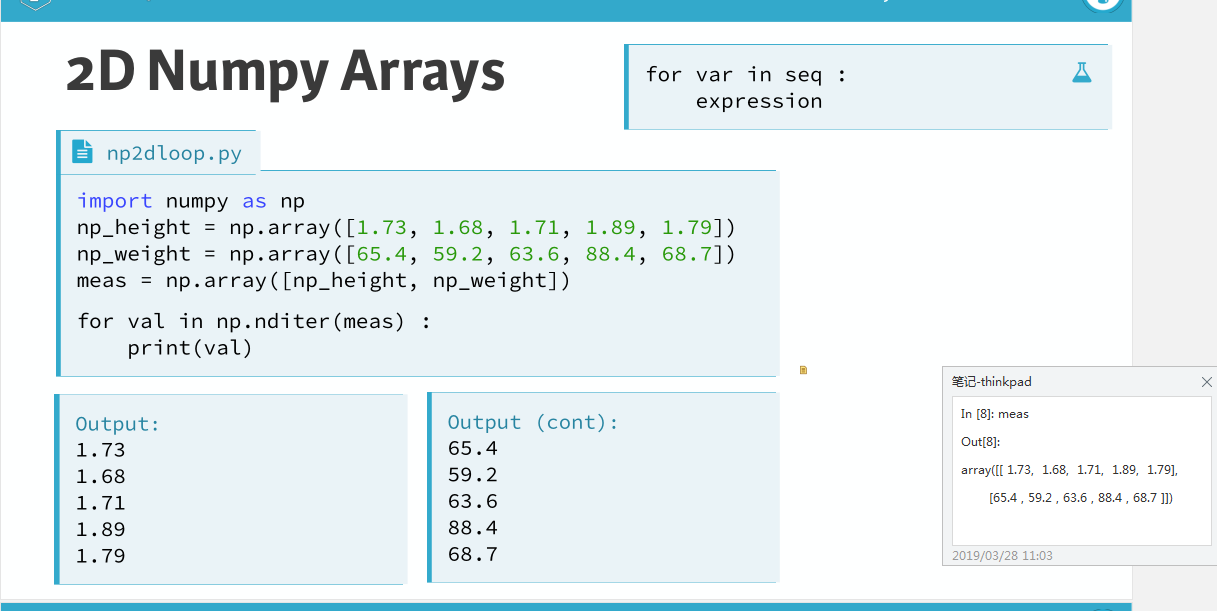
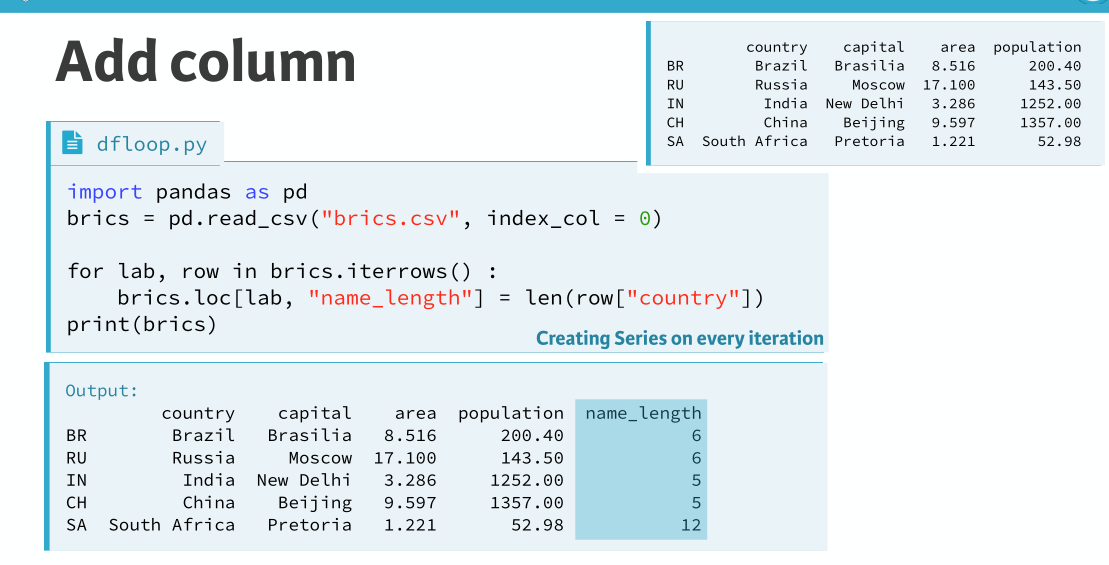
part2\_4

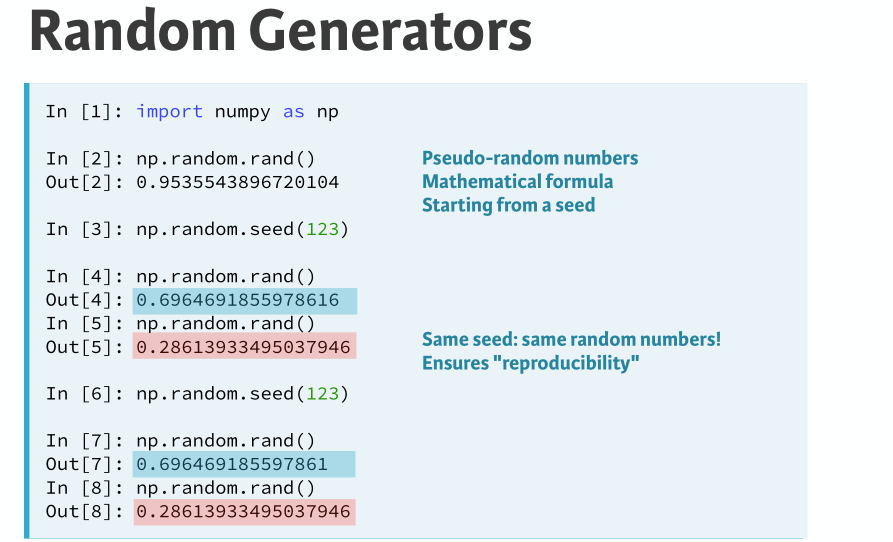




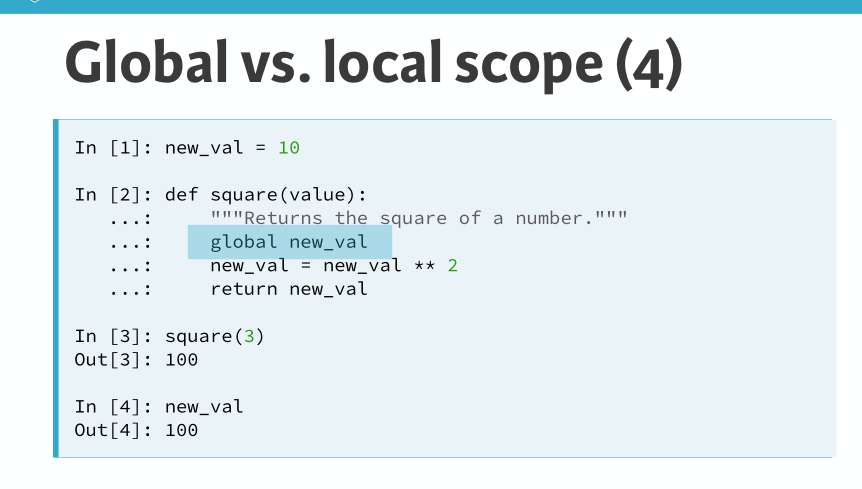
https://blog.csdn.net/jiangjiang\_jian/article/details/77540599

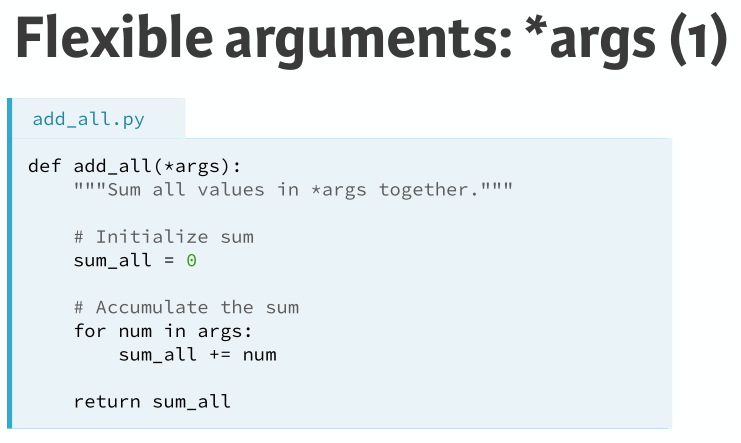
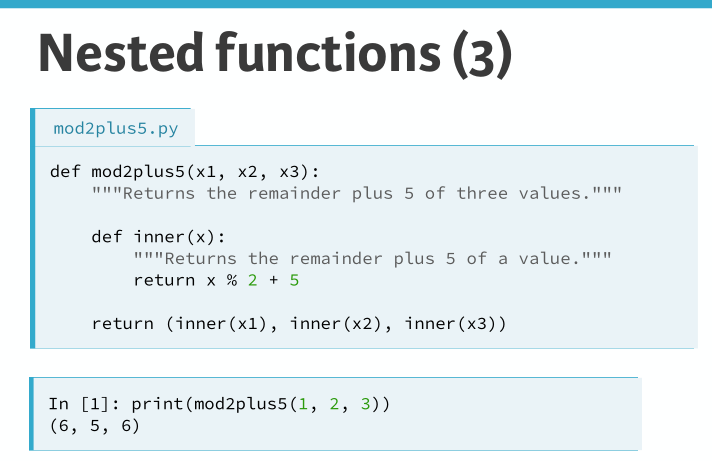
1. a = np.arange(6).reshape(2, 3)
2. **print** a
4. **for** x **in** np.nditer(a, op\_flags = ['readwrite']):
5. x[...] = 2\*x
7. **print** a
9. [[0 1 2]
10. [3 4 5]]
11. [[ 0  2  4]
12. [ 6  8 10]]

