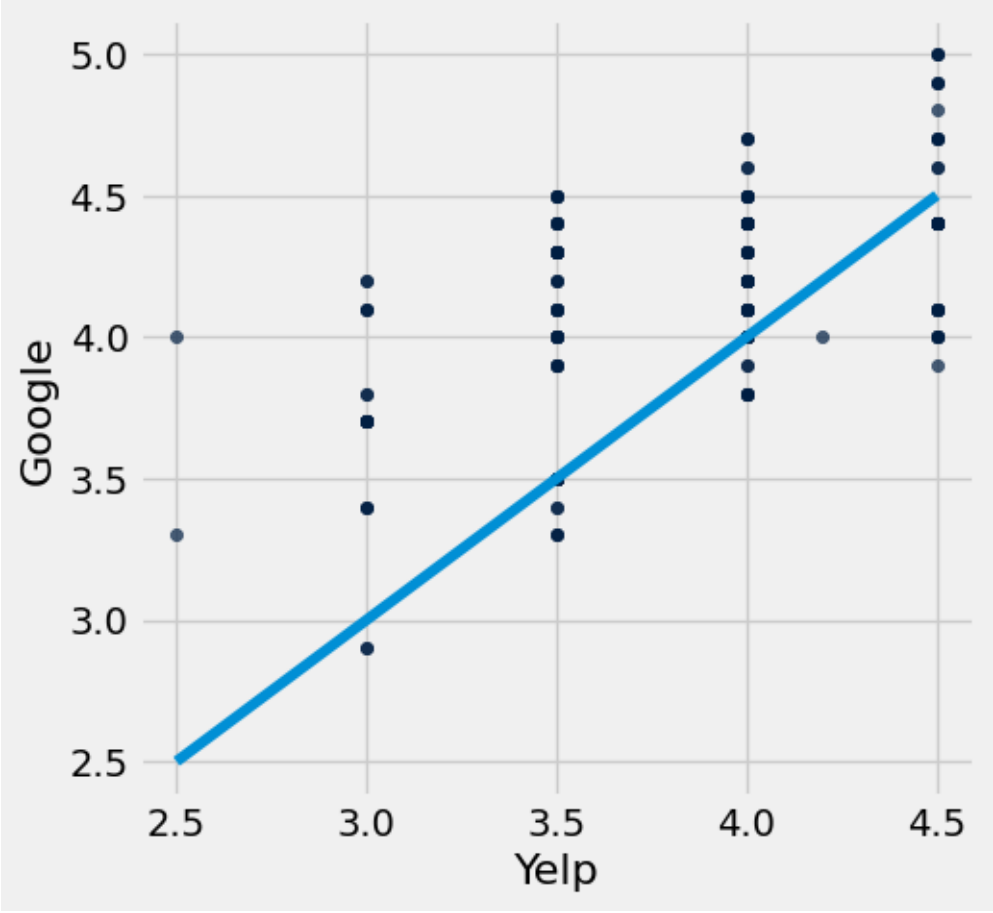

Question 2. Let's look at how the Yelp scores compare to the Google scores in the `burritos` table. First, assign `yelp_google_tbl` to a table only containing the columns `Yelp` and `Google`. Then, make a scatter plot with Yelp scores on the x-axis and the Google scores on the y-axis. **(8 Points)**

```
In [7]: yelp_google_tbl = burritos.select('Yelp', 'Google')
        yelp_google_tbl.scatter("Yelp")

        # Don't change/edit/remove the following line.
        # To help you make conclusions, we have plotted a straight line on the graph (y=x).
        plt.plot(np.arange(2.5,5,.5), np.arange(2.5,5,.5));
```



