3. (20 points) Counties

Every state in the United States is divided into counties that do not overlap with each other and together cover the whole state. A table `counties` contains one row for each county in the United States:

State	County	2010 Pop	2014 Pop
Alabama	Autauga County	54684	55395
Alabama	Baldwin County	183216	200111
Alabama	Barbour County	27336	26887

... (3139 rows omitted)

The table contains four columns:

- **State**: a string, the name of the state
- **County**: a string, the name of the county
- **2010 Pop**: an int, the population in 2010 (as estimated by the US Census Bureau)
- **2014 Pop**: an int, the population in 2014 (as estimated by the US Census Bureau)

In addition, we have the function `first` defined below:

```
def first(x):
    """x is an array"""
    return x.item(0)
```

In each part below, fill in the blanks of the Python expressions. **You must use ONLY the lines provided.** Some of the chained operations we might normally do in one line have been broken up into two or more lines, storing intermediate results in tables or arrays. Do not write any code outside the blanks provided. The expression in the last line should evaluate to the value described in the question.

(a) (2 pt) The name of the largest county in the United States (by 2014 population):

```
sorted = counties._____(_____, _____)

first(sorted._____(_____))
```

(b) (2 pt) The number of counties in which the population grew by more than 10,000 people between 2010 and 2014:

```
counties_with_change = counties.with_column('Pop Change',

    counties._____(_____) - counties._____(_____) )

counties_with_change._____(_____, _____).num_rows
```

(c) (2 pt) A new table called `states` that has one row for each state. It should have three columns: the state's name, the total population of the state in 2010, and the total population of the state in 2014. It should not have any column corresponding to county names.

```
counties_3column = counties._____(_____)

states = counties_3column._____(_____, _____)

states
```

(d) (4 pt) A new table called `biggest_county` that has one row for each state. Its columns should contain the state's name, the name of the largest county in the state (by 2014 population), the 2010 population of that county, and the 2014 population of that county. It doesn't matter what the column names are.

```
sorted = counties._____(_____, _____)

biggest_county = sorted._____(_____, _____)

biggest_county
```

(e) (4 pt) The table `biggest_county` with an additional column called `'Pct in Largest County'`. The column should contain what percent of that state's population lived in the largest county in 2014. For example, if a state has only three counties and the populations in 2014 are 250,000, 200,000, and 50,000, then `'Pct in Largest County'` should be 50.

```
with_states = biggest_county._____(_____, _____)

biggest_county_with_pct = biggest_county.with_column('Pct in Largest County',

    100 * _____.column('2014 Pop first') /_____.column('2014 Pop sum'))

biggest_county_with_pct
```

(f) (6 pt) The table `states` with an additional column called `'Pop in Large Counties'`. The column should contain the total number of people in each state that lived in counties with population more than 100,000 in 2014. For example, if a state has only three counties and the populations in 2014 are 250,000, 200,000, and 50,000, then `'Pop in Large Counties'` should be 450,000. You may assume every state has at least one such county.

```
all_large = counties._____(_____, _____)

all_large = all_large._____(_____, _____)

all_large = all_large.select('State','2014 Pop sum').relabeled(1, 'Pop in Large Counties')

states_with_large_pop = states._____(_____, all_large)

states_with_large_pop
```