Question 1.2.

a). Complete the `percentages_in_resamples` function such that it simulates and returns a numpy array of 2022 elements, where each element represents a bootstrapped estimate of the percentage of voters who will vote for Khow Thai. You should use the `one_resampled_percentage` function you wrote above.

b). Then run your function `percentages_in_resamples` and store the results in a numpy array called `resampled_percentages`. Then create a density histogram of the entries in `resampled_percentages` array. Label your axes and include a title on your plot.

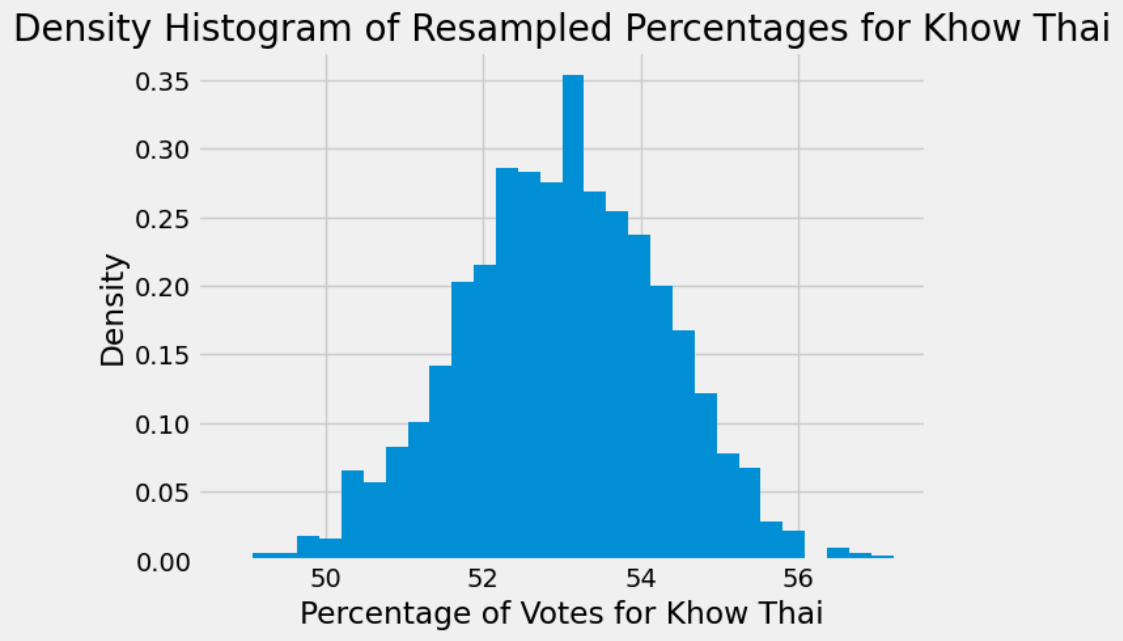
```
In [6]: def percentages_in_resamples():
            resampled_percentages = np.zeros(2022)
            for i in range(2022):
                resampled_percentages[i] = one_resampled_percentage(votes)

            return resampled_percentages
```

```
In [7]: resampled_percentages = percentages_in_resamples()
        plt.hist(resampled_percentages, bins=30, density=True)

        plt.xlabel('Percentage of Votes for Khow Thai')
        plt.ylabel('Density')
        plt.title('Density Histogram of Resampled Percentages for Khow Thai')

        plt.show()
        # your code for histogram above this line
```



In [8]:
```
grader.check("q1_2")
```

Out[8]:
```
q1_2 results: All test cases passed!
```

Question 2.5. The staff also created 70%, 90%, and 99% confidence intervals from the same sample, but we forgot to label which confidence interval represented which percentages! **First**, match each confidence level (70%, 90%, 99%) with its corresponding interval in the cell below

(e.g. ___ % CI: [52.1, 54] → replace the blank with one of the three confidence levels).

Then, explain your thought process and how you came up with your answers.

The intervals are below:

- [50.03, 55.94]
- [52.1, 54]
- [50.97, 54.99]

For the 99% confidence interval it shows that the interval is the widest, meaning it is the most range, thus it shows that it is very confident that the interval contains the true parameter.99% $\approx$ 5.9

For the 70% confidence level, it should be the one with the interval that is the narrowest that shows less precision in the estimate, as it is less confident about containing the true parameter compared to higher confidence levels.70% $\approx$ 1.9

For the 90% confidence the interval is wider than the 70%, while narrower than the 99%, thus the second largest confidence interval.90% $\approx$ 4.02

Question 4.1. Michelle wants to use 10,000 bootstrap resamples to compute a confidence interval for the proportion of all Colorado voters who will vote Yes.

a). Use bootstrap resampling to simulate 10,000 election outcomes, and assign the np.array `resample_yes_proportions` to contain the Yes proportion of each bootstrap resample.

b). Calculate the 95% bootstrapped confidence interval for the Yes proportion.

c). Then, plot a density histogram of `resample_yes_proportions`. Include a title and label both axes. **You should see a bell shaped curve centered near the proportion of Yes in the original sample.** We have provided code that overlays your confidence interval at the bottom of your histogram.

