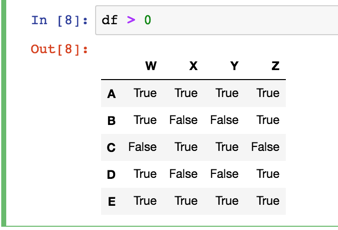
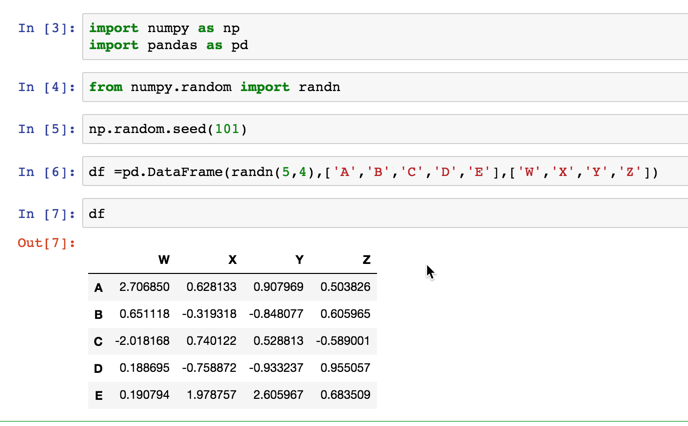
欢迎继续我们的学习。

在上一课的内容中，我们学习了Pandas中DataFrames的基本用法。

而在这一课的内容中，我们将继续学习DataFrames。

首先打开Anaconda和JupyterNotebook，然后打开DataFrames这个notebook。



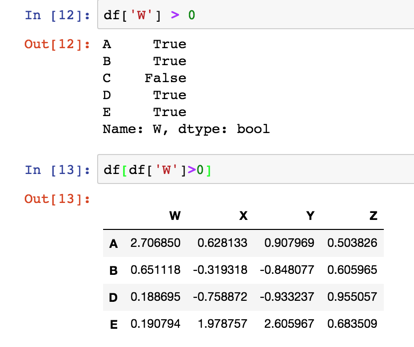
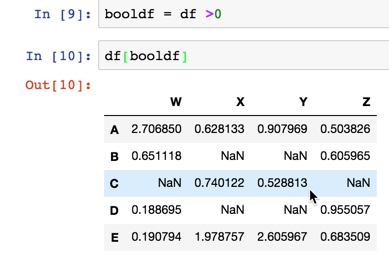
在代码中先创建一个基本的DataFrame对象：

接下来看看如何使用条件选择的方式获取特定的数据。

在以上的代码中，我们获得了矩阵中每个元素是否大于零。

在以上代码中，我们相当于定义了一个元素全部为布尔值的新DataFrame，并使用它来筛选出所有大于零的数据。

不过这种显示方式不是很完美，让我们换一种方式：

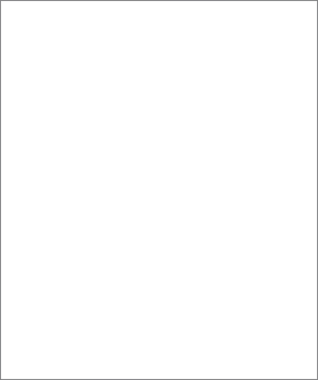
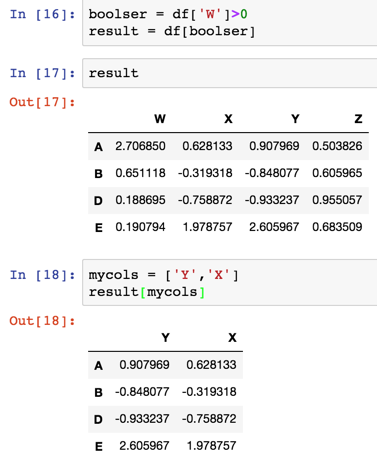
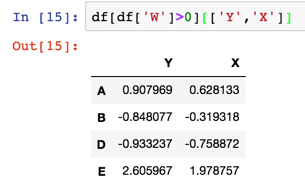


在以上代码中，我们获取了W列全部为正数的子矩阵。

需要注意的是，以上操作不会影响原始的DataFrame对象。

再来看一个复杂点的例子：

在以上的代码中，将两步操作合并，首先筛选出W列为正数的子矩阵，然后再筛选出仅包含Y和X两列的子矩阵。

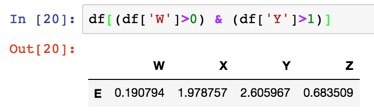
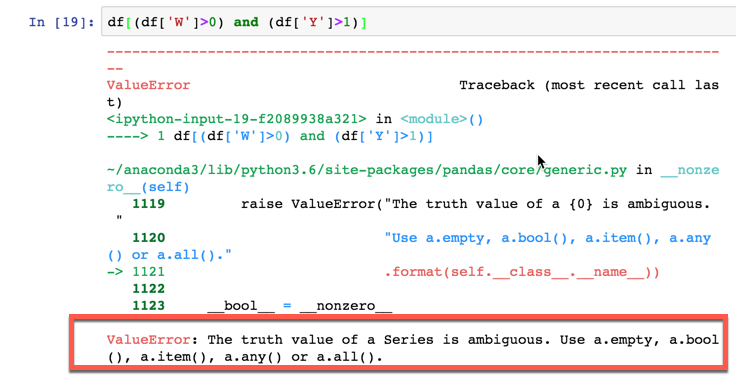


