Question 1.1. Let's run a hypothesis test using confidence intervals to see if there is a linear relationship between egg weight and bird weight. Define the null and alternative hypotheses that will allow you to conduct this test. **(8 points)**

Note: Please write your answer **in the cell below** in the following format: - **Null Hypothesis:** - **Alternative Hypothesis:**

Type your answer here, replacing this text.

Question 1.4. Create an array called `resampled_slopes` that contains the slope of the best fit line for 1000 bootstrap resamples of `birds`. Plot the distribution of these slopes. **(8 points)**

```
In [28]: resampled_slopes = make_array()

         for i in np.arange(1000):
             birds_bootstrap = birds.sample()
             bootstrap_line = fit_line(birds_bootstrap, "Egg Weight", "Bird Weight")
             bootstrap_slope = bootstrap_line.item(0)
             resampled_slopes = np.append(resampled_slopes, bootstrap_slope)

         # DO NOT CHANGE THIS LINE
         Table().with_column("Slope estimate", resampled_slopes).hist()
```
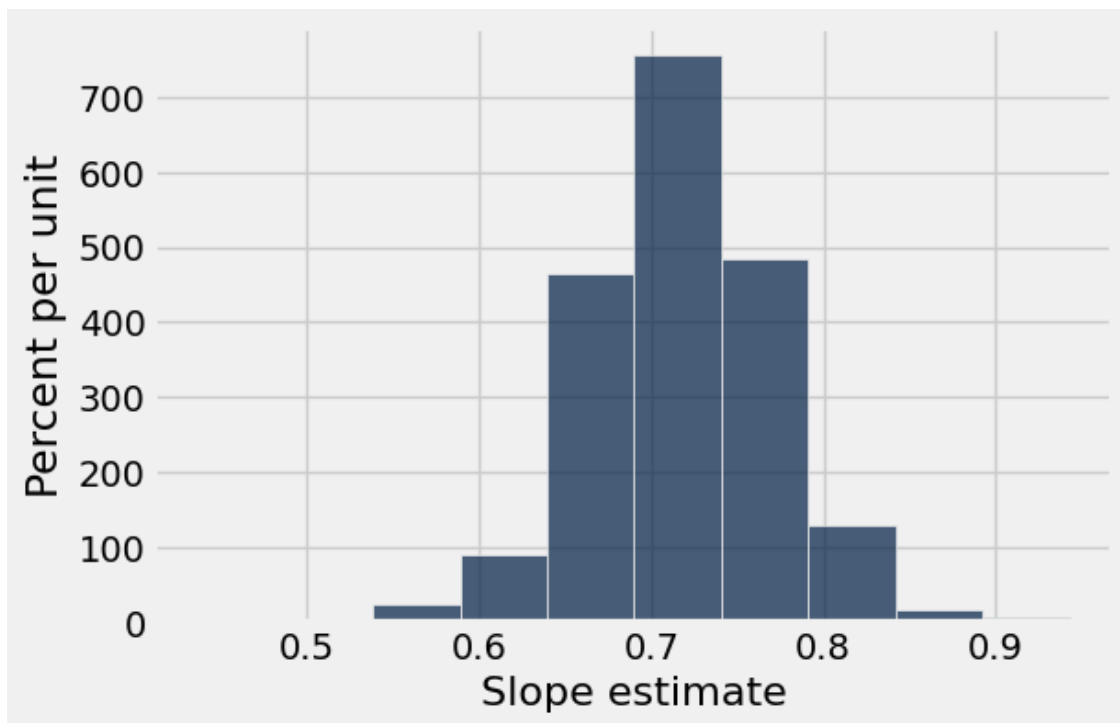


```
In [29]: grader.check("q1_4")
```

```
Out[29]: q1_4 results: All test cases passed!
```

Question 1.6. Based on your confidence interval, would you accept or reject the null hypothesis that the true slope is 0? Why? What p-value cutoff are you using? **(8 points)**

Reject 0 khong nam trong khoang 95% slope (tu 0.6 den 0.8)

Question 1.7. What do you think the true slope is? You do not need an exact number. How confident are you of this estimate? **(8 points)**

Hint: Can you provide an interval that you think the true slope falls in?

Type your answer here, replacing this text.

Question 2.4. Create an array called `predictions_for_eight` that contains the predicted bird weights based on an egg of weight 8 grams for each regression line in `regression_lines`. **(8 points)**

```
In [51]: slope1 = regression_lines.column('Slope')
         intercept1 = regression_lines.column('Intercept')
         predictions_for_eight = slope1 * 8 + intercept1

         # This will make a histogram of your predictions:
         table_of_predictions = Table().with_column('Predictions at Egg Weight=8', predictions_for_eigh
         table_of_predictions.hist('Predictions at Egg Weight=8', bins=20)
```
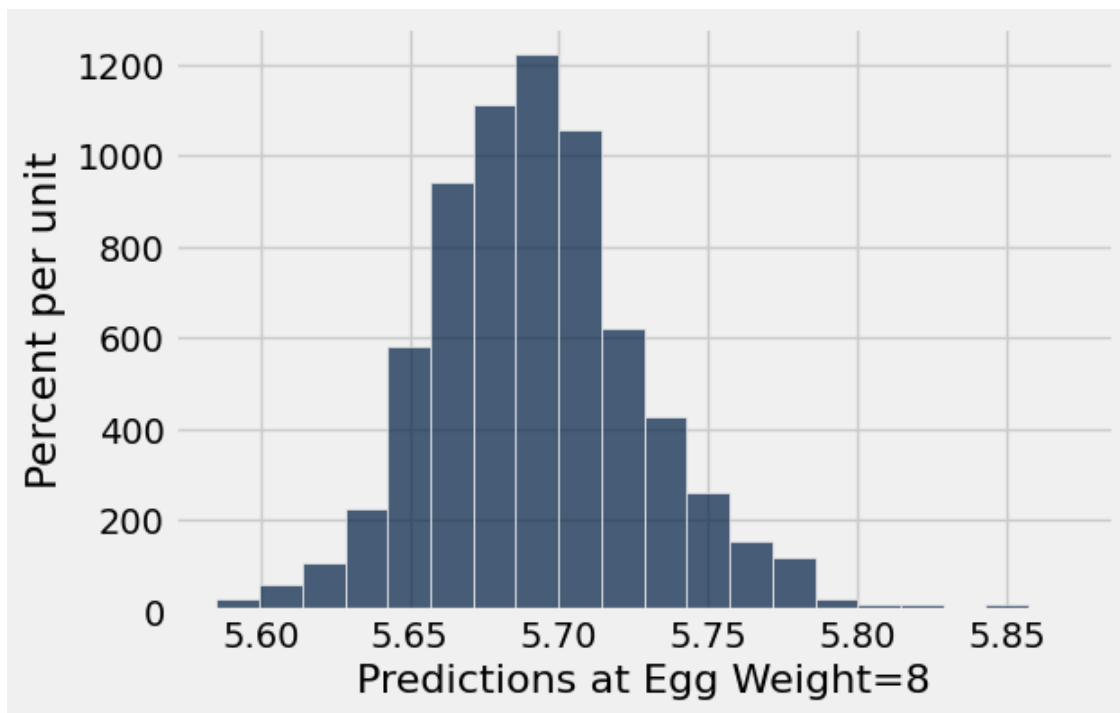


```
In [52]: grader.check("q2_4")
```

```
Out[52]: q2_4 results: All test cases passed!
```