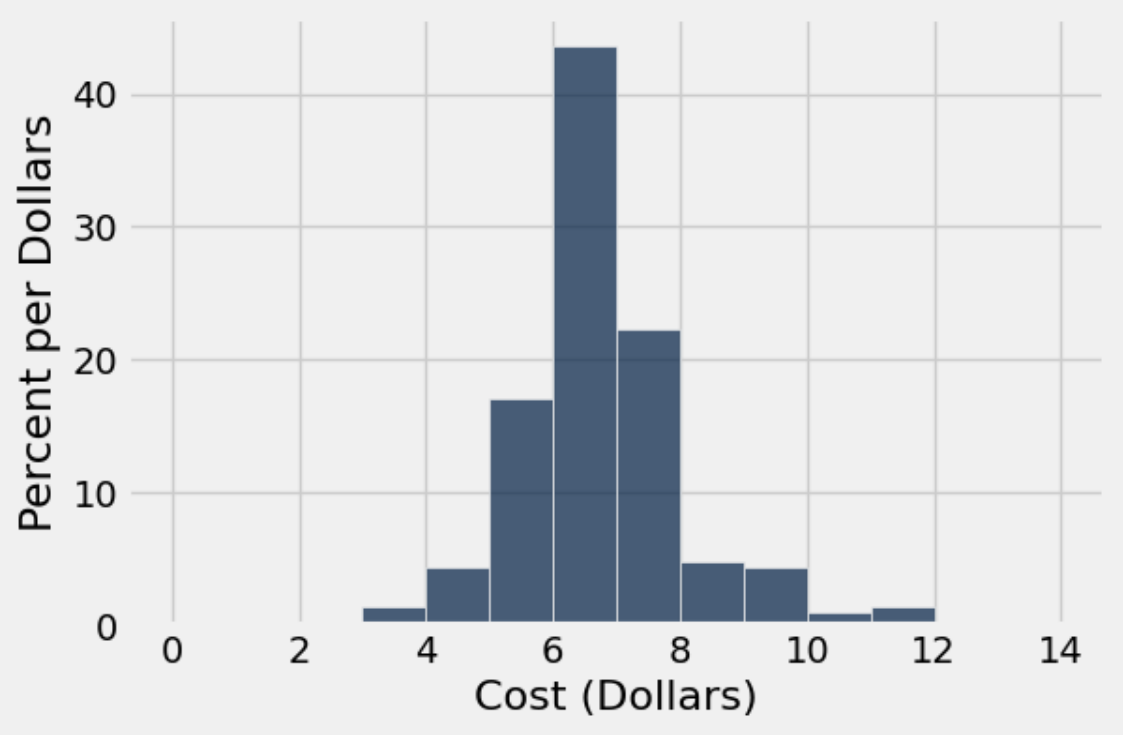
Question 3. Looking at the scatter plot you just made in Question 1.2, do you notice any pattern(s) or relationships between Yelp and Google ratings (i.e. is one of the two types of scores consistently higher than the other one)? If so, describe them **briefly** in the cell below. **(8 Points)**

Google's scores are consistently higher than Yelp's

Question 6. Edwin thinks that burritos in San Diego are cheaper (and taste better) than the burritos in Berkeley. Plot a histogram that visualizes that distribution of the costs of the burritos from San Diego in the `burritos` table. Also use the provided `cost_bins` variable when making your histogram, so that the histogram is more visually informative. **(8 Points)**

```
In [35]: cost_bins = np.arange(0, 15, 1) # Do not change this line
         # Please also use the provided bins

         burritos.hist("Cost", bins=cost_bins)
```



Question 2. At the moment, the `Job` column of the `sf` table is not sorted (no particular order). Would the arrays you generated in the `Jobs` column of the previous question be the same if we had sorted alphabetically instead before generating them? Explain your answer. To receive full credit, your answer should reference *how* the `.group` method works, and how sorting the `Jobs` column would affect this. **(8 Points)**

Note: Two arrays are the **same** if they contain the same number of elements and the elements located at corresponding indexes in the two arrays are identical. An example of arrays that are NOT the same: `array([1,2]) != array([2,1])`.

No, since .group will take into account all different values of the column that we group upon, then pass through a function that we define, if inside that function, we sort the values, we will receive an array that is sorted ascendingly or descendingly, and different from original array withour sorting

---

Question 4. Why might some of the row values be `0` in the `department_ranges` table from the previous question. **(8 Points)**

Because some departments are not exist in that organization group