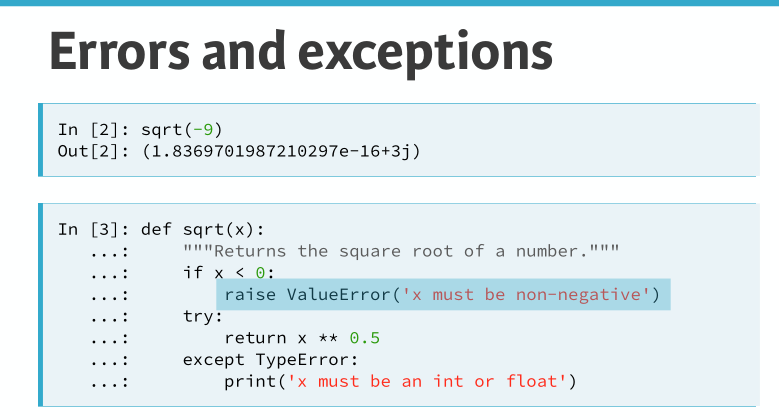




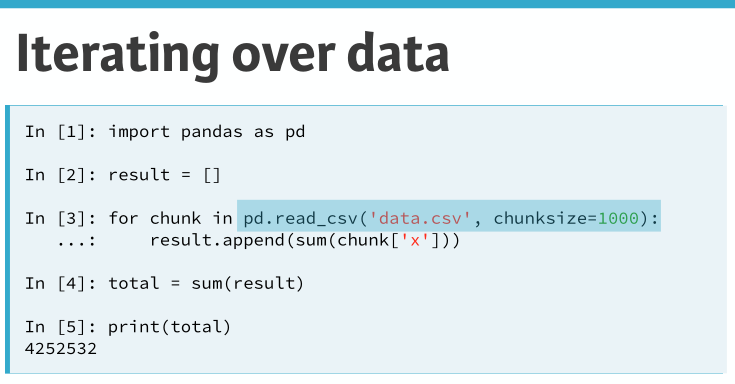
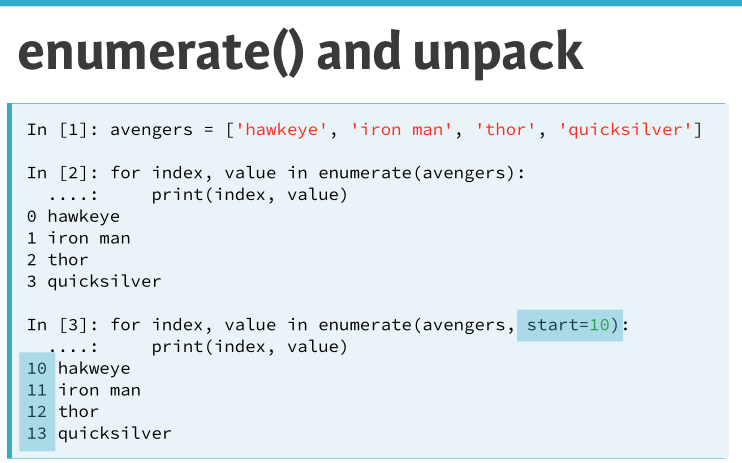
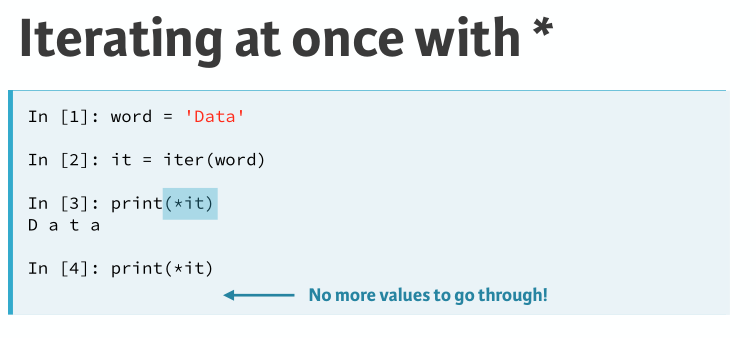
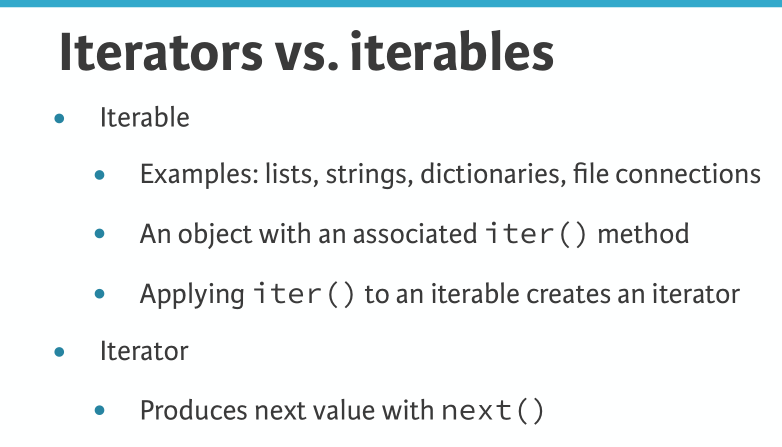
part3-2

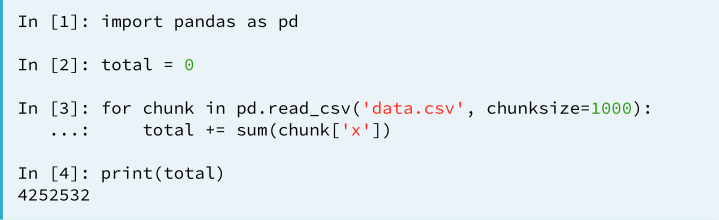






part4\_1



or:

part4\_2

In [7]: n=[2,3,4]

In [8]: n+1

TypeError: can only concatenate list (not "int") to list

In [9]: n+[1]

Out[9]: [2, 3, 4, 1]

In [10]: m=[1,1,1,1]

In [11]: n+m # list + list is a kind of concat

Out[11]: [2, 3, 4, 1, 1, 1, 1]

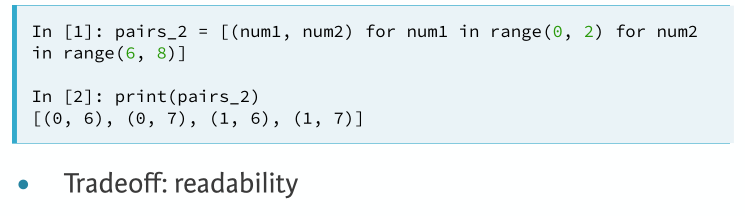
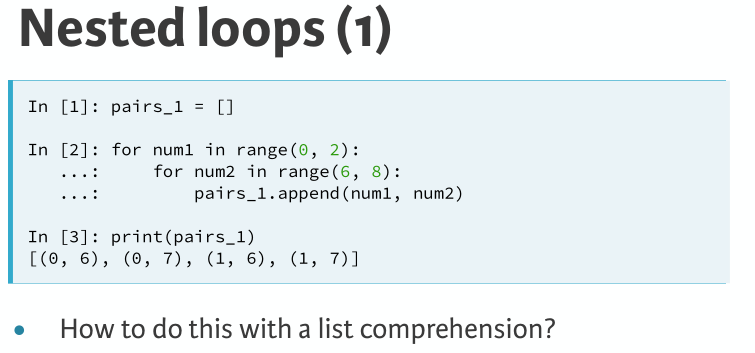
In [14]: k=[2,2,2,2]

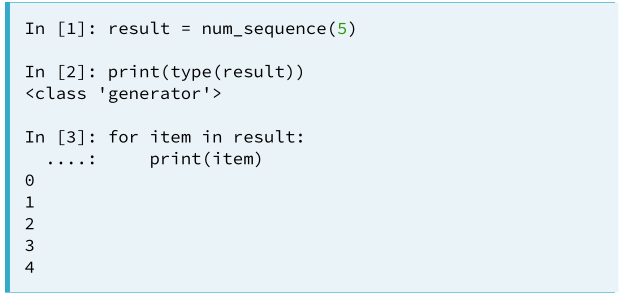
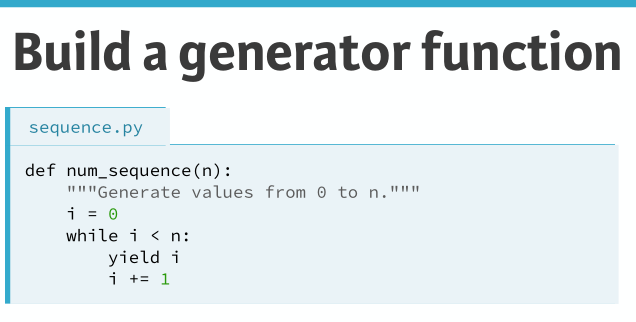
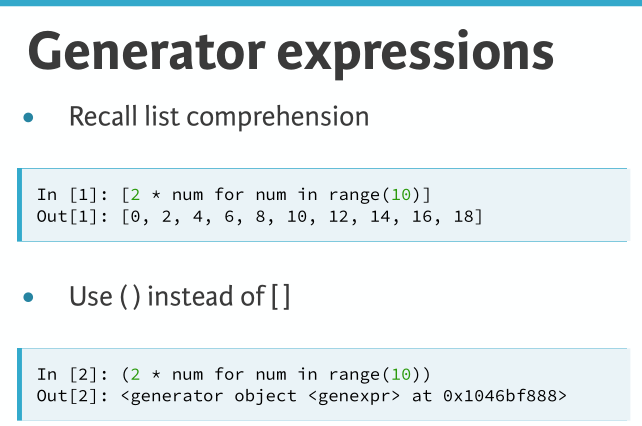
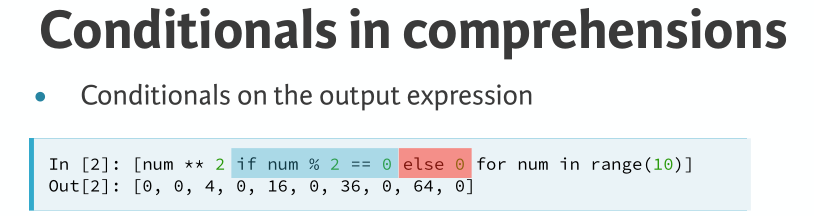
In [15]: np.add(k,m) # numpy calculate two list adding.

Out[15]: array([3, 3, 3, 3])

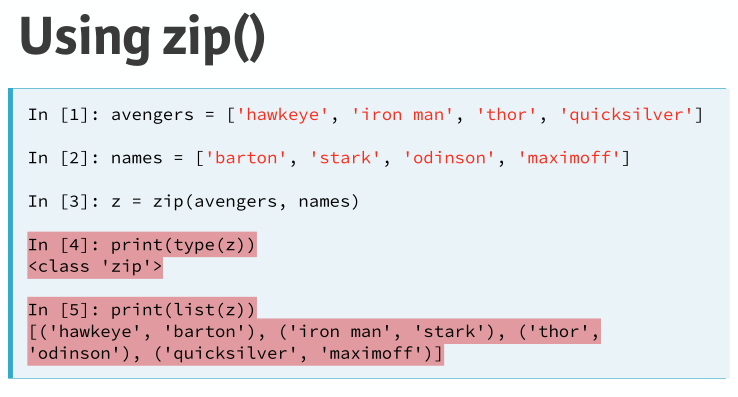
In [17]: np.add(k,10)

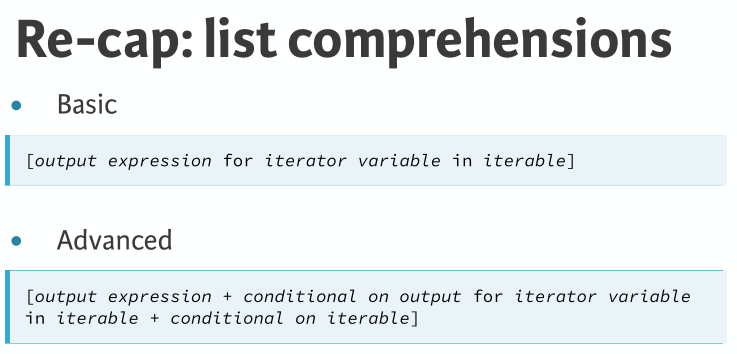
Out[17]: array([12, 12, 12, 12])



