

表示它的 Python 代码如下。

```
graph = {}
graph["you"] = ["alice", "bob", "claire"]
graph["bob"] = ["anuj", "peggy"]
graph["alice"] = ["peggy"]
graph["claire"] = ["thom", "jonny"]
graph["anuj"] = []
graph["peggy"] = []
graph["thom"] = []
graph["jonny"] = []
```

顺便问一句：键-值对的添加顺序重要吗？换言之，如果你这样编写代码：

```
graph["claire"] = ["thom", "jonny"]
graph["anuj"] = []
```

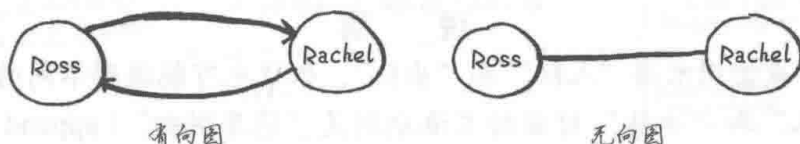
而不是这样编写代码：

```
graph["anuj"] = []
graph["claire"] = ["thom", "jonny"]
```

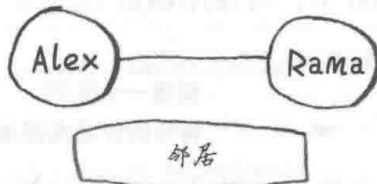
对结果有影响吗？

只要回顾一下前一章介绍的内容，你就知道没影响。哈希表是无序的，因此添加键-值对的顺序无关紧要。

Anuj、Peggy、Thom 和 Jonny 都没有外邻点，这是因为虽然有指向他们的箭头，但没有从他们出发指向其他人的箭头。这被称为有向图（directed graph），其中的关系是单向的。无向图（undirected graph）没有箭头。下面两个图是等价的。



在无向图中，没有内邻点和外邻点的概念，而可使用更简单的术语——邻居。



## 6.5 实现算法

先概述一下这种算法的工作原理。