In writing code you do not need to worry about specifying the proper `import` statements. Don’t worry about getting function or method names exactly right. Assume that all libraries and packages we’ve discussed are imported in any code you write. You can write any helper functions you would like in solving the problems. You should show your work on any analysis questions.

**PROBLEM 1 : (T/F)**

A study followed 369 people with cardiovascular disease, randomly selected from hospital patients. A year later, those who owned a dog were four times more likely to be alive than those who didn’t. Briefly justify your answer.

1.1. This study is a randomized controlled experiment.

   [True] [False]

1.2. This study shows that dog owners live longer than cat owners on average.

   [True] [False]

1.3. This study shows that for someone with cardiovascular disease, adopting a dog will probably cause them to live longer.

   [True] [False]
PROBLEM 2 :  (Expressions (10 points)
An array of integers named ca contains the (estimated) population of California every 10 years.
The array has 11 items. The first item is the population of California in 1900. The last is the population in
2000.

array ([1485053, 2377549, 3426861, ..., 23667902, 29760021, 33871648])

Write a Python expression below for each of the following descriptions that computes its value. The first
one is provided as an example. Do not include numbers above (e.g., 1485053) in your solutions.

• The population in 1900.

   ca.item(0)

2.1. The population change from 1940 to 1960, expressed as a number of persons (not a proportion).

2.2. Whether the population ever grew by less than 500000 in a decade represented by ca. (True or False)

2.3. The annual (yearly) growth rate from 1920 to 1930.

2.4. The population in 1924, assuming a fixed exponential annual growth rate from 1920 to 1930. You
    may use the name g for the annual growth rate from 1920 to 1930 (computed above).
A dodecahedron is a twelve-sided solid as shown.

The volume of a dodecahedron is given by this formula in terms of the length of a side $s$:

$$\frac{15 + 7\sqrt{5}}{4} \times s^3$$

The surface area is given by this formula again in terms of the length of a side $s$.

$$s^2 \times 3 \times \sqrt{25 + 10 \times \sqrt{5}}$$

You may use the `math.sqrt` function to calculate square roots, e.g., `math.sqrt(25)` evaluates to 5. Complete both functions below to return the volume and surface areas of a dodecahedron.

```python
def dodecavolume(s):
    """
    return volume of dodecahedron with side s
    """

    def dodecasurface(s):
        """
        return surface area of dodecahedron with side s
        """
```
PROBLEM 4:  (Tables (8 points))

Each row of the trip table from lecture describes a single bicycle rental in the San Francisco area. Durations are integers representing times in seconds. The first three rows out of 338,343 appear below.

<table>
<thead>
<tr>
<th>Start</th>
<th>End</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ferry Building</td>
<td>SF Caltrain</td>
<td>765</td>
</tr>
<tr>
<td>San Antonio Shopping Center</td>
<td>Mountain View City Hall</td>
<td>1036</td>
</tr>
<tr>
<td>Post at Kearny</td>
<td>2nd at South Park</td>
<td>307</td>
</tr>
</tbody>
</table>

Write a Python expression below each of the following descriptions that computes its value. You may use up to two lines and introduce variables.

4.1.  The average duration of a rental that lasted more than 2 minutes.

4.2.  The number of rentals that started at the SF Caltrain station.

4.3.  The name of the station where the most rentals ended (assume no ties)
4.4. We have access to the following menu table which contains each food item, the number of calories, and the category for the item:

<table>
<thead>
<tr>
<th>Item</th>
<th>Cal</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>Egg McMuffin</td>
<td>300</td>
<td>Breakfast</td>
</tr>
<tr>
<td>Baked Apple Pie</td>
<td>250</td>
<td>Desserts</td>
</tr>
<tr>
<td>McRib</td>
<td>500</td>
<td>Beef &amp; Pork</td>
</tr>
<tr>
<td>Daily Double</td>
<td>430</td>
<td>Beef &amp; Pork</td>
</tr>
</tbody>
</table>

... (257 rows omitted)

Which of the following Python expressions evaluates to the number of desserts with less than 200 calories? Select all that apply.

A

(menu.where(Category, are.equal_to(Desserts))
 .where(Calories, are.below(200))
 .num_rows)

B

(menu.where(Calories, are.below(200))
 .where(Category, are.equal_to(Desserts))
 .num_rows)

C

(menu
 .with_column(Below200, menu.column(Calories) < 200)
 .pivot(Category, Below200)
 .num_rows)

D

(menu
 .with_column(Below200, menu.column(Calories) < 200)
 .pivot(Category, Below200)
 .where(Below200, are.equal_to(True))
 .num_rows)

E

(menu.group(Category)
 .where(Calories, are.below(200))
 .where(Category, are.equal_to(Desserts))
 .column(count).item(0))

F

(menu.where(Calories, are.below(200))
 .group(Category)
 .where(Category, are.equal_to(Desserts))
 .column(count).item(0))
4.5. Assume we have a tables for the 2016-2017 NBA Season. Assume that if a column contains numbers, then they are integers, and otherwise, it is a column of strings. The nba table contains 8 columns. The first few rows are shown below

Fill in the blanks of the Python expressions to compute the described values. You must use only the lines provided.

The last line of each answer should evaluate to the value described. Assume that the statements `from datascience import *` and `import numpy as np` have been executed. You may add anything you would like to the blanks below, but you may not add code outside of the blanks.

A. The age of the oldest NBA player.

```python
______________________(nba._____________________________________(age))
```

B. The three-letter prefix of the team which has the highest paid player with the position center (C) in the NBA. You may assume there is only one such player.

```python
centers = nba.__________(position, ___________)

centers.__________(______________________________).column(______________).item(0)
```
C. The number of teams that have fewer than 5 players older than 30.

\[ \text{old} = \text{nba.}\_\text{____________}\(_\text{_______________}, \text{are.}\_\text{________________}_) \]

\[ \text{num_old} = \text{old.}\_\text{____________}\(_\text{________________}_) \]

\[ \text{old.}\_\text{____________}\(_\text{_______________}, \text{________}).\_\text{____________}_) \]

D. The number of positions for which the total points scored by CLE players in that position was higher than the total points scored by BOS players in that position.

\[ \text{positions} = \text{nba.pivot(prefix,\_\text{___________________________})} \]

\[ \text{sum(positions}\_\text{___________________________}) \]
PROBLEM 5: (Distributions)  
The two histograms of bike trip durations below were both generated by `trip.hist(...)` using different bins.

Write the proportion of trips that fall into each range of durations below. *Show your work.* If it is not possible to tell from the histograms, instead write *Not enough information.*

5.1. Between 200 (inclusive) and 400 (exclusive) seconds

5.2. Between 300 (inclusive) and 900 (exclusive) seconds

5.3. Between 400 (inclusive) and 900 (exclusive) seconds

5.4. Between 200 (inclusive) and 300 (exclusive) seconds