

1.1 将序列分解为单独的变量¶

问题¶

现在有一个包含 N 个元素的元组或者是序列，怎样将它里面的值解压后同时赋值给 N 个变量？

解决方案¶

任何的序列（或者是可迭代对象）可以通过一个简单的赋值操作来分解为单独的变量。唯一的要求就是变量的总数和结构必须与序列相吻合。

代码示例：

```
>>> p = (4, 5)
>>> x, y = p
>>> x
4
>>> y
5
>>>
>>> data = [ 'ACME', 50, 91.1, (2012, 12, 21) ]
>>> name, shares, price, date = data
>>> name
'ACME'
>>> date
(2012, 12, 21)
>>> name, shares, price, (year, mon, day) = data
>>> name
'ACME'
>>> year
2012
>>> mon
12
>>> day
21
>>>
```

如果元素的数量不匹配，会得到一个错误提示。

代码示例：

```
>>> p = (4, 5)
>>> x, y, z = p
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ValueError: need more than 2 values to unpack
>>>
```

讨论¶

不仅仅只是元组或列表，只要对象是可迭代的，就可以执行分解操作。包括字符串，文件对象，迭代器和生成器。

代码示例：

```
>>> s = 'Hello'
>>> a, b, c, d, e = s
>>> a
'H'
>>> b
'e'
>>> e
'o'
>>>
```

有时候，你可能只想解压一部分，丢弃其他的值。对于这种情况 Python 并没有提供特殊的语法。但是你可以使用任意变量名去占位，到时候丢掉这些变量就行了。

代码示例：

```
>>> data = [ 'ACME', 50, 91.1, (2012, 12, 21) ]
>>> _, shares, price, _ = data
>>> shares
50
>>> price
91.1
>>>
```

你必须保证你选用的那些占位变量名在其他地方没被使用到。