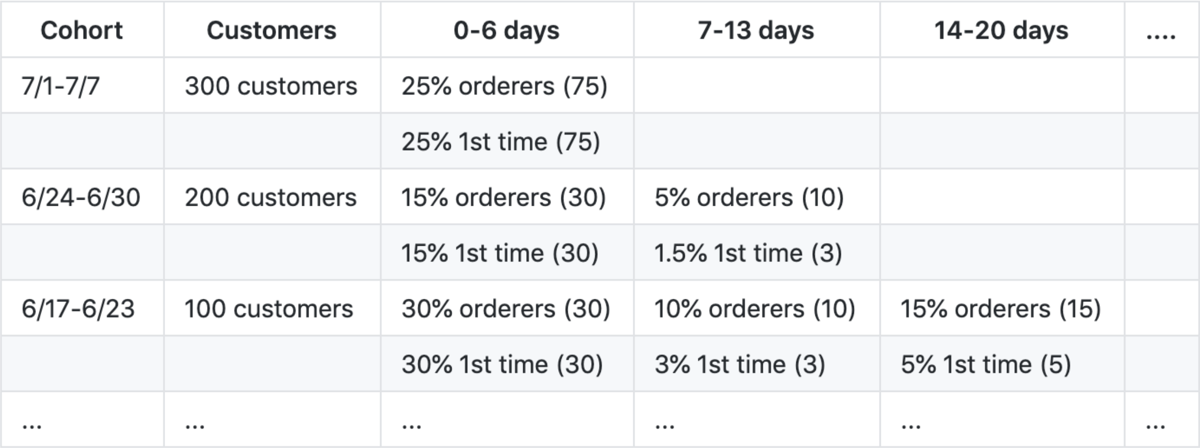
We would like to perform a cohort analysis on our customers to help identify changes in ordering behavior based on their signup date.

For this exercise group the customers into week long (7 days) cohorts and then calculate how many *distinct* customers ordered within X days from their signup date, where X is a multiple of 7. Older cohorts will have more buckets: 0-6 days, 7-13 days, 14-20 days, etc.

You may write your solution in the language of your choice, but we do have a preference towards Python (preferably using 3.6 or 3.7 with type hints). Please provide a README with clear instructions on how to setup any dependencies and execute your program or Jupyter notebook. The program should output an HTML table or CSV in a format similar to:



Cohort	Customers	0-6 days	7-13 days	14-20 days	....
7/1-7/7	300 customers	25% orderers (75)			
		25% 1st time (75)			
6/24-6/30	200 customers	15% orderers (30)	5% orderers (10)		
		15% 1st time (30)	1.5% 1st time (3)		
6/17-6/23	100 customers	30% orderers (30)	10% orderers (10)	15% orderers (15)	
		30% 1st time (30)	3% 1st time (3)	5% 1st time (5)	
...	...	...	...	...	...

Please also include one plot created from the data set output and give a small description of what the plot displays.

The program should read the data from both customers.csv and orders.csv and calculate at least 8 weeks of cohorts. All dates are stored in UTC but grouping should be handled in a configurable timezone (ex: PDT).

After a few submissions that were clearly simple adaptations of Greg Reda's blog post "Cohort Analysis with Python" we are asking submitters to not use guides or examples when doing this exercise and to solve the problem on their own. However, information about what a cohort analysis is and how it is used (ex: http://www.cohortanalysis.com/) is completely fine.