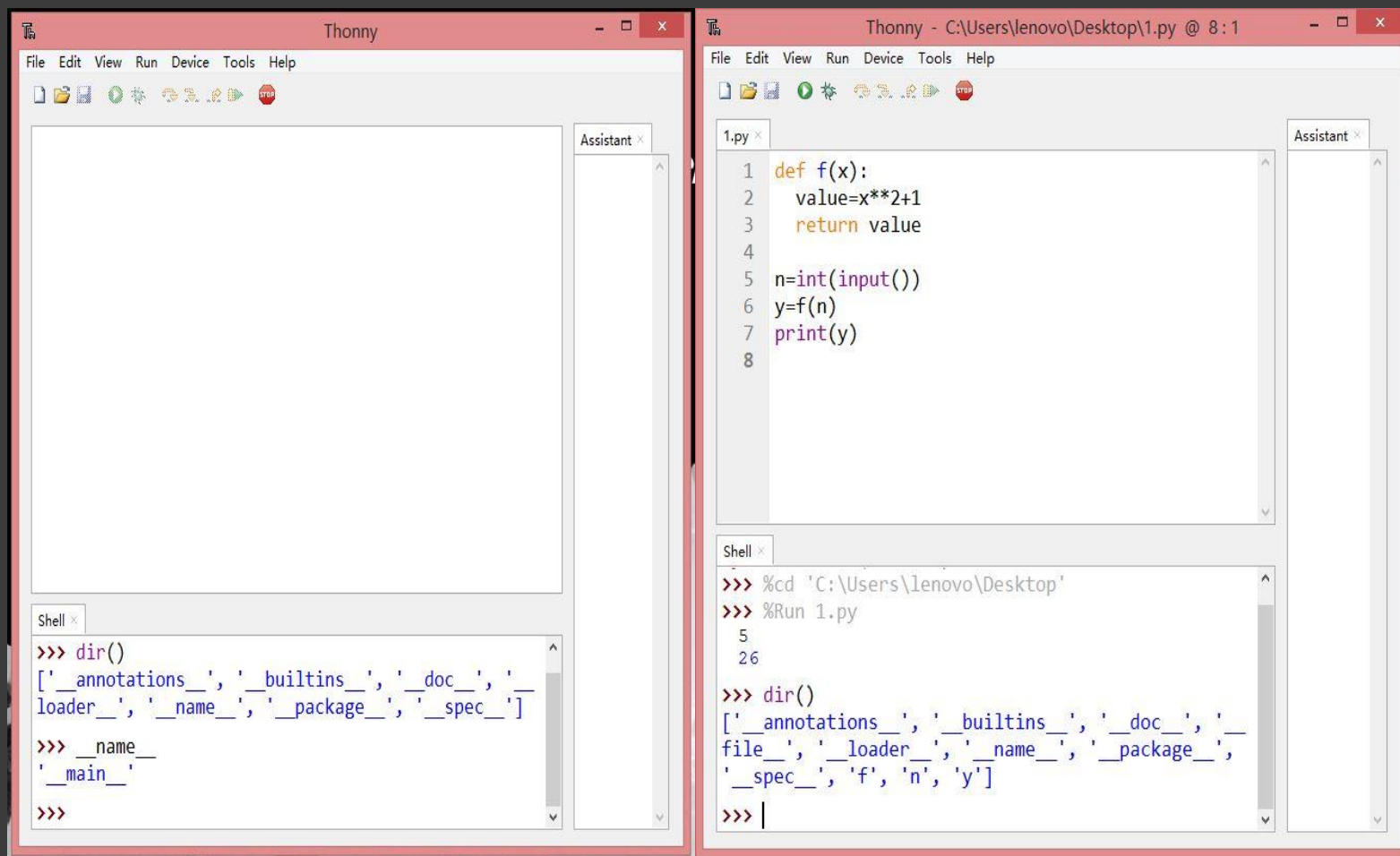


6.4 命名空间和作用域

1. 变量可被访问范围称为变量的作用域，也称为变量命名空间或变量名字空间。Python程序用命名空间区分不同空间的相同名字。
2. Python解释器启动时建立一个全局命名空间,全局变量就放在这个空间,还建立内置命名空间 (built-in namespace), 记录所有标准常量名、标准函数名等。在全局命名空间中定义的变量是全局变量。
3. 每一个函数定义自己的命名空间，函数内部定义的变量是局部变量。如果在一个函数中定义一个变量x，在另外一个函数中也定义x变量，因为是在不同的命名空间，所以两者指代的是不同的变量。可以通过多种方式获取其他命名空间的变量。

全局命名空间



局部变量与全局变量

- ◎ Python语言规定赋值即定义。“var=1”赋值语句定义了变量“var”并赋值为1
- ◎ 全局变量：定义在函数外，作用域是整个程序。
- ◎ 局部变量：定义在函数内，作用域是函数内部。形参也是局部变量

局部变量与全局变量同名

- def scope():
- var1=1
- print("函数内部打印结果")
- print(var1,var2)
- var1=10
- var2=20
- scope()
- print("函数外部打印结果")
- print(var1,var2)

Thonny - C:\Users\lenovo\Desktop\1.py @ 6:1

```
1 def scope():
2     var1=1
3     print("函数内部打印结果")
4     print(var1,var2)
5
6 var1=10
7 var2=20
8 scope()
9 print("函数外部打印结果")
10 print(var1,var2)
11
```

Variables

Name	Value
scope	<function scope at 0x037f33d8>
var1	10
var2	20

scope()

```
def scope():
    var1=1
    print("函数内部打印结果")
    print(var1,var2)
```

Local variables

Name	Value
var1	1

Shell

```
函数外部打印结果
10 20

>>> %Debug 1.py

函数内部打印结果
1 20
函数外部打印结果
10 20

>>> %Debug 1.py
```

global关键字

如希望在函数中使用全局变量，而不是创建局部变量，需要用global关键字声明

- ◎ def scope():
- ◎ global var1
- ◎ var1=1
- ◎ print("函数内部打印结果")
- ◎ print(var1,var2)
- ◎ var1=10
- ◎ var2=20
- ◎ scope()
- ◎ print("函数外部打印结果")
- ◎ print(var1,var2)

练习一结果是什么？

- ⦿ `x=100`
- ⦿ `def f():`
- ⦿ `y=x`
- ⦿ `x=0`
- ⦿ `print(x)`
- ⦿ `f()`