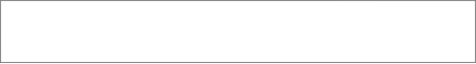
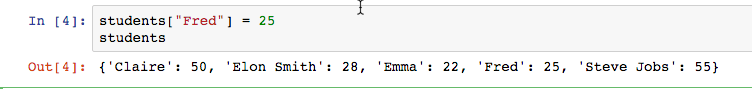
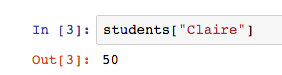
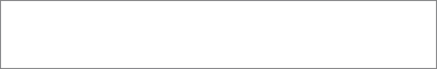
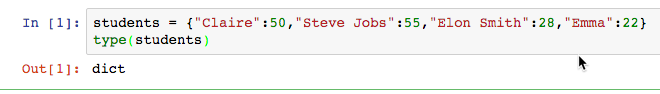
从这一课开始，我们将介绍Python中的另外一种数据类型Dictionary（词典）。

在日常生活中，我们也用过词典，左边是一个单词，而右边则是对单词的解释。在Python中，Dictionary与之类似。

不管怎样，先不要被名词吓到，我们直接用代码来说明它的作用。  
打开Jupyter Notebook，创建一个新的项目，并将其命名为dic。

假定我们要创建一个学生词典，用来保存学生的名字及其对应的年龄。那么可以用以下的方式：

在上面的代码中，我们使用花括号来定义了一个词典，在冒号的左侧是代表学生姓名的字符串，在冒号的右侧是代表学生年龄的数字变量，然后使用逗号将不同的值分开。



在Python中，每个具体的值被称为item，冒号左侧的“单词”被称为key，冒号右侧的内容被称为value。

如果我们希望了解某个key对应的value，应该怎么办呢？很简单，使用下面的方式：

可以看到，我们只需要在方括号中使用key，就可以获得value了。

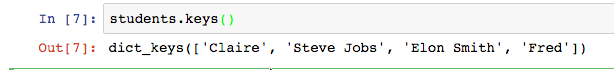
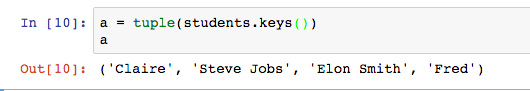
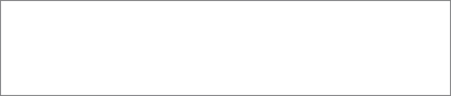
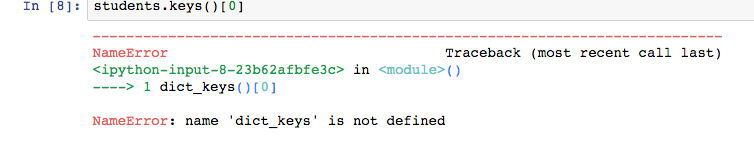
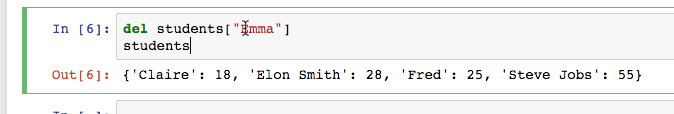
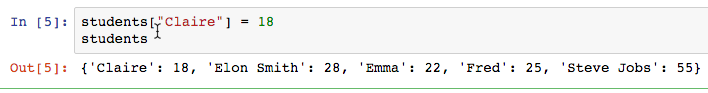
如果我们希望添加新的item，应该怎么办呢？用以下的方式：

可以看到，我们只需要给新的key赋值，就会将其自动添加到词典中了。

如果我们希望修改已有item的value，也是用类似的方式：

如果我们希望删除已有的item，怎么办呢？很简单，使用之前我们接触过的del：

如果我们希望获取词典中的所有key，可以用以下的方式：



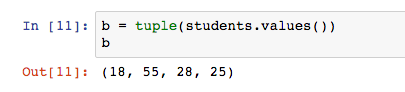
需要注意的是，所获取的词典中的key是不支持index的，所以下面的用法是错误的：

如果希望获取词典中的某个具体key，可以用以下的方式：

当然，这里用的是tuple，其实用list也是可以的~

对于词典中的value，也可以用类似的方式：

需要特别注意的是，和以往的列表与tuple都不同，词典中的元素是无序的。也就是说，我们不能使用index来获取词典中的某个key。



要获取词典内容的唯一方法就是使用key。

好了，本课的内容就到此结束了，我们下一课再见~