

## 1.17 从字典中提取子集¶

### 问题¶

你想构造一个字典，它是另外一个字典的子集。

### 解决方案¶

最简单的方式是使用字典推导。比如：

```
prices = {
    'ACME': 45.23,
    'AAPL': 612.78,
    'IBM': 205.55,
    'HPQ': 37.20,
    'FB': 10.75
}
# Make a dictionary of all prices over 200
p1 = {key: value for key, value in prices.items() if value > 200}
# Make a dictionary of tech stocks
tech_names = {'AAPL', 'IBM', 'HPQ', 'MSFT'}
p2 = {key: value for key, value in prices.items() if key in tech_names}
```

### 讨论¶

大多数情况下字典推导能做到的，通过创建一个元组序列然后把它传给 `dict()` 函数也能实现。比如：

```
p1 = dict((key, value) for key, value in prices.items() if value > 200)
```

但是，字典推导方式表意更清晰，并且实际上也会运行的更快些（在这个例子中，实际测试几乎比 `dict()` 函数方式快整整一倍）。

有时候完成同一件事会有多种方式。比如，第二个例子程序也可以像这样重写：

```
# Make a dictionary of tech stocks
tech_names = {'AAPL', 'IBM', 'HPQ', 'MSFT' }
p2 = { key:prices[key] for key in prices.keys() & tech_names }
```

但是，运行时间测试结果显示这种方案大概比第一种方案慢 1.6 倍。如果对程序运行性能要求比较高的话，需要花点时间去做计时测试。关于更多计时和性能测试，可以参考 14.13 小节。