以上操作等同于下面的代码：

以上代码的作用跟刚才的一行代码是相同的。

随着我们对Pandas越来越熟悉，我们将更多的使用单行代码完成多步操作。如果你觉得哪里看不太懂了，可以用类似上面的方式将其拆解，以便理解。

接下来我们来了解如何使用多重条件选择。



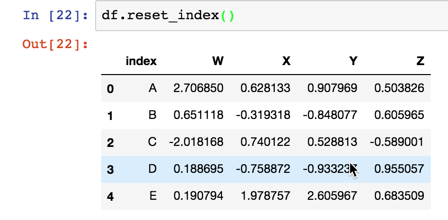
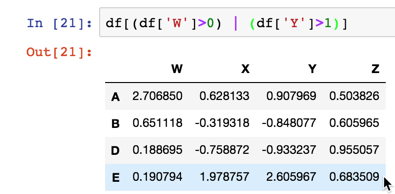
首先来看新手容易使用的错误用法：

错误的原因很明显，在python中and只能对布尔值进行运算，而不能直接对Series值进行运算。

正确的用法如下：

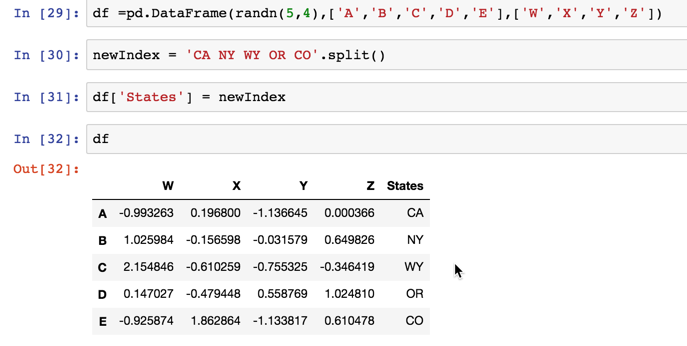
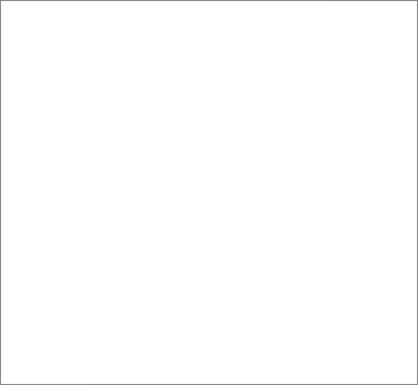
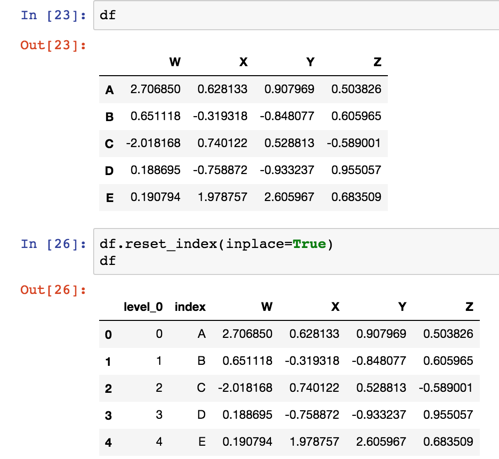
同样的，如果要使用逻辑或，那么需要使用|，而不是or:

接下来看看在DataFrame中如何操作index，如果我们希望使用数字编号的index，而非自定义的index,那么可以用以下方式：



不过这种操作不会影响原始的df，如果要直接修改df,那么需要使用以下方式：

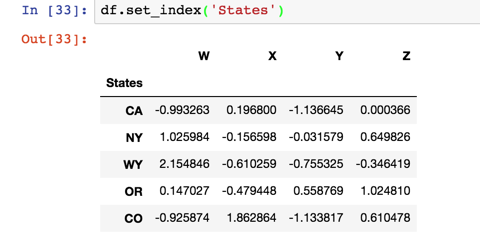
接下来再来看看如何设置新的index：



在以上代码中，我们给df添加了新的一列，接下来将States设置为新的index。

同样的，只有设置了inplace参数，才会真正更改df的值。

好了，本课的内容就到这里了。



我们下一课再见

讨论群-笨猫学编程QQ群：

375143733

知乎专栏：

<https://zhuanlan.zhihu.com/kidscoding>

新浪博客:

<http://blog.sina.com.cn/eseedo>

Github:

<https://github.com/eseedo>

个人网站：

<http://icode.ai/>