	椰香	其他
切片面包	110.4	22.08	44.15	11.04	22.08	2.208	22.08	44.15	
羊角面包	132.5	11.04	662.3	8.83	37.53	17.66	4.415	44.15	11.04
甜甜圈	88.3	44.15	220.8	15.45	0	22.08	6.623	33.11	44.15
松饼	132.5	44.15	22.08	2.208	44.15	11.04	66.23		
马卡龙	110.4	55.2	11.04	22.08	33.11	2.208	22.08	66.23	

最简单的方法如下：但需要优化：

```
Option Explicit

Sub changeUnit()

    Dim i

    i = 2

    Do While Cells(i, 1) <> ""

        Cells(i, 2) = Cells(i, 2) * 0.453
        Cells(i, 3) = Cells(i, 3) * 0.453
        Cells(i, 4) = Cells(i, 4) * 0.453
        Cells(i, 5) = Cells(i, 5) * 0.453
        Cells(i, 6) = Cells(i, 6) * 0.453
        Cells(i, 7) = Cells(i, 7) * 0.453
        Cells(i, 8) = Cells(i, 8) * 0.453
        Cells(i, 9) = Cells(i, 9) * 0.453
        Cells(i, 10) = Cells(i, 10) * 0.453

        i = i + 1
    Loop

    Cells(i, 9) = "磅"

End Sub
```

```
Option Explicit

Sub changeUnit()

    Dim i

    i = 2

    Do While Cells(i, 1) <> ""

        Cells(i, 2) = Cells(i, 2) * 0.453
        Cells(i, 3) = Cells(i, 3) * 0.453
        Cells(i, 4) = Cells(i, 4) * 0.453
        Cells(i, 5) = Cells(i, 5) * 0.453
        Cells(i, 6) = Cells(i, 6) * 0.453
        Cells(i, 7) = Cells(i, 7) * 0.453
        Cells(i, 8) = Cells(i, 8) * 0.453
        Cells(i, 9) = Cells(i, 9) * 0.453
        Cells(i, 10) = Cells(i, 10) * 0.453

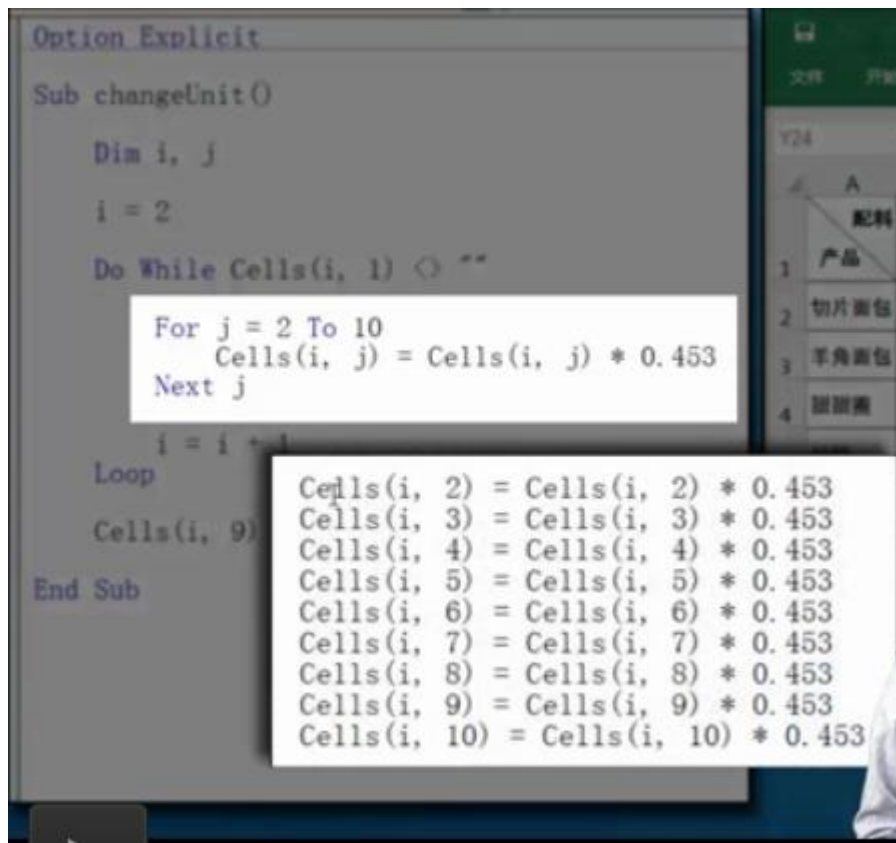
        i = i + 1
    Loop

    Cells(i, 9) = "磅"

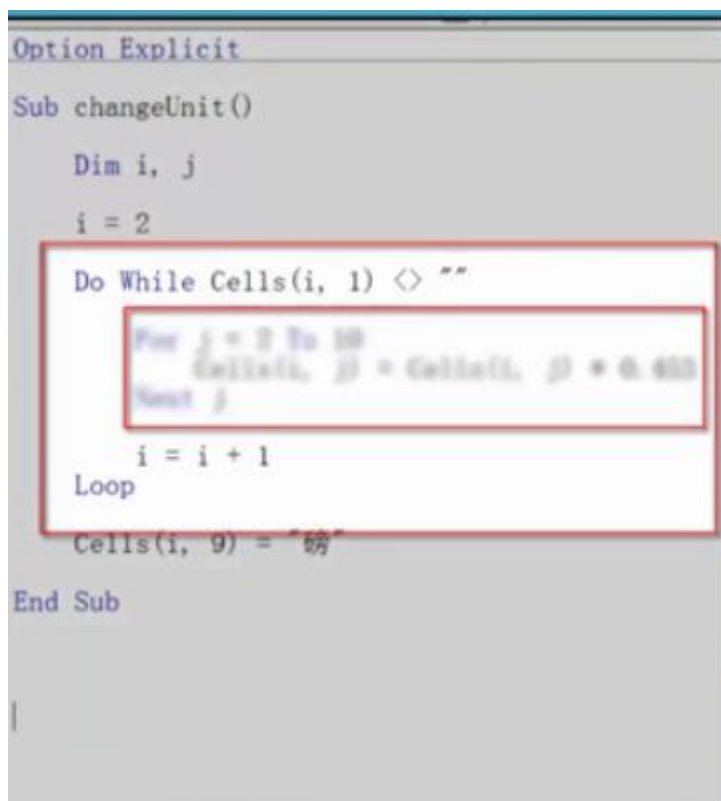
End Sub
```

