Use the `head` command on your three files again. This time, describe at least one potential problem with the data you see. Consider issues with missing values and bad data.

One potential problem with the data I use is that some phone numbers are missing and instead have values such as -9999.

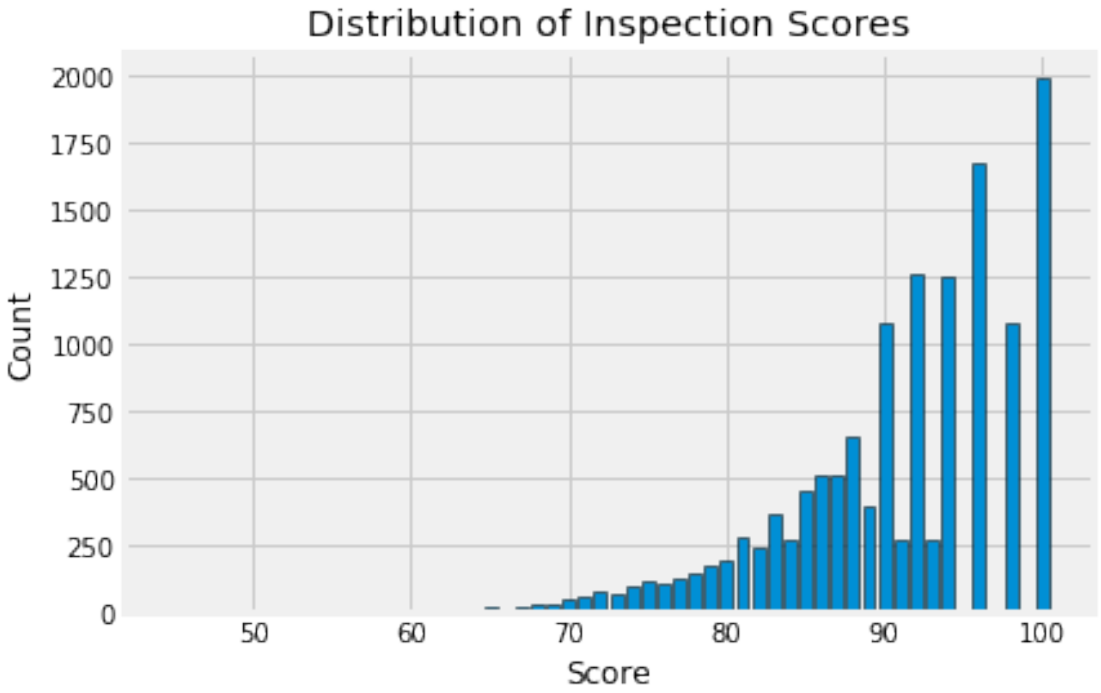
In the cell below, write the name of the restaurant with the lowest inspection scores ever. You can also head to yelp.com and look up the reviews page for this restaurant. Feel free to add anything interesting you want to share.

The name of the restaurant with the lowest inspection scores ever is Lollipot. Yelp said this restaurant is already closed.

0.1 Question 6a

Let's look at the distribution of inspection scores. As we saw before when we called head on this data frame, inspection scores appear to be integer values. The discreteness of this variable means that we can use a barplot to visualize the distribution of the inspection score. Make a bar plot of the counts of the number of inspections receiving each score.

It should look like the image below. It does not need to look exactly the same (e.g., no grid), but make sure that all labels and axes are correct.