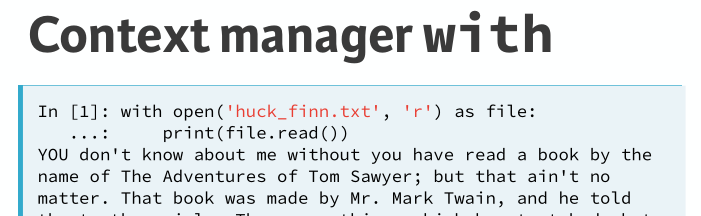


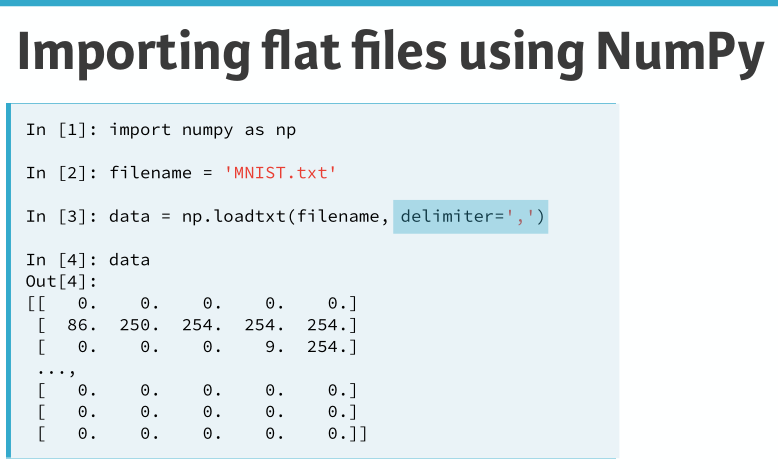
part4\_3





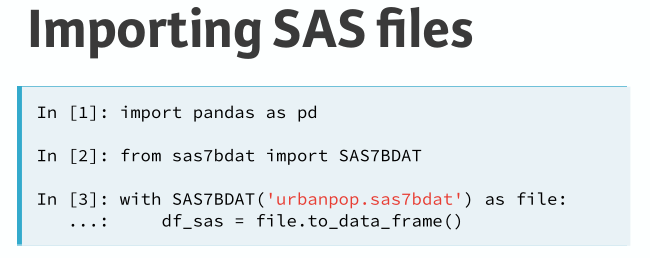
part5\_1



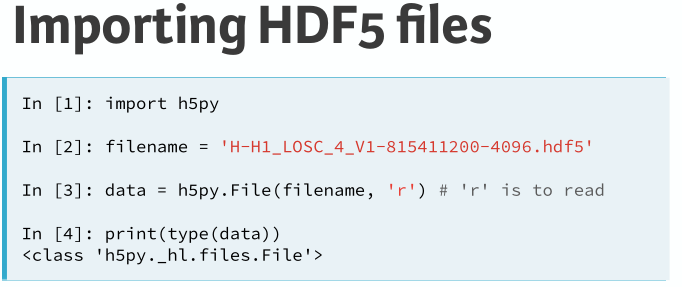


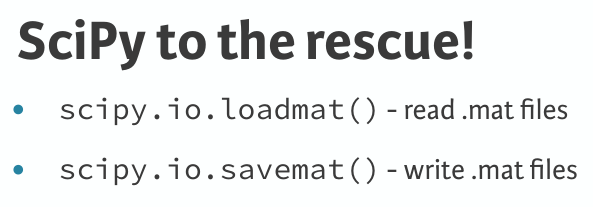
part5\_2

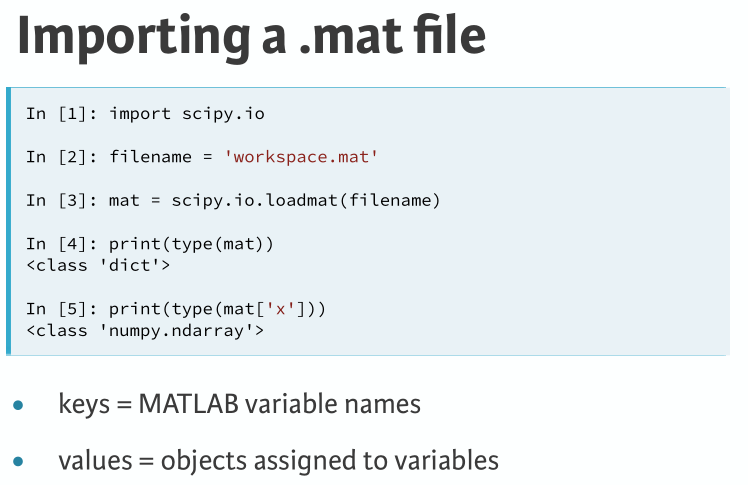




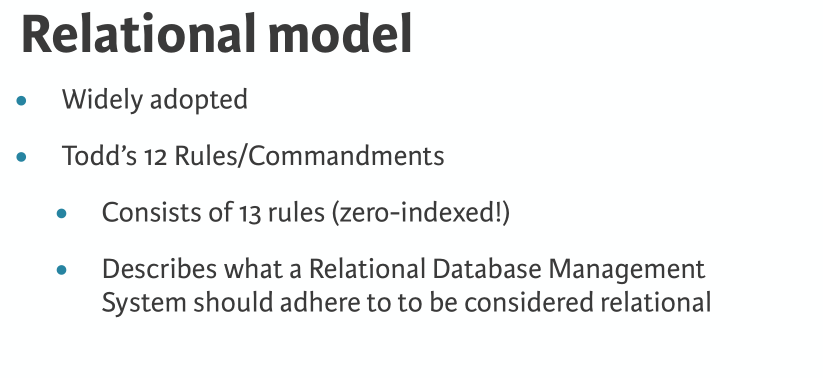


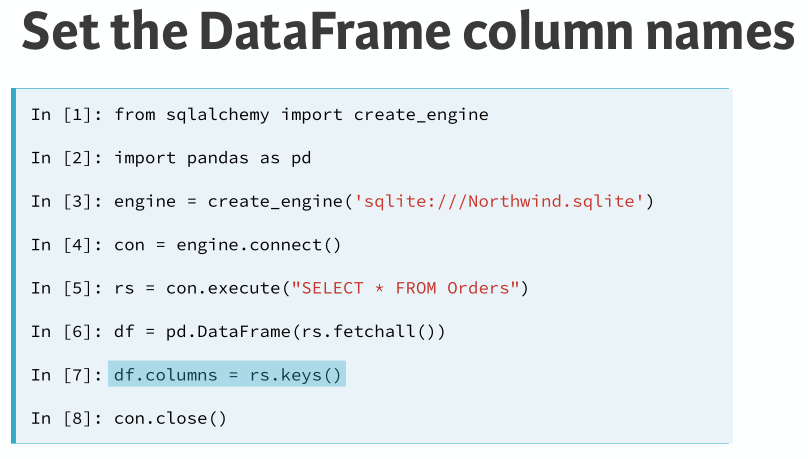


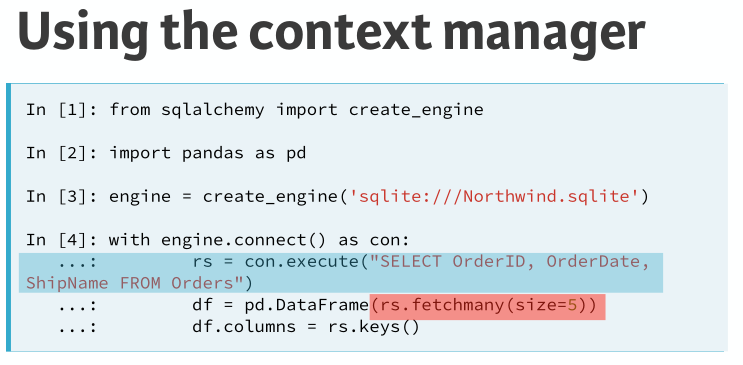


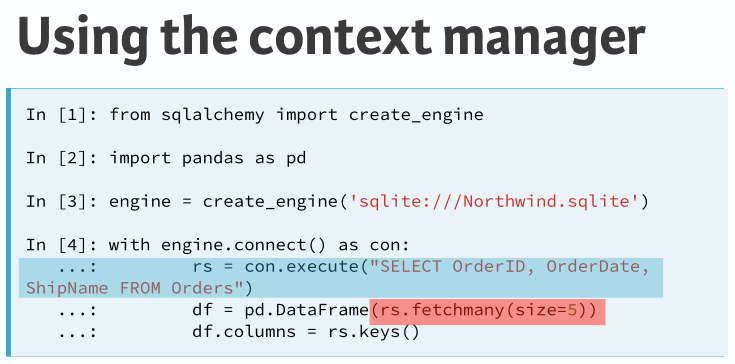


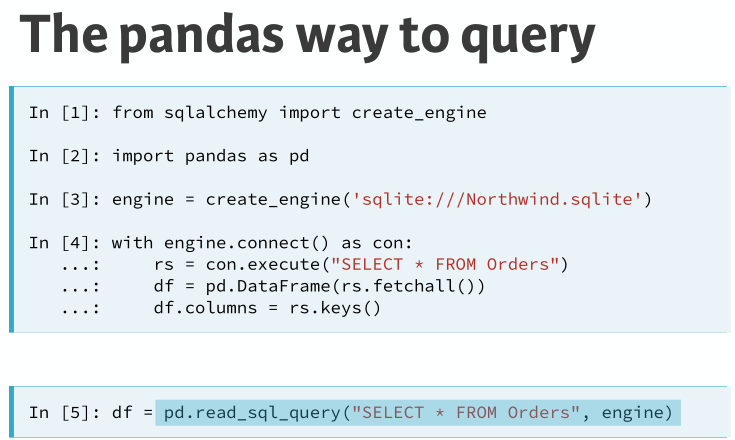
part5\_3

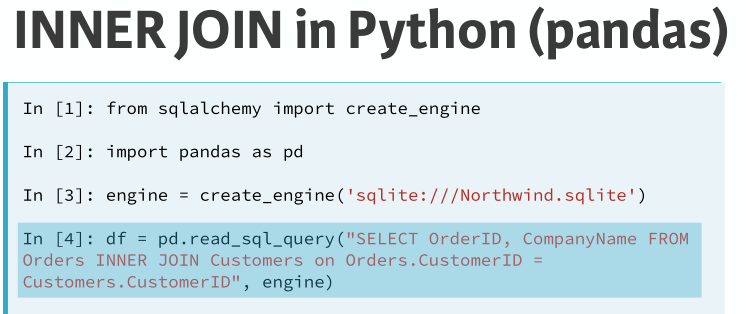




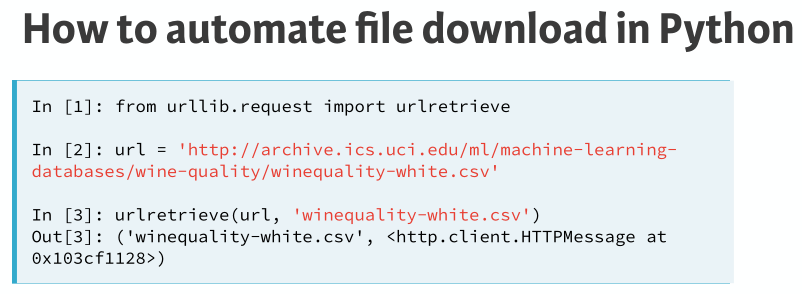








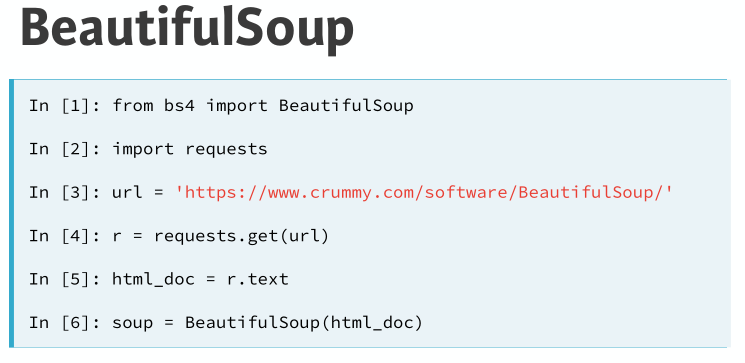