规则明确

机械重复



看到的就是典型的二套的嵌套循环：

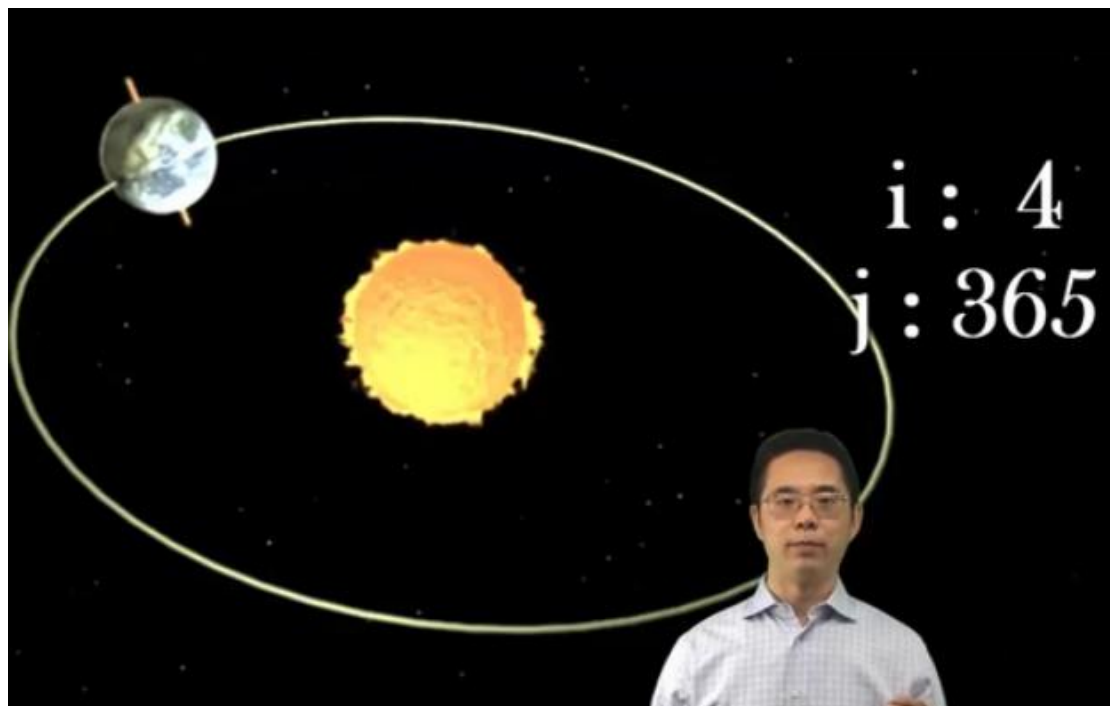


外边一层负责行号的变化，里面的一层负责列号的变化，当行号和列号都执

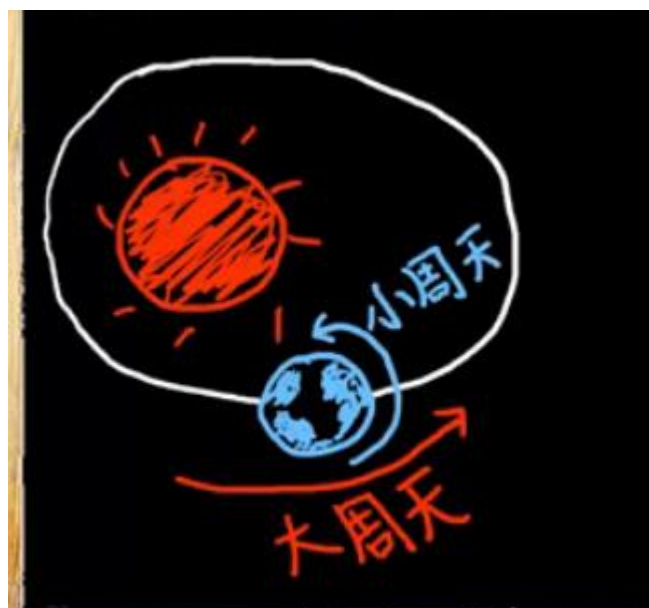


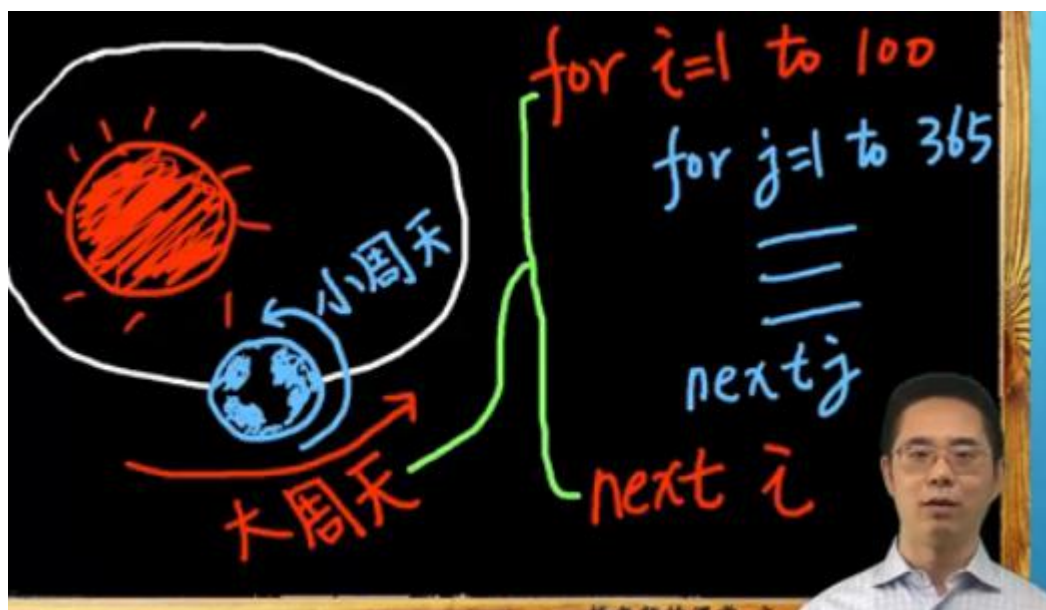
行完了之后，我们在执行比例转换操作。大家应该能理解嵌套循环结构的流程了。如果大家还是有点困惑，可以类比一下地球与太阳的公转系统，未免这层循环就相当于地球绕太阳的公转，每转一个循环就是一年。