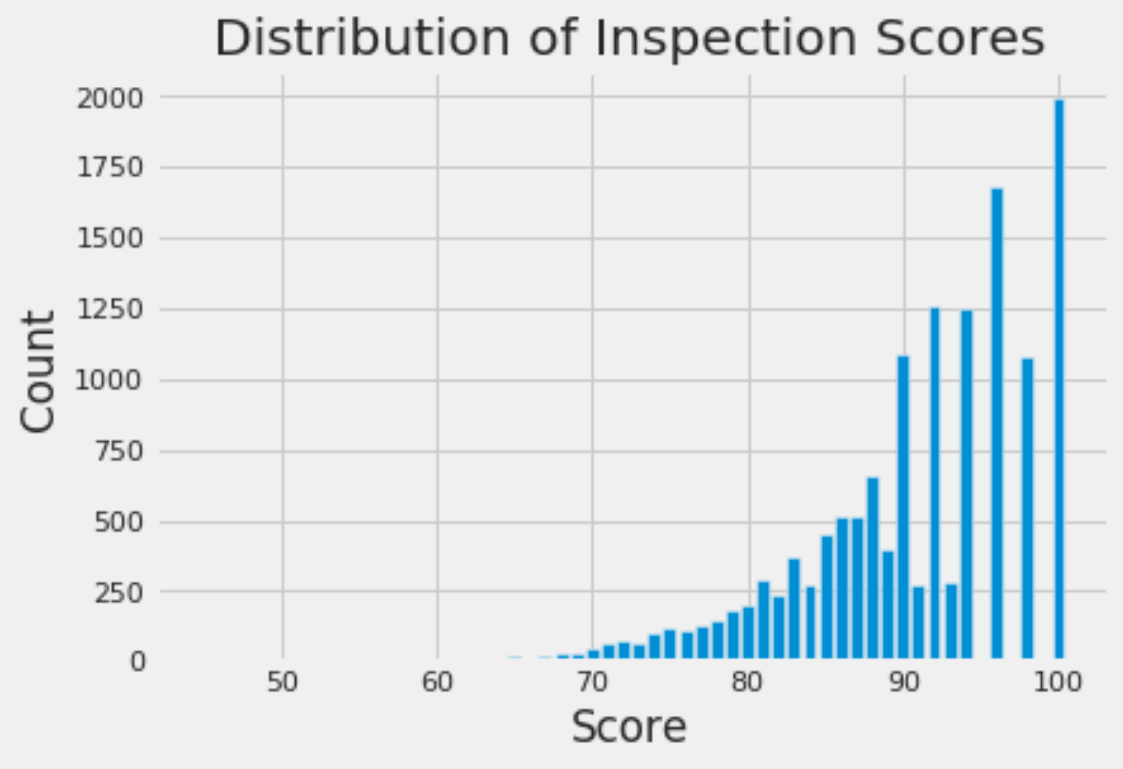
You might find this matplotlib.pyplot tutorial useful. Key syntax that you'll need:

```
plt.bar
plt.xlabel
plt.ylabel
plt.title
```

Note: If you want to use another plotting library for your plots (e.g. plotly, sns) you are welcome to use that library instead so long as it works on DataHub. If you use seaborn sns.countplot(), you may need to manually set what to display on xticks.

In [75]:
```
plt.bar(ins.groupby('score').size().index, ins.groupby('score').size().values )
plt.xlabel('Score')
plt.ylabel('Count')
plt.title('Distribution of Inspection Scores')
```

Out[75]:
```
Text(0.5, 1.0, 'Distribution of Inspection Scores')
```

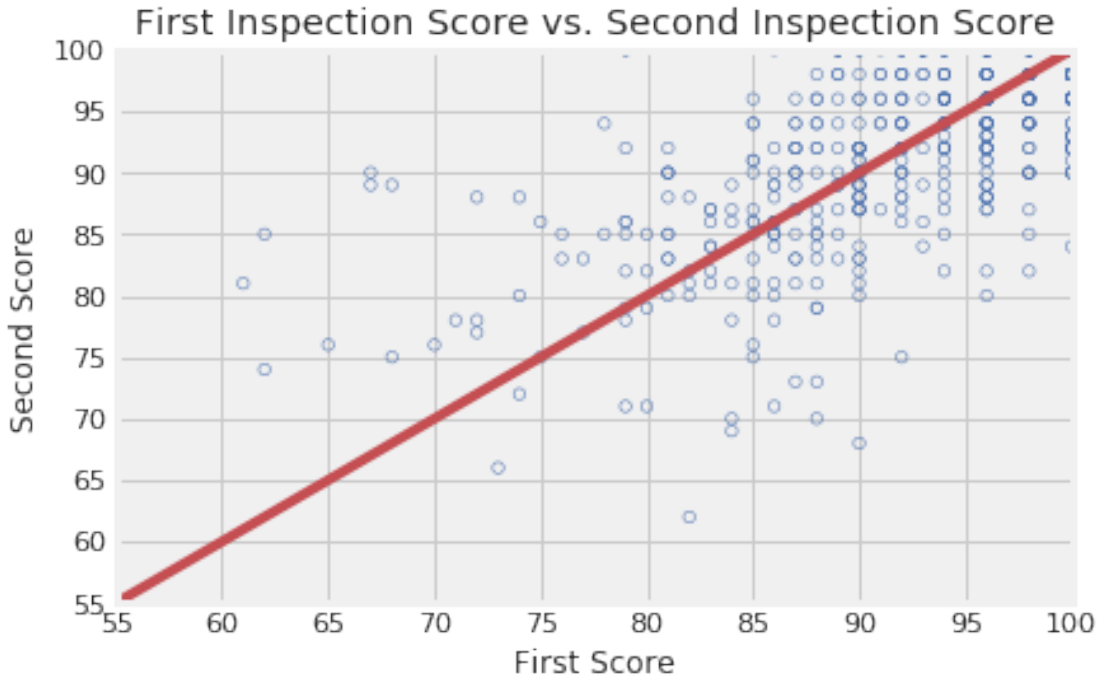


0.1.1 Question 6b

Describe the qualities of the distribution of the inspections scores based on your bar plot. Consider the mode(s), symmetry, tails, gaps, and anomalous values. Are there any unusual features of this distribution? What do your observations imply about the scores?