part6\_1

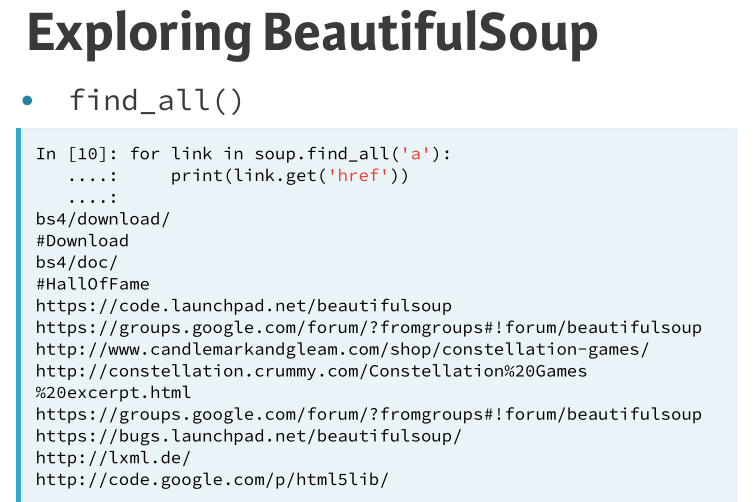








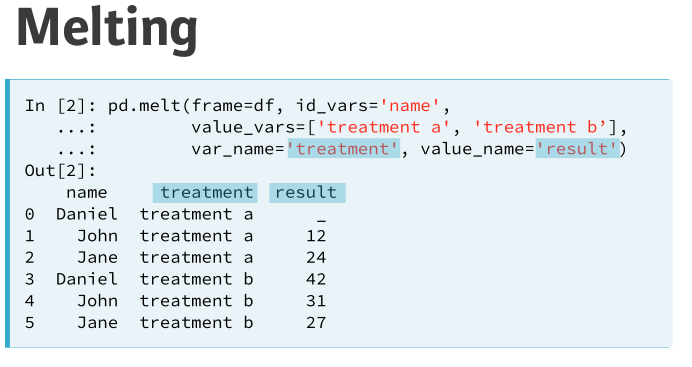


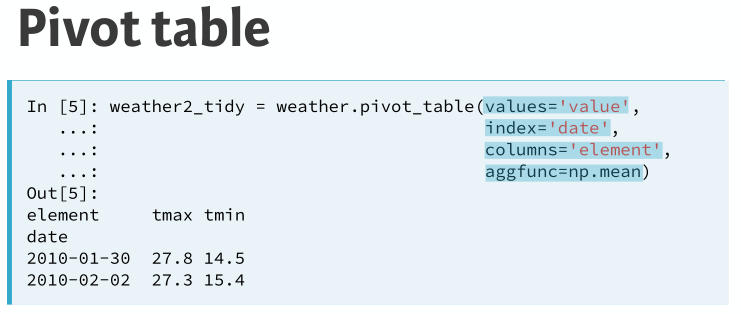


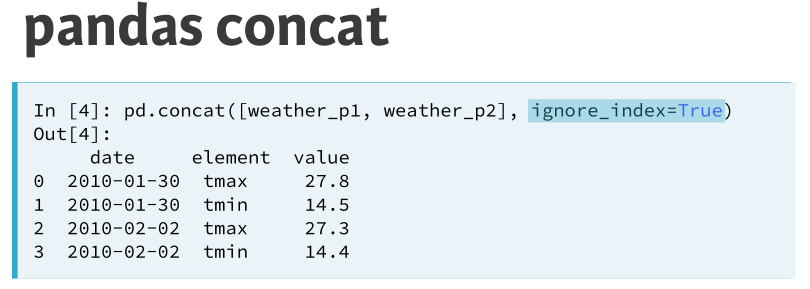
part6\_2

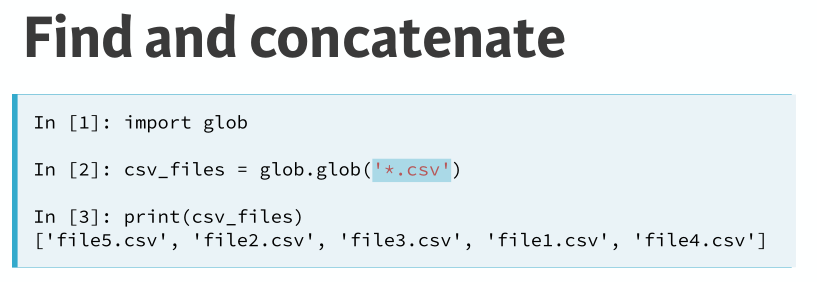


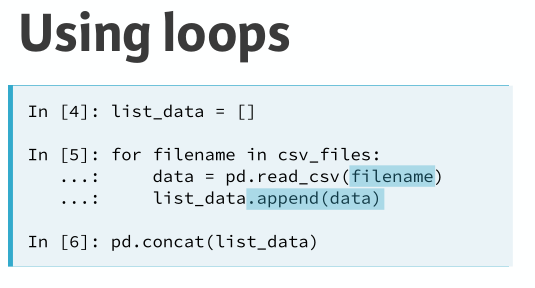
part7\_2

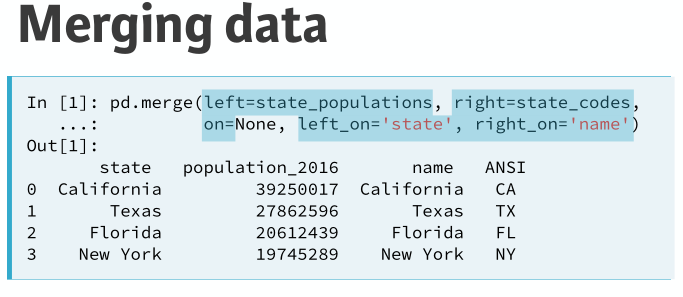






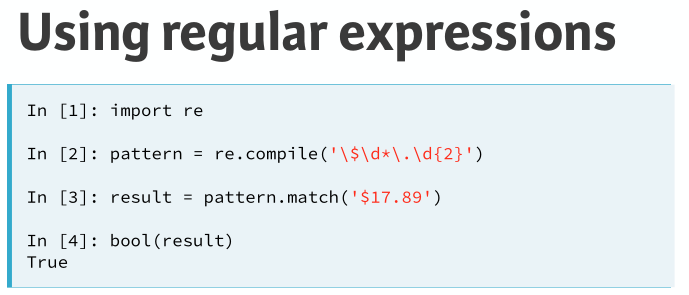




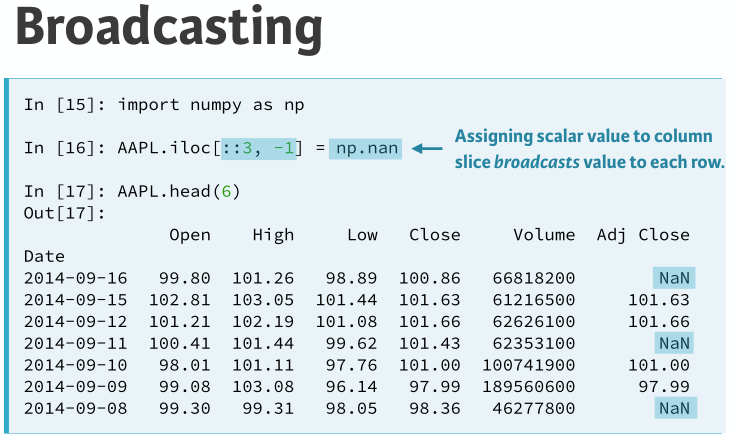


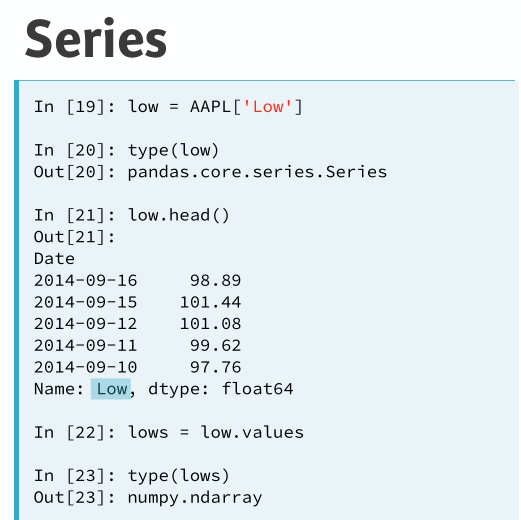
part7\_4

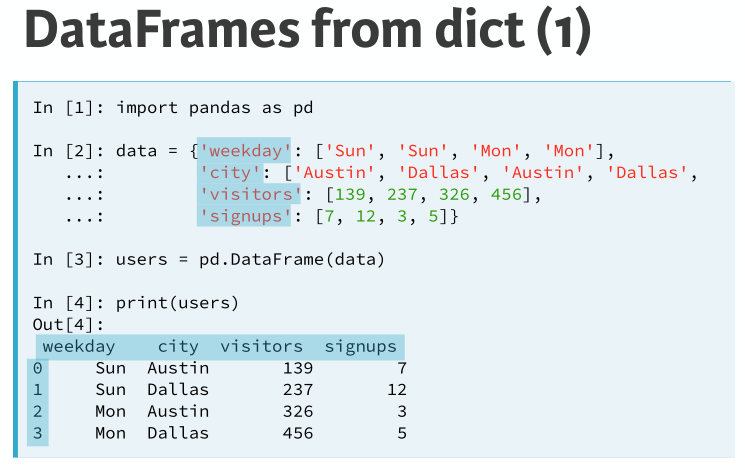




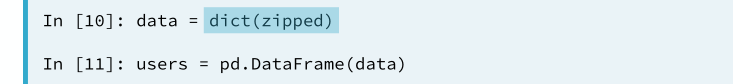
part8\_1

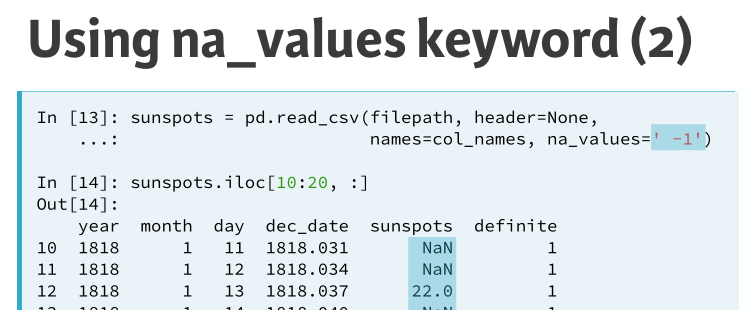
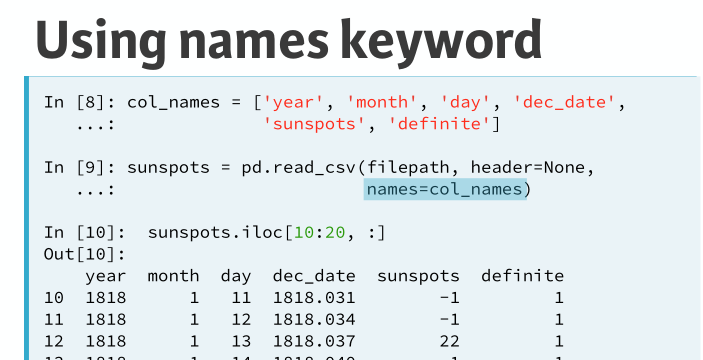


