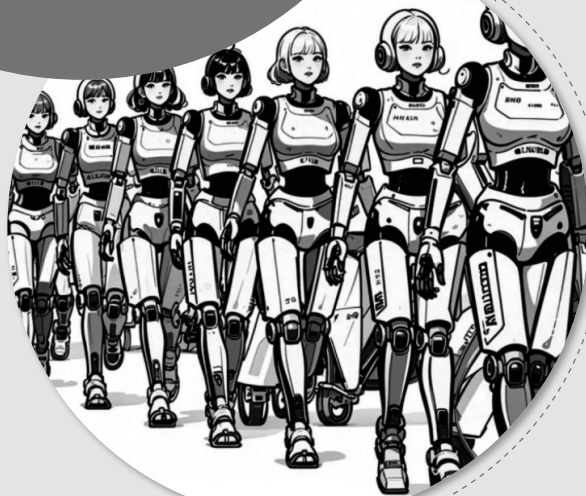


第 5 章

嵌套循环



本章学习嵌套循环的用法，先来看看典型实例。

例 5.1 九九乘法表 (1)

任务描述

输出乘法“九九表”。

程序代码如下。

Python

源程序

```
for i in range(1,10):
    for j in range(1,10):
        print(i,'*',j,'=',i*j,' ',sep='',end='')
    print('')
```

程序注解

- `print(end='')`: `print` 函数默认在结尾会换行, 如果不想换行, 则将结尾设为空字符串''。
- `print('')`: 输出一个空字符串, 目的是使用默认的换行。

运行结果

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

C++

源程序

```
#include <iostream>
#include <iomanip>
using namespace std;

int main()
{
    int a,b;
```

```

for(b=1;b<=9;b++)
{
    for(a=1;a<=9;a++)
        cout<<a<<"* "<<b<<"="<<a*b<<" ";
    cout<<endl;
}
}

```

运行结果

与 Python 相同。

可以看到，输出结果并没有对齐。可以在输出函数参数中加入输出宽度格式符。

Python

```
print(i, '*', j, '=', '%2d'%(i*j), ' ', sep='', end='')
```

C++

```
cout<<a<<"* "<<b<<"="<<setw(2)<<a*b<<" ";
```

运行结果

```

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

```

例 5.2 九九乘法表 (2)

任务描述

我们小时候背过的“九九表”通常只有上述结果的下半部，我们可以将内循环的范围 $1 \sim 9$ 改为 $1 \sim i$ 。程序代码如下。

Python

源程序

```

for i in range(1,10):
    for j in range(1,i+1):
        print(i, '*', j, '=', '%2d'%(i*j), ' ', sep='', end='')

```

```
print('')
```

运行结果

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

C++

源程序

```
#include <iostream>
#include <iomanip>
using namespace std;

int main()
{
    int a,b;
    for(b=1;b<=9;b++)
    {
        for(a=1;a<=b;a++)
            cout<<a<<"* "<<b<<"="<<setw(2)<<a*b<<" ";
        cout<<endl;
    }
}
```

运行结果

与 Python 相同。

本章要点

- 注意变循环范围的用法。

练习 5 输出三角形阵列

任务描述

在窗口中输出如下页所示的三角形阵列。

1

1 2 1

1 2 3 2 1

1 2 3 4 3 2 1

1 2 3 2 1

1 2 1

1