The mode of the bar plot is at 100 and the data is not symmetric at all. From the shape of the plot, the data is skewed left meaning that the mean is smaller than the median. This also means that most of the scores are on the right with higher value, with a few smaller number showing up on the left side. Moreover, the gaps increase as values increase, especially after the score hits 90. From the gaps, we can see that the variabel score is discrete since continuous data means no gap between the bars. There doesn't seem to have any unusual features of the distribution since anomalies are hard to determine without knowing the typical standard of scoring for these restaurants. Though it is to note that there is a small distribution situating at around score 65.

Now, create your scatter plot in the cell below. It does not need to look exactly the same (e.g., no grid) as the sample below, but make sure that all labels, axes and data itself are correct.



Key pieces of syntax you'll need:

`plt.scatter` plots a set of points. Use `facecolors='none'` and `edgecolors=b` to make circle markers with blue borders.

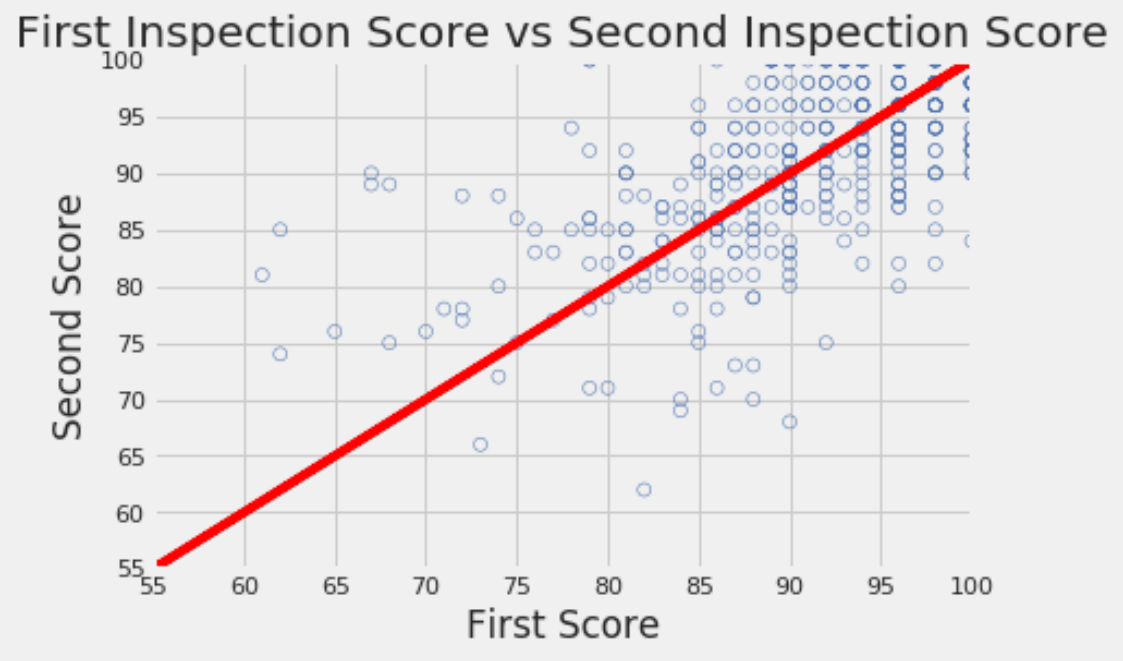
`plt.plot` for the reference line.

`plt.xlabel`, `plt.ylabel`, `plt.axis`, and `plt.title`.

Hint: You may find it convenient to use the `zip()` function to unzip scores in the list.

```
In [111]: x = tbl.iloc[:, 0].values
          y = tbl.iloc[:, 1].values
          plt.scatter(x, y, edgecolors = 'b', facecolors = 'none')
          plt.plot(x, x, 'red')
          plt.xlabel('First Score')
          plt.ylabel('Second Score')
          plt.axis(xmin = 55, ymin = 55, xmax = 100, ymax = 100)
          plt.title('First Inspection Score vs Second Inspection Score')
```

