0.1 Question 3c

In the cell below, run the following line of code: `q3c_df = ice_cream_shops.sort_values('timestamp').groupby('bid').`

Is the granularity of `ice_cream_at_least_3` the same as the granularity of `q3c_df`? In other words, what does a single row of `q3c_df` represent, and what does a single row in ice_cream_at_least_3 represent? Explain the granularity of each `DataFrame`. Your answer does not need to be more than 2-3 lines, but you should be specific.

In [26]:
```
q3c_df = ice_cream_shops.sort_values('date').groupby('bid').agg('first')
q3c_df.head()
```

Out[26]:

bid	iid	date	score	type	Missing Score	name	address	lowercase_name
31	31_20180615	06/15/2018 12:00:00 AM	96	Routine - Unscheduled	False	Norman's Ice Cream and Freezes	2801 Leavenworth St	norman's ice cream and freezes
758	758_20190417	04/17/2019 12:00:00 AM	90	Routine - Unscheduled	False	BAKERY/ICE CREAM/STOREROOM	2 New Montgomery St	bakery/ice cream/storeroom
4671	4671_20170117	01/17/2017 12:00:00 AM	98	Routine - Unscheduled	False	MARCO POLO ITALIAN ICE CREAM	1447 TARAVAL St	marco polo italian ice cream
5032	5032_20170627	06/27/2017 12:00:00 AM	94	Routine - Unscheduled	False	MITCHELLS ICE CREAM	688 SAN JOSE Ave	mitchells ice cream
5524	5524_20190412	04/12/2019 12:00:00 AM	100	Routine - Unscheduled	False	AT&T Park - Coffee and Ice Cream (4A+4B)	24 WILLIE MAYS PLAZA	at&t park - coffee and ice cream (4a+4b)

In [27]:
```
q3c_df = ice_cream_shops.sort_values('timestamp').groupby('bid').agg('first')
q3c_df.head()
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
/tmp/ipykernel_490/92580563.py in ?()
```

```
----> 1 q3c_df = ice_cream_shops.sort_values('timestamp').groupby('bid').agg('first')
      2 q3c_df.head()

~/.local/lib/python3.10/site-packages/pandas/core/frame.py in ?(self, by, axis, ascendi g, inplace, ki
   7185             )
   7186         elif len(by):
   7187             # len(by) == 1
   7188
-> 7189             k = self._get_label_or_level_values(by[0], axis=axis)
   7190
   7191             # need to rewrap column in Series to apply key function
   7192             if key is not None:

~/.local/lib/python3.10/site-packages/pandas/core/generic.py in ?(self, key, axis)
   1907             values = self.xs(key, axis=other_axes[0])._values
   1908         elif self._is_level_reference(key, axis=axis):
   1909             values = self.axes[axis].get_level_values(key)._values
   1910         else:
-> 1911             raise KeyError(key)
   1912
   1913         # Check for duplicates
   1914         if values.ndim > 1:

KeyError: 'timestamp'
```

The granularity is not different.
q3c_df represent the first inspection information every ice cream shop received. It's a dataframe.
ice_cream_at_least_3 represents all the inspection information about qualified ice cream shops. It's a dataframe groupby object.

0.2 Question 3e

Finally, to examine different parts of a chained pandas statement, describe the purpose of each of the functions used (`.loc`, `.groupby`, `idxmax()`) in words.

Secondly, share what you think this line of code accomplishes. In other words, write a question that could be answered using this statement.

While the first part of this question will be graded for correctness, the second part of this question is a bit more open-ended. Answers demonstrating your understanding will get full credit.

In [ ]:
```
ice_cream_at_least_3.loc[ice_cream_at_least_3.groupby("bid")["score"].idxmax()].head()
```

.loc: apply a conditional selection on rows
.groupby: for each shop in ice_cream_at_least_3
.idxmax(): get the index for highest score
result: we retrieve the inspection information for each shop's highest score inspection

In [ ]:
```
# You may do some scratch work in this cell, however, only your written answer will be graded.
# Any outputs or dataframes you generate here will not be counted as part of your explanation.
```