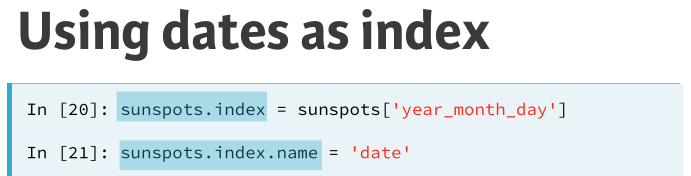
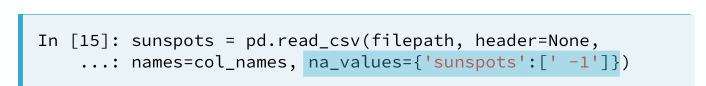


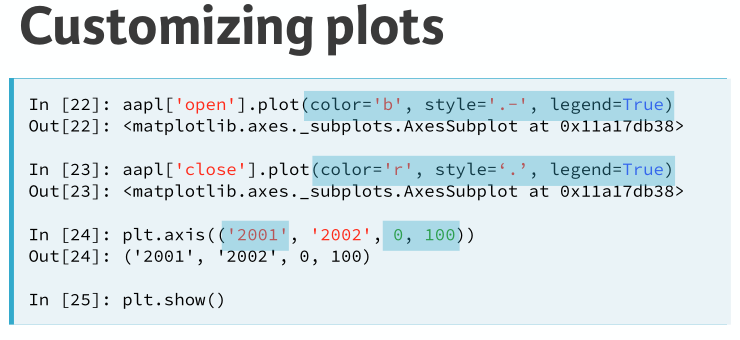


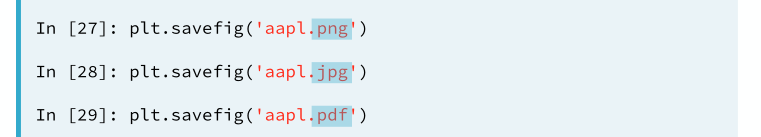




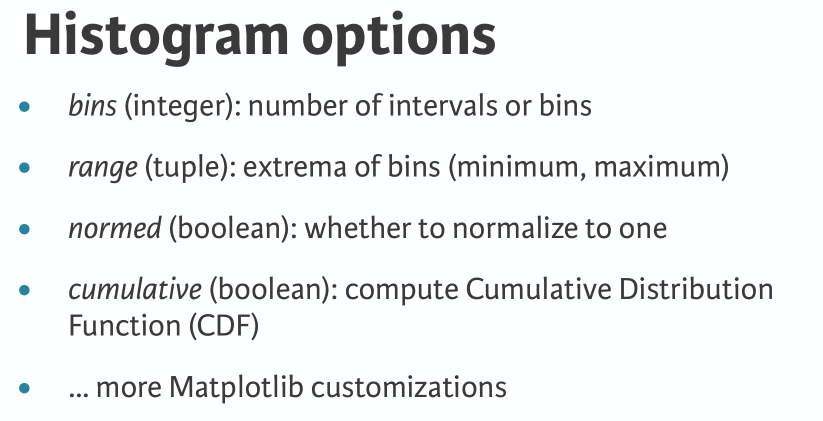


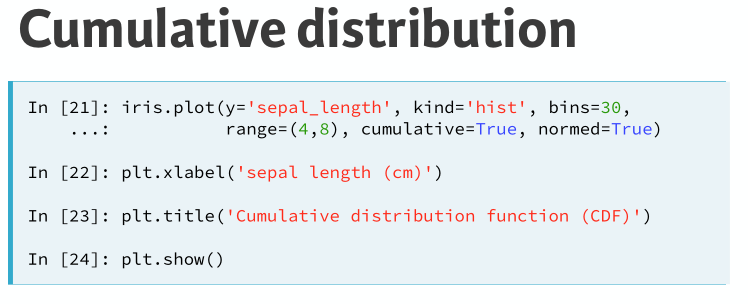


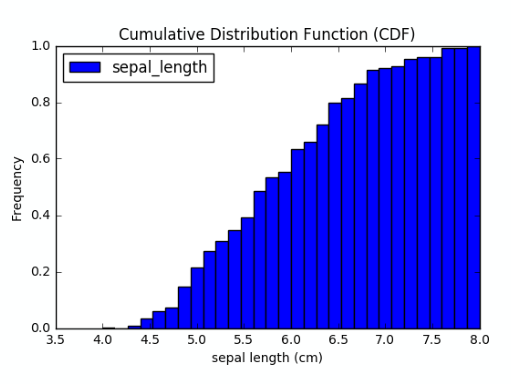


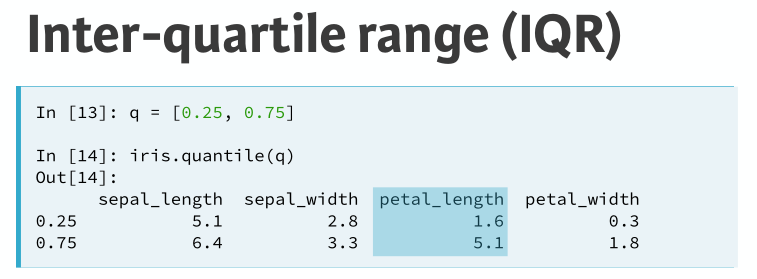


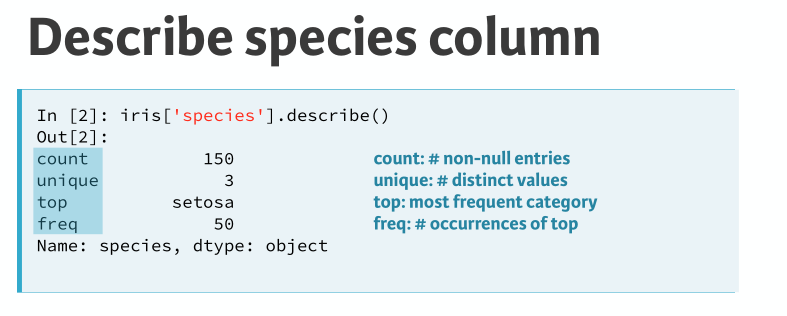
part8\_2

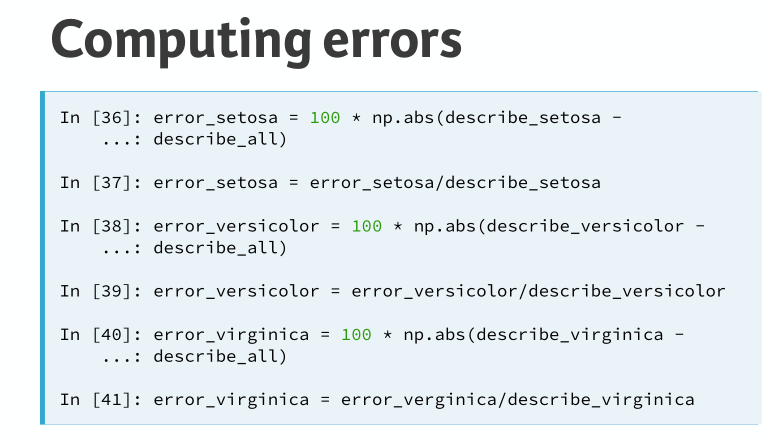




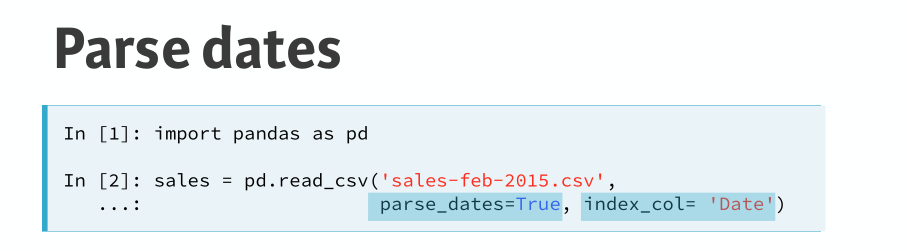


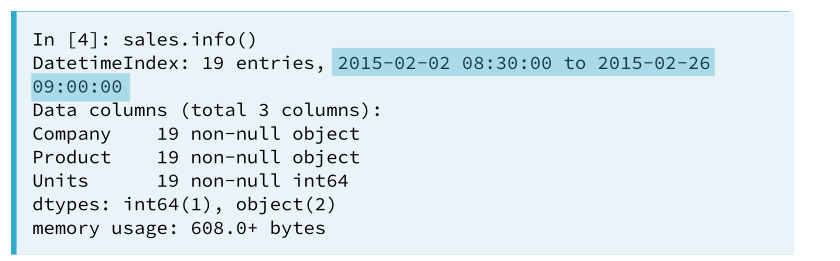