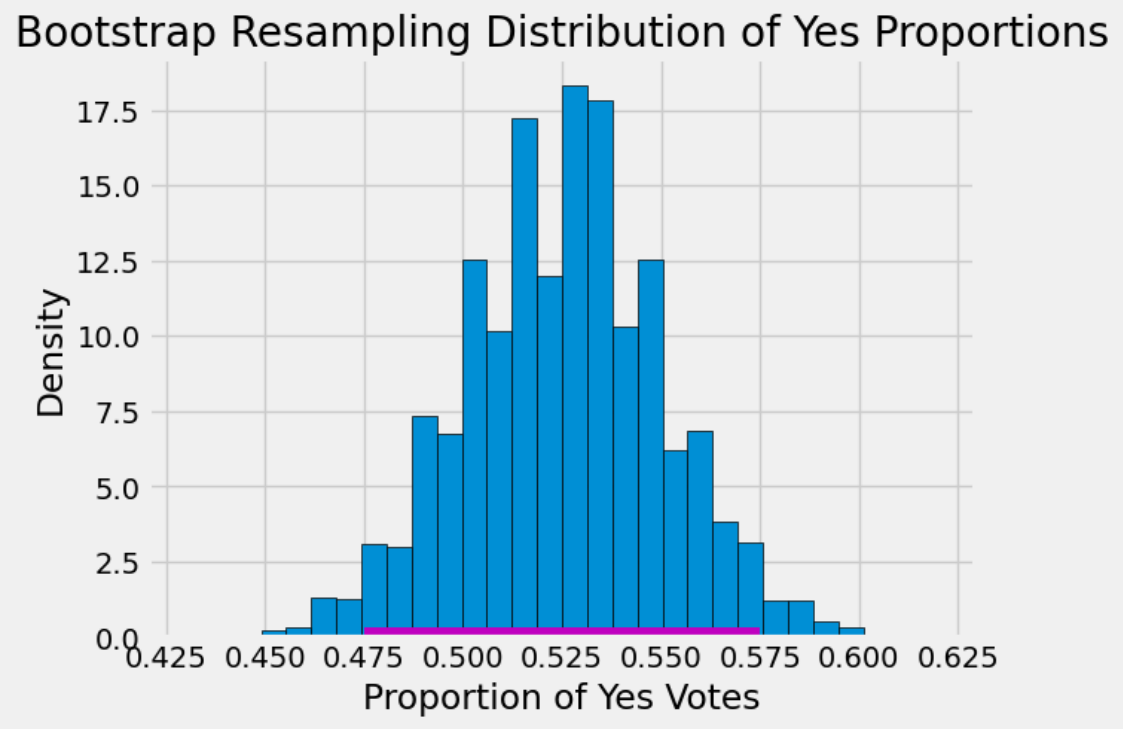
```
In [34]: sample = np.array([1] * 210 + [0] * 190)
         num_simulations = 10_000
         resampled_data = np.random.choice(sample, size=(num_simulations, len(sample)), replace=True)
         resample_yes_proportions = np.sum(resampled_data, axis=1) / len(sample)
```

```
In [35]: CI_lower = np.percentile(resample_yes_proportions, 2.5)
         CI_upper = np.percentile(resample_yes_proportions, 97.5)
         [CI_lower,CI_upper]
```

```
Out[35]: [0.475, 0.575]
```

```
In [36]: plt.hist(resample_yes_proportions, bins=30, density=True, edgecolor='black')
         plt.title('Bootstrap Resampling Distribution of Yes Proportions')
         plt.xlabel('Proportion of Yes Votes')
         plt.ylabel('Density')
         plt.plot(np.array([CI_lower, CI_upper]), np.array([0, 0]), c='m', lw=10)

         plt.show()
```




```
In [37]: grader.check("q4_1")
```

```
Out[37]: q4_1 results: All test cases passed!
```

Question 4.2.

a). Why does the Central Limit Theorem (CLT) apply in this situation, and how does it explain the distribution we see above?

b). Prove the following: In a population whose members are 0 or 1, the **standard deviation** of that population is:

$$\text{standard deviation of population} = \sqrt{(\text{proportion of 0s}) \times (\text{proportion of 1s})}$$

Write up your answers to both parts in the same Markdown cell below:

a) The CLT applies in this situation because the sample size bigger than 30, where it is a size of 10000 bootstrap samples. Each sampling is a proportion of people who votes yes out of the 400 people. This shows how the distribution is a bell shaped curve.

b) Let p be the value that someone votes yes, and p - 1 that someone votes no. SD = $\sqrt{Var(x)}$ $ E[X] = 1 p + 0 (1-p) = p \ E[X^2] = 1^2 p + 0(1-p) = p \ Var(x) = E[X^2] + (E[X])^2 = p - p^2 = p(1-p) $

SD_P = $\sqrt{(\text{proportion of 0s}) \times (\text{proportion of 1s})} = \sqrt{p(1-p)}$