而里面的循环就是地球的自转，一转就是一天，外面的循环走一次，里面的循环就会走 365 次。



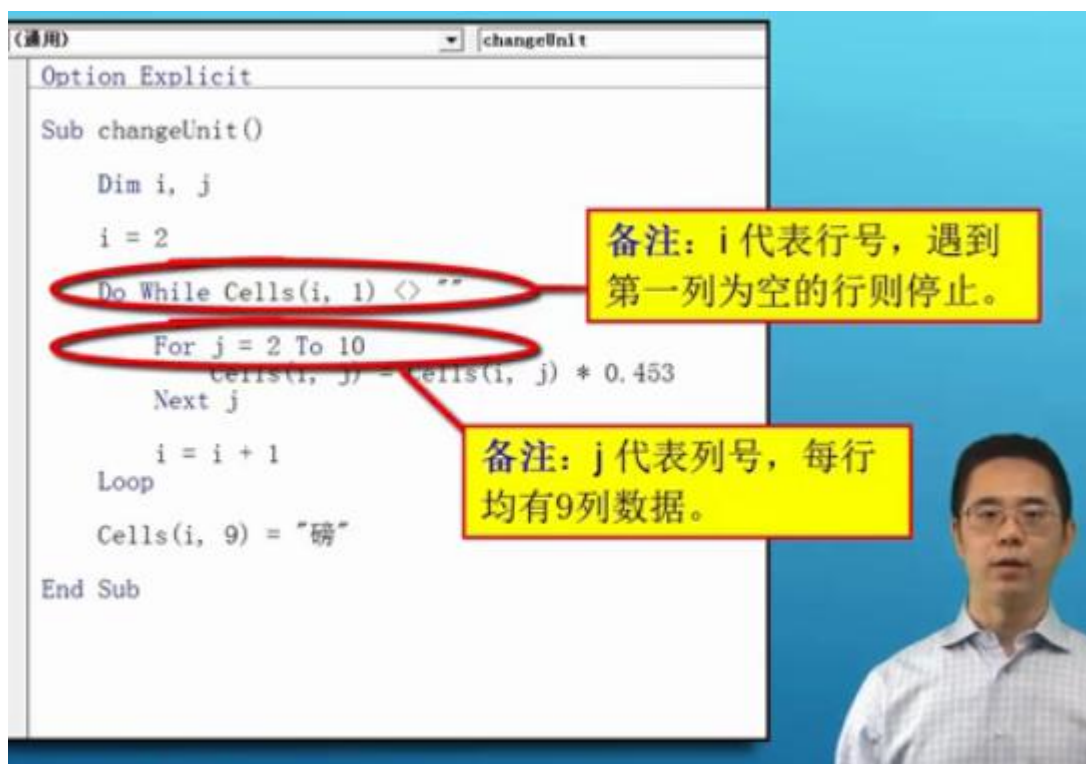
古代的说法：公转是大周天，自传是小周天。





但是一方面多重循环让代码变得更加强大，但是另一方面也令代码变得更加复杂难懂。

我们同时要养成非常好的写注释的习惯，



怎么写？在 VBA 中就是三个字母 REM，（remark 的缩写）



```
Option Explicit

Sub changeUnit()

    Dim i, j

    i = 2

    Rem i代表行号，遇到第一列为空的行则停止
    Do While Cells(i, 1) <> ""

        Rem j代表列号，每行都是9列
        For j = 2 To 10
            Cells(i, j) = Cells(i, j) * 0.453
        Next j

        i = i + 1
    Loop

    Cells(i, 9) = "磅"

End Sub
```

由于使用 REM 比较麻烦，所以使用了它的替代符号，也就是单引号。

```
(通用) changeUnit

Option Explicit

Sub changeUnit()

    Dim i, j

    i = 2

    ' i代表行号，遇到第一列为空的行则停止
    Do While Cells(i, 1) <> ""

        ' j代表列号，每行都是9列
        For j = 2 To 10
            Cells(i, j) = Cells(i, j) * 0.453
        Next j


        i = i + 1
    Loop

    Cells(i, 9) = "磅"

End Sub
```

写注释看起来是比较麻烦，感觉没什么用的工作，但其实是非常重要的，因为长一点的程序，因为不写注释，那么过了一段时间，我们就很难去读懂它，更别说去修改它了。





```
Option Explicit
Sub changeUnit()
    Dim i, j, ws
    'ws代表工作簿中的每一张工作表
    For Each ws In Worksheets
        i = 2
        ' i代表行号，遇到第一列为空的行则停止
        Do While ws.Cells(i, 1) <> ""
            ' j代表列号，每行都是9列
            For j = 2 To 10
                ws.Cells(i, j) = ws.Cells(i, j) * 0.453
            Next j
            i = i + 1
        Loop
        ws.Cells(i, 9) = "磅"
    Next ws
End Sub
```

这样的话，我们就可以把一个工作簿中的所有的工作表都处理完成。