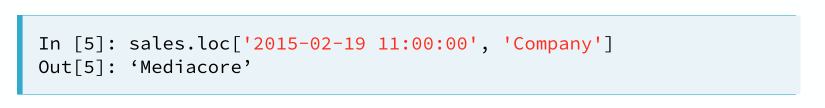


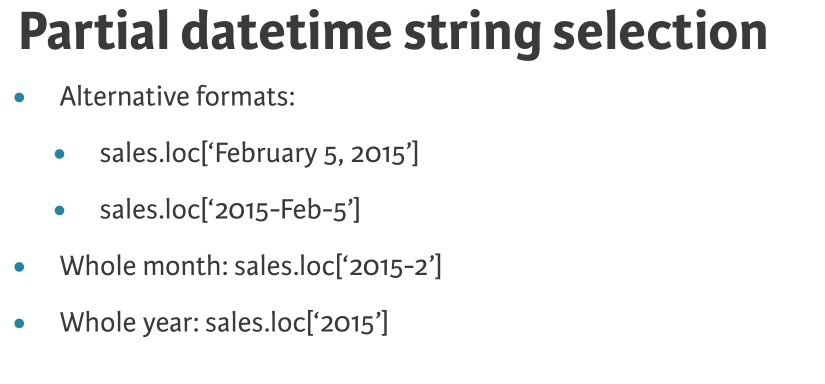


part8\_3

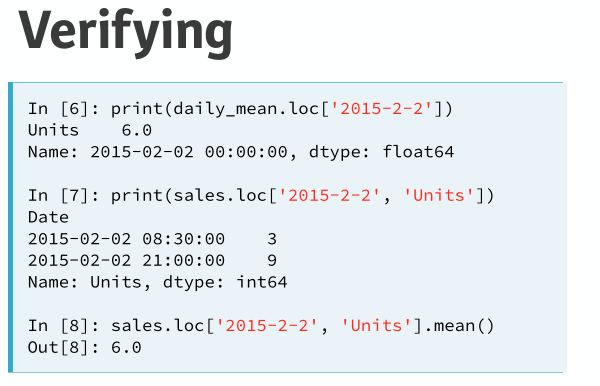




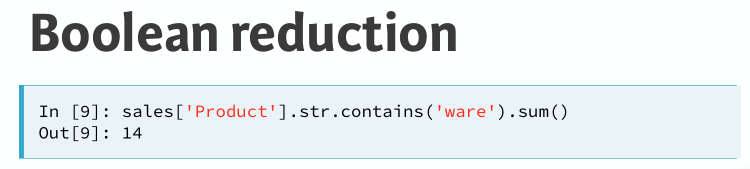


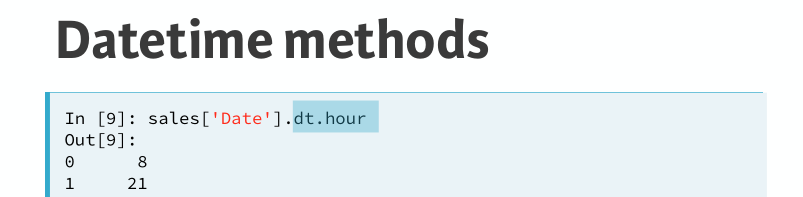


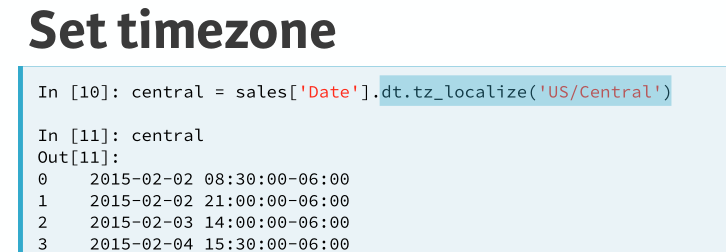


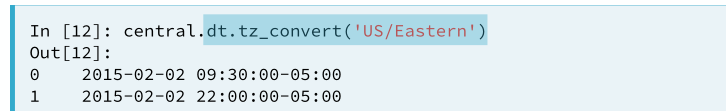








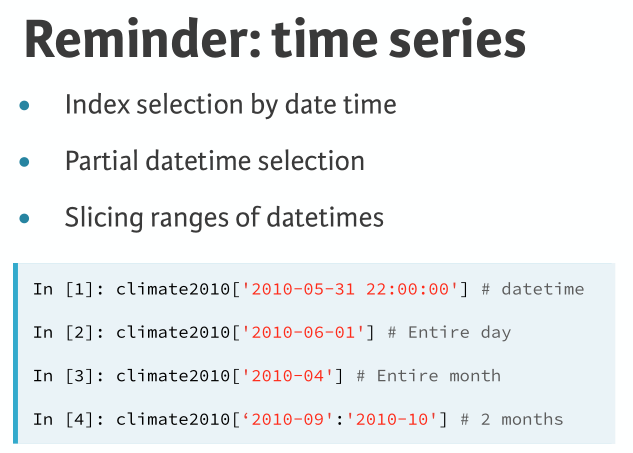




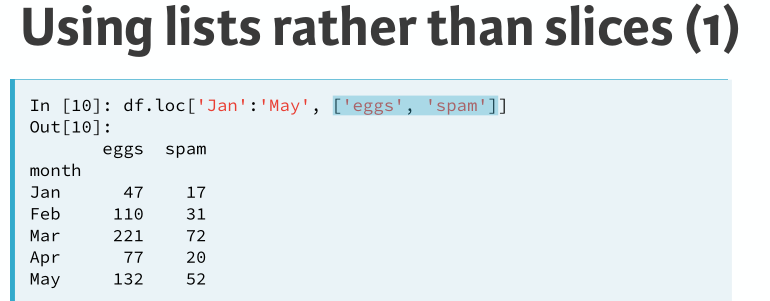


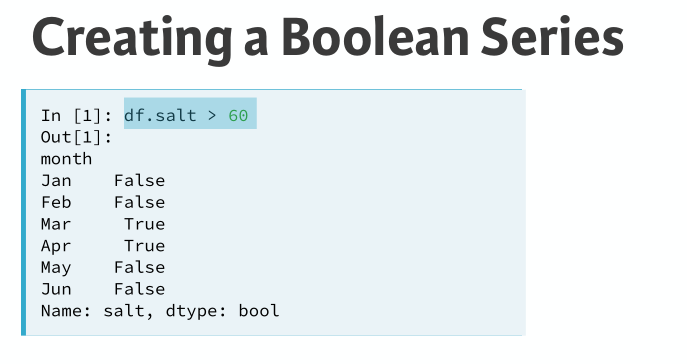


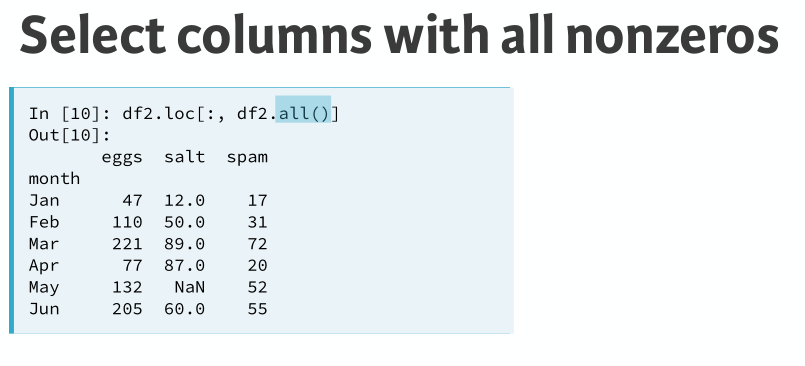
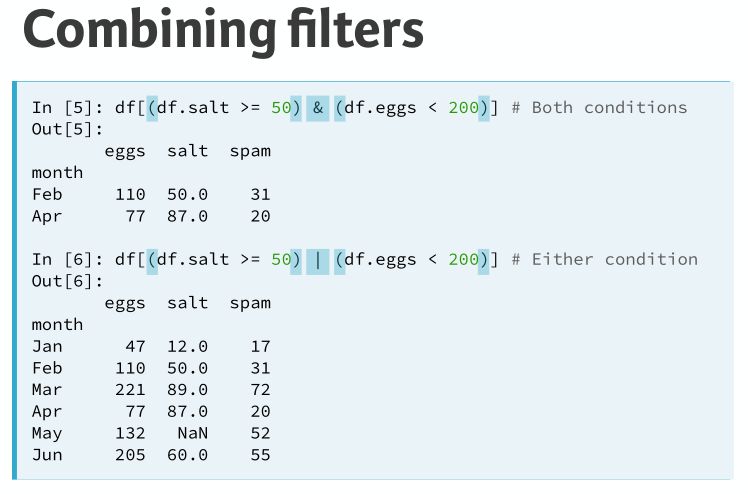
part8-4

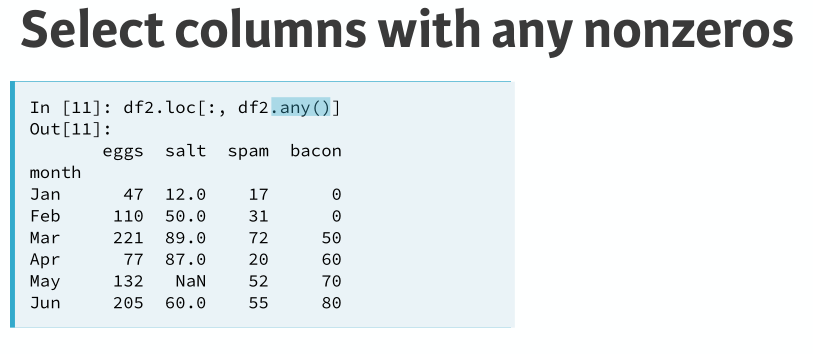


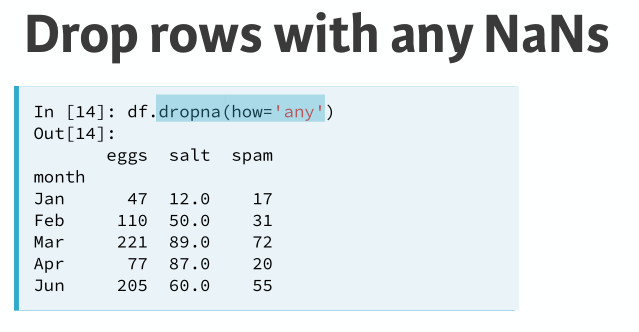
part9\_1

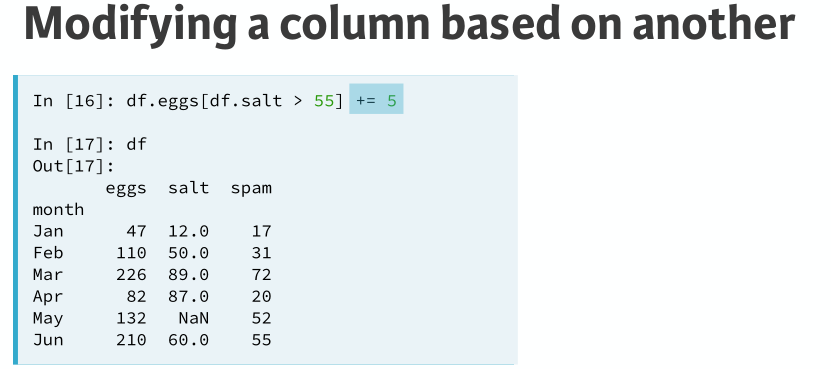


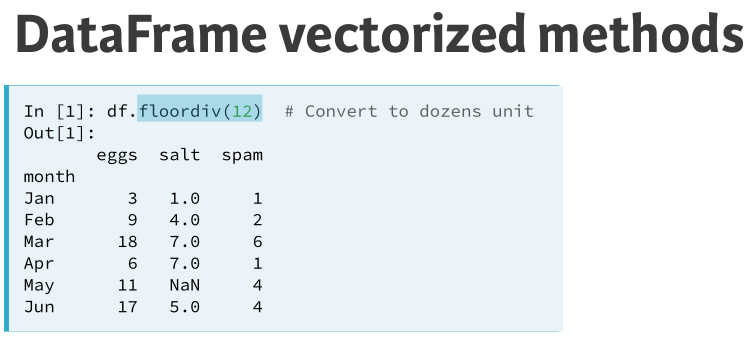


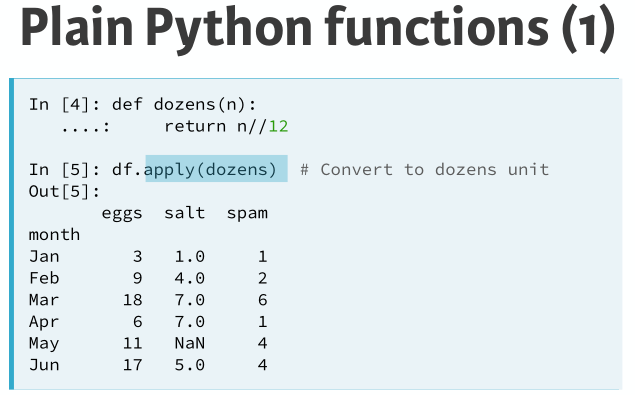




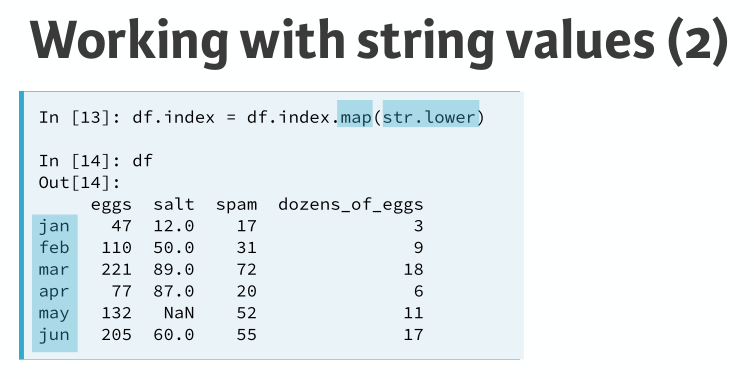




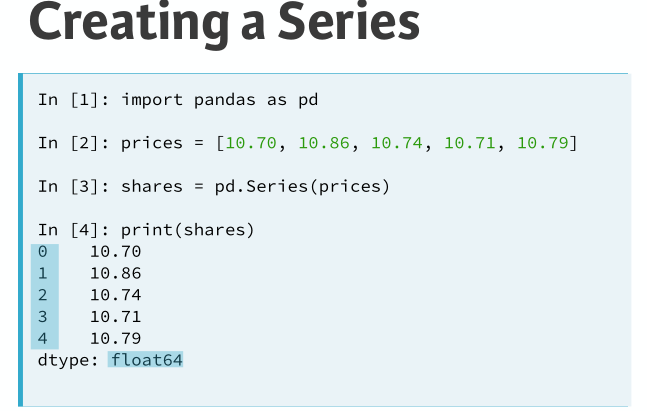


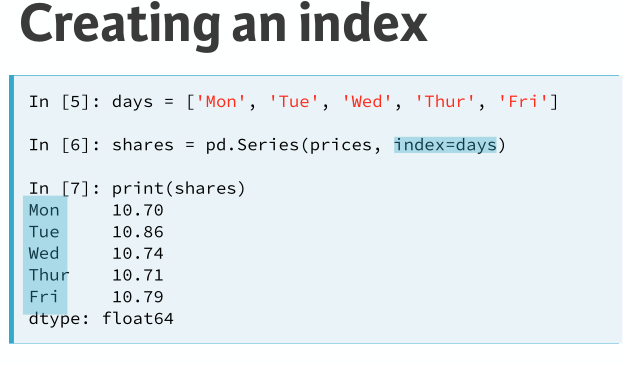


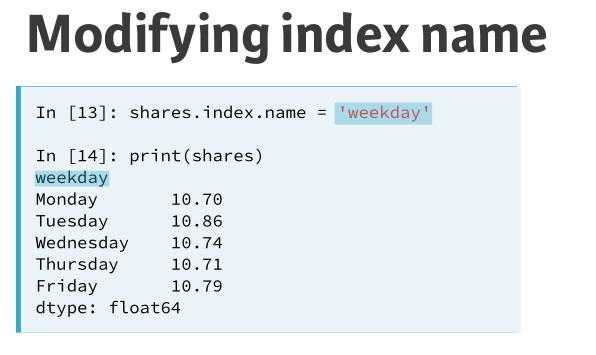


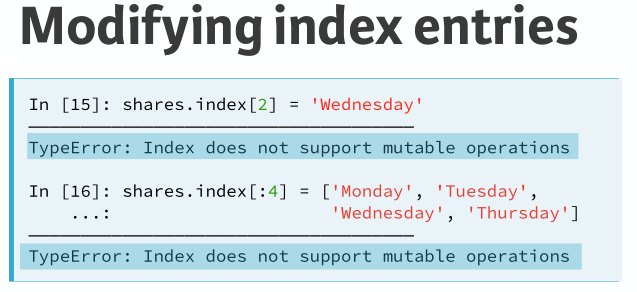


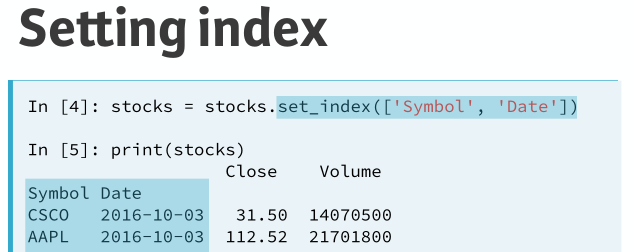
part9\_2

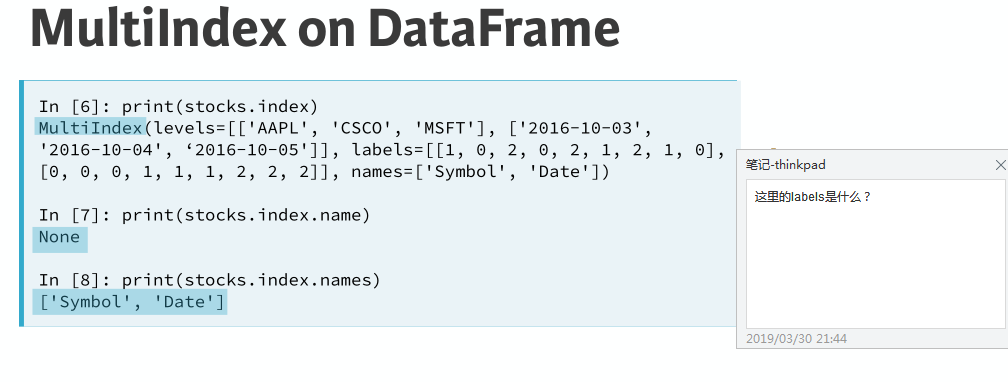


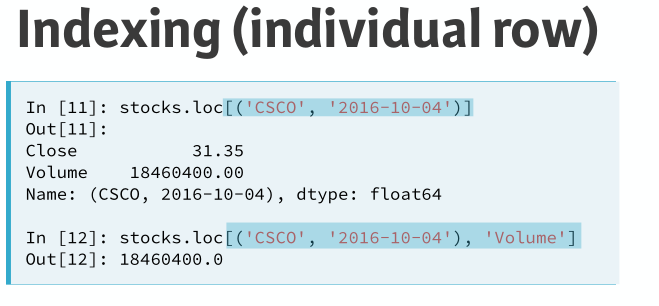


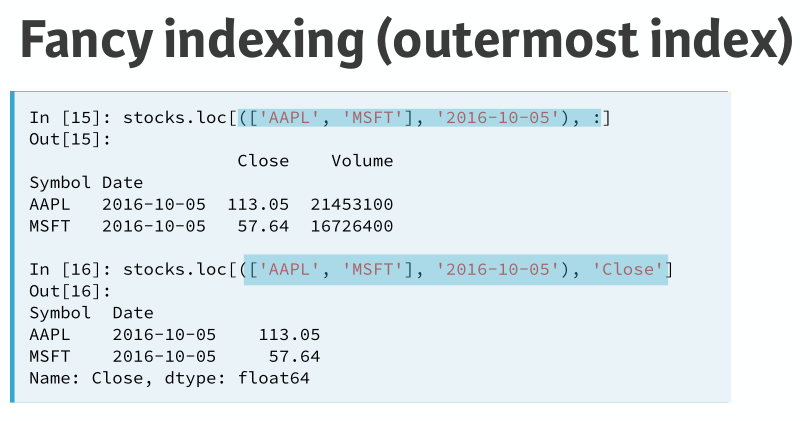


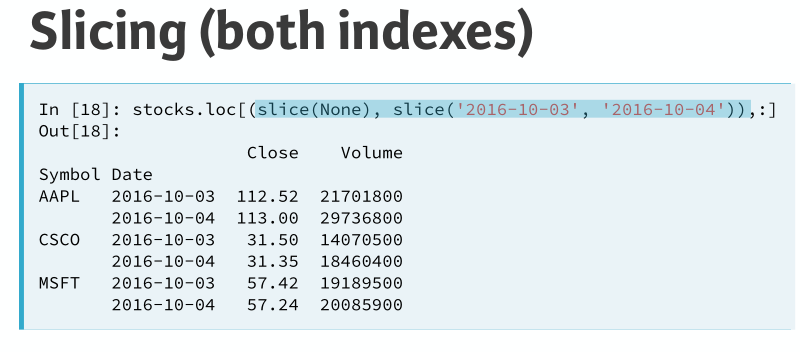




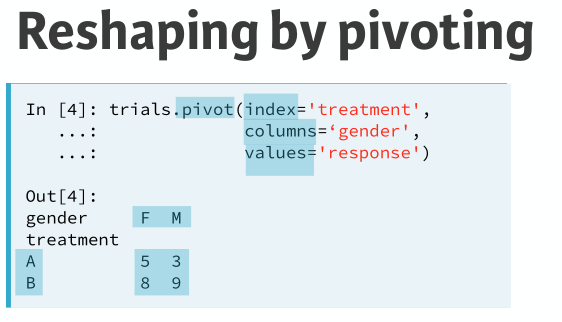


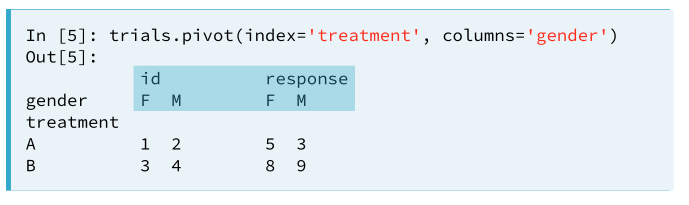


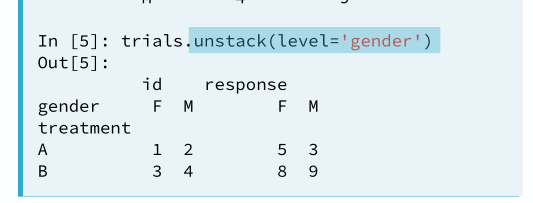


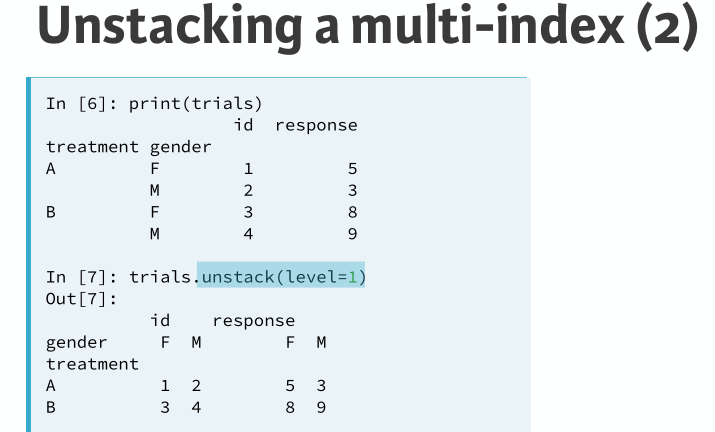


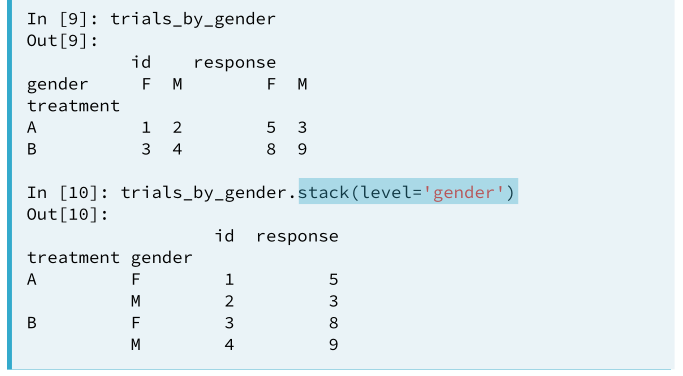
part9\_3

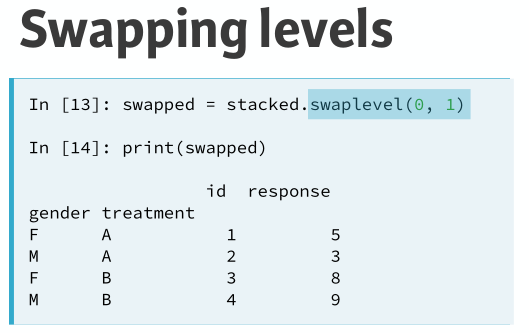


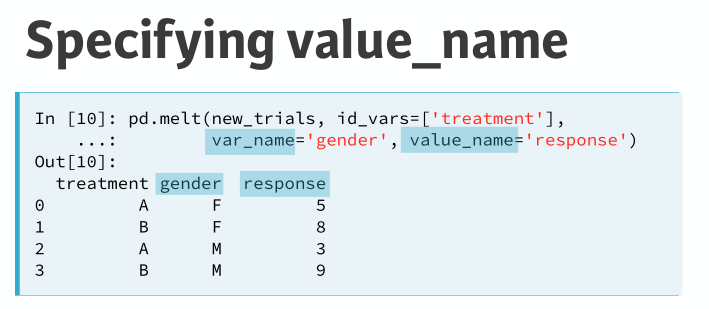


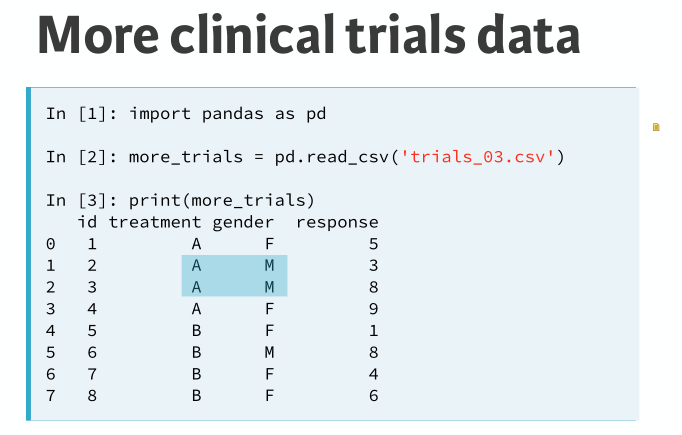






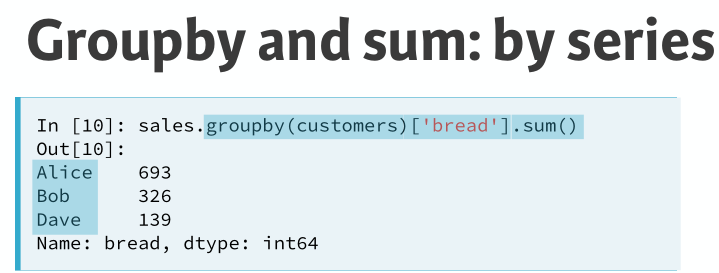


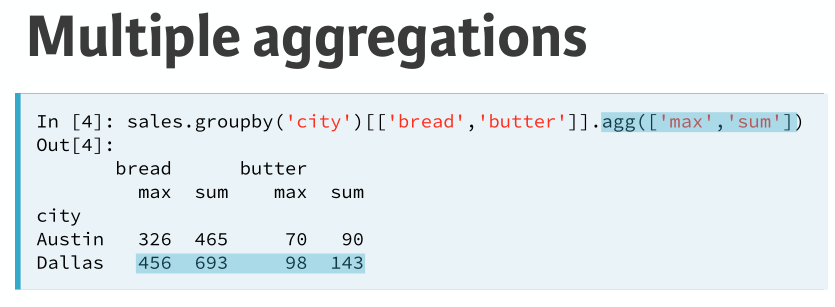




part9\_4

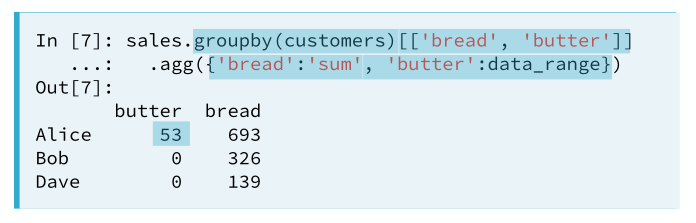


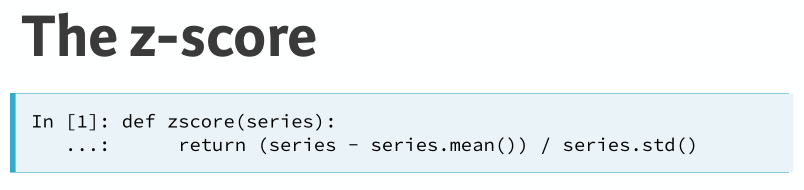


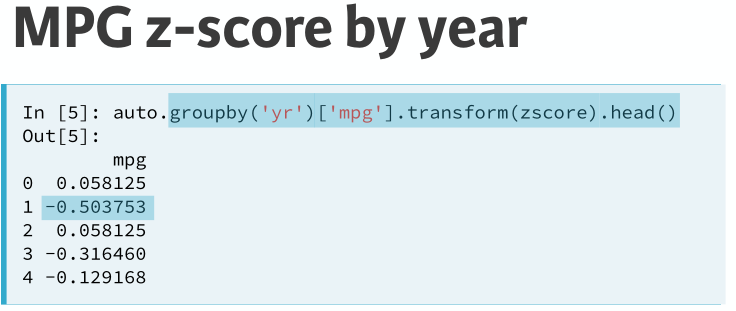


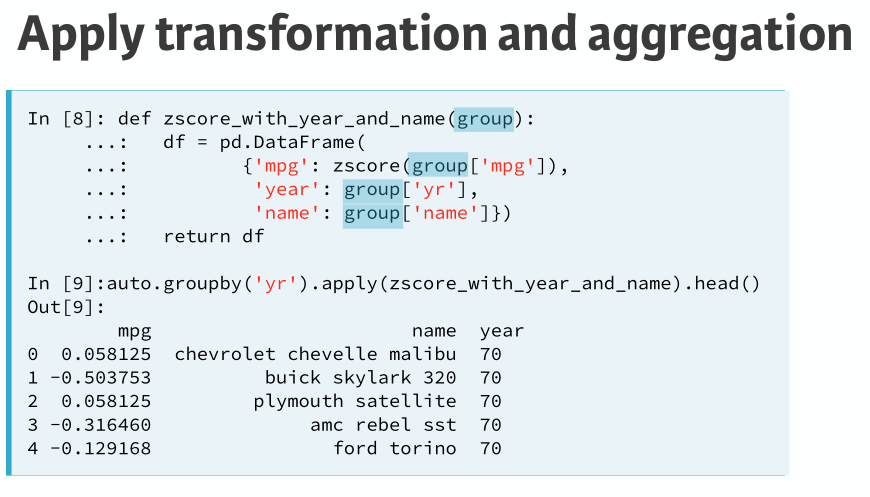


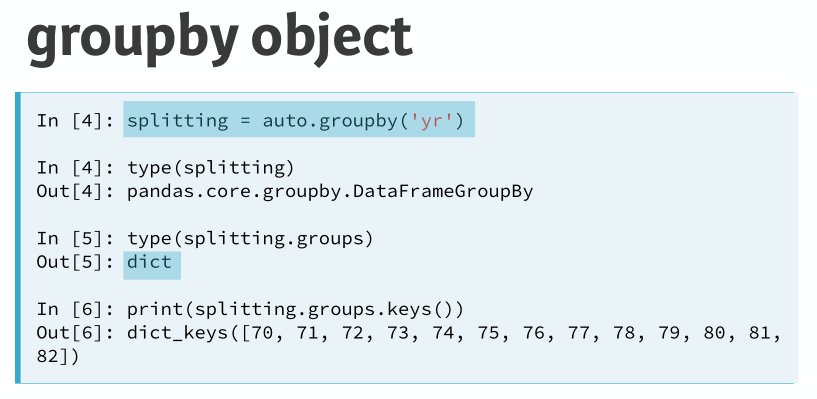


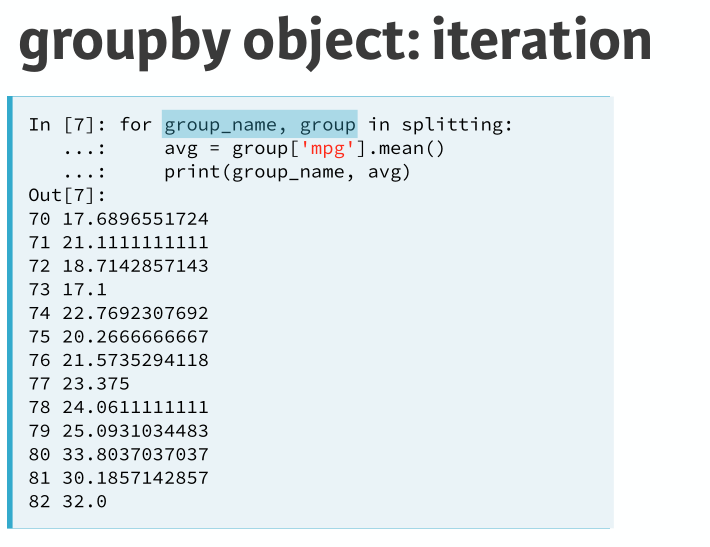


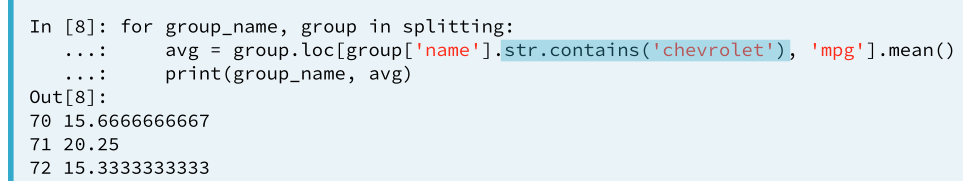


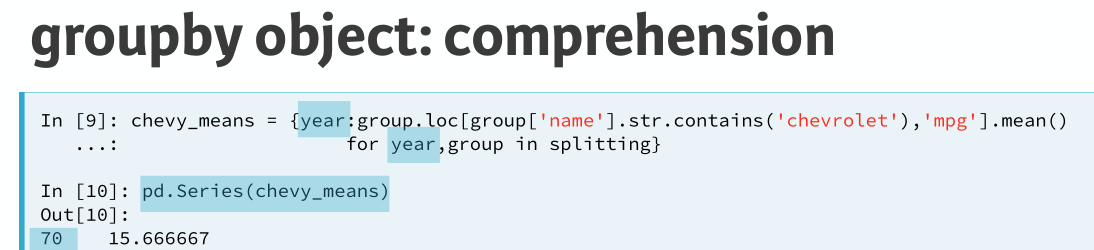


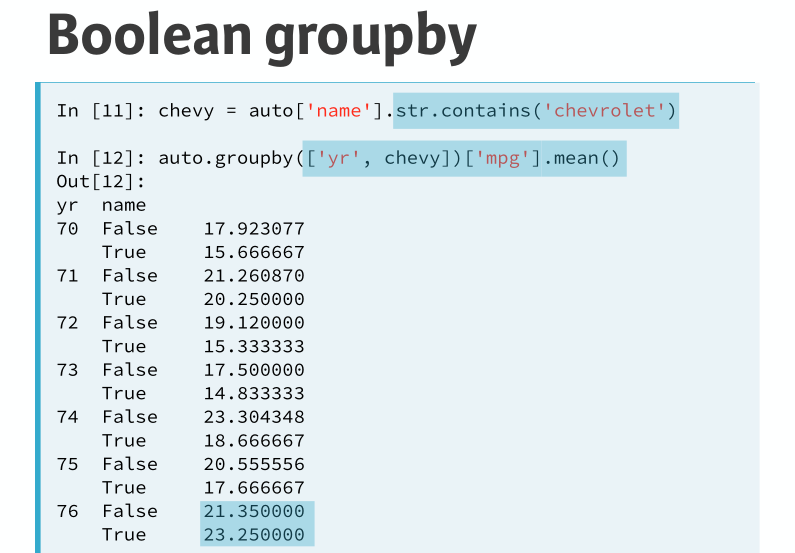












part10\_1

