The squirrels of UC Berkeley have decided to have a faceoff with the squirrels at Stanford. Its up to you to make sure that our squirrels are prepared with the most important knowledge before they head out!

1 A Test of Skill

Before they allow you to help them, the squirrels want to make sure that you're up for the task. Answer their questions to pass their inspection!

1.1 Ints, Floats, and Strings

Impress the squirrels with your knowledge of data types! Print the outputs (or errors) of the following lines of Python code!

```
>>> 6 / 3
```

... 2.0

```
>>> round(2 / 3, 2)
```

... .66

```
>>> "42".replace("4", "ans").replace("2", "wer").upper()
```

... "ANSWER"

```
>>> "Doggo" + "Squirrel"
```

... "DoggoSquirrel"

```
>>> 1337 + "Squirrel"
```

ERROR

1.2 Diagram review

The squirrels want you to diagram the following expression. Show them what you got!

```
>>> add(15, sub(mul(round(5), 3), 2))
```

add(15, sub(mul(round(5), 3), 2))
=28

add   15   sub(mul(round(5), 3), 2)
=13

sub   mul(round(5), 3)   2
=15

mul   round(5)   3
=5

5

`... 28`

1.3   To be, or None to be

As a final question, the squirrels want you to prove you know the difference between prints and outputs. Show them your skills!

```
>>> x = "Wassup?"
>>> print(x)
```

... Wassup?

```
>>> x
```

`... "Wassup?"`

2   Organize the data!

2.1   Make some arrays!

The squirrels have a bunch of data about some neutral squirrels in the Bay Area. Firstly, they want you to put all the data about distance to UC Berkeley in an array assigned to **dst_to_berk**. Then, they want to you to put all the data about cleverness score in an array assigned to **cleverness**.

dst_to_berk	cleverness
5	1
3	5
2	7

Write down the code to conduct this request!

```
>>> dst_to_berk = make_array(5, 3, 2)
>>> cleverness = make_array(1, 5, 7)
```

2.2 Arraying the Forces

Now the Squirrels want you to put data about the various populations into a collective array. Put the population of Berkeley squirrels, Stanford squirrels, and neutral squirrels into an array, in that order. Assign the array to the variable **forces**.

Locations	Pop
Berkeley	20
Stanford	10
Neutral	5

Write down the code to make such an array!

```
>>> forces = make_array(20, 10, 5)
```

2.3 Indexing into the Array

Now the squirrels want to know the smallest absolute difference in population between the three groups. Index into the forces array that you just made to find the smallest difference between the squirrel populations of Berkeley, Stanford, and the neutrals.

Remember, we can index into an array by using the method .item:

```
>>> forces.item(0)
... 20
```

Fill in the missing code:

```
>>> min(_____, _____, _____)
```

```
>>> min(abs(forces.item(0) - forces.item(1)), abs(forces.item(0) - forces.item(2)), abs(forces.item(1) - forces.
    item(2)))
```

3   Visualizing the Forces

3.1   Population Bar Chart!

The Squirrels have a difficult task for you: create a bar chart from the population data. Luckily, they've supplied a function named **bar**(labelArray, dataArray) to do the graphing. The bar function takes in two inputs: a labelArray, which contains the names of the bars in the chart, and a dataArray, which contains the value of each label to be charted. You will have to create the label array (which should consist of the values Berkeley, Stanford, and Neutral). Use the forces array that you've already created to supply the data. Remember that the values should match up - we want the values in the data array to correspond with the correct locations.

The final graph is displayed below.

Write your code here!

```
>>> locations = make_array("Berkeley", "Stanford", "Neutral")
>>> bar(locations, forces)
```
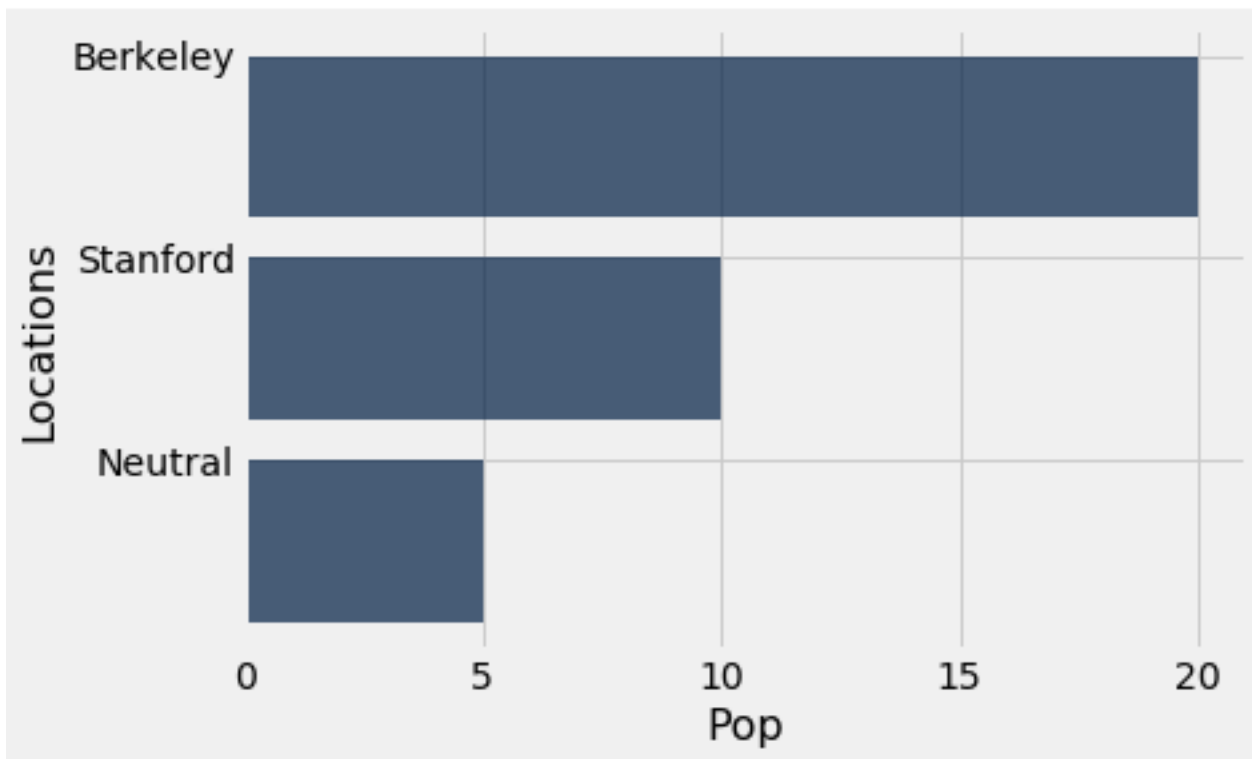
3.2   Scatter the Cleverness Data

Now the squirrels want you to create a scatter plot of the cleverness data. Luckily, they've created a function called **scatter**(xArray, yArray), that will create the scatter plot for you. Use the arrays that you've created from before to create the scatter plot, using the distance to berkeley data as the x, and the cleverness score data as the y.

The final graph is displayed below.

Write your code here!

```
>>> scatter(x, y)
```
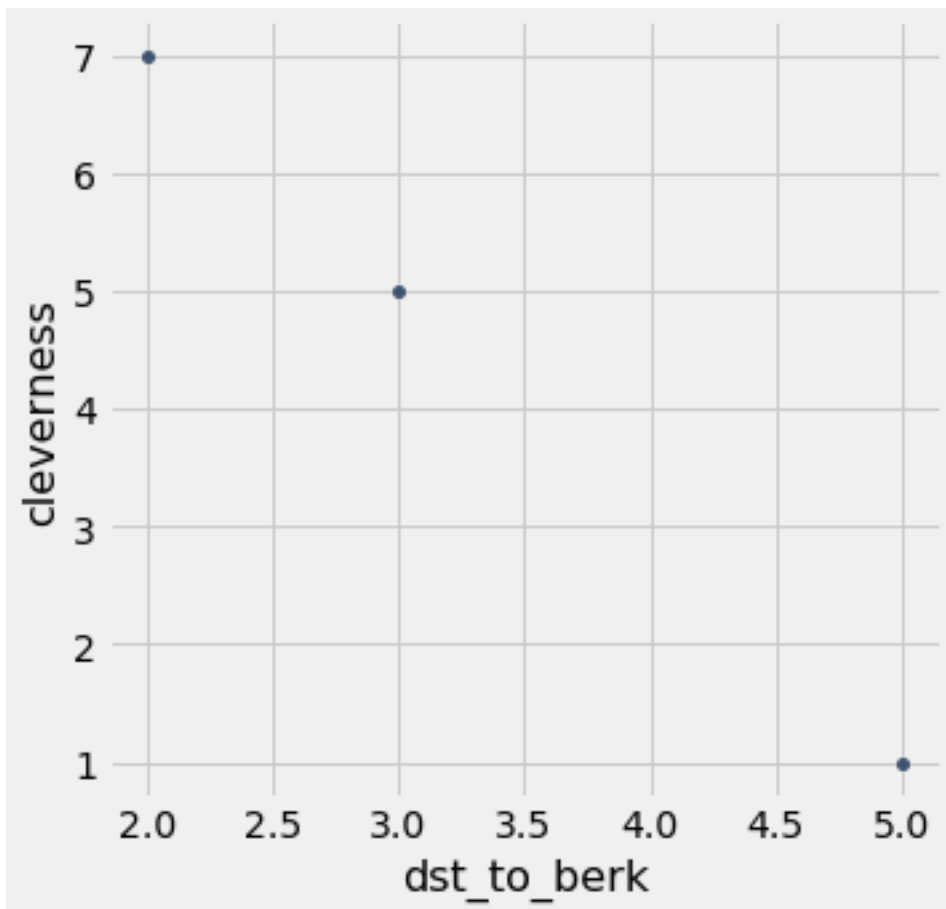
4 Data Manipulation

4.1 Calculations: Calculations Calculations

The squirrels now want you to calculate a couple of values for them. Fill in the code you would use beneath each value. Remember to use array methods!

The product of all the values in the **dst_to_berk** array

```
>>> dstProd = _____
```

```
>>> dstProd = np.prod(dst\_to\_berk)
```

The product of all cleverness score values:

```
>>> cleverProd = _____
```

```
>>> cleverProd = np.prod(cleverness)
```

The length of both arrays:

```
>>> dstLen = _____
>>> cleverLen = _____
```

```
>>> dstLen = len(dst\_to\_berk)
>>> cleverLen = len(cleverness)
```

Now the squirrels want you to invert the lengths (Hint: To invert a number, use

$$\frac{1}{number}$$

```
>>> dstLenInverted = _____
>>> cleverLenInverted = _____
```

```
>>> dstLenInverted = 1 / dstLen
>>> cleverLenInverted = 1 / cleverLen
```

The squirrels want you to calculate the value of the products to the power of the lengths inverted:

```
>>> finalValueDist = _____
>>> finalValueClever = _____
```

```
>>> finalValueDist = dstProd ** dstLenInverted
>>> finalValueClever = cleverProd ** cleverLenInverted
```

Note: Those tricky squirrels got you to calculate the geometric mean, which is another measure of "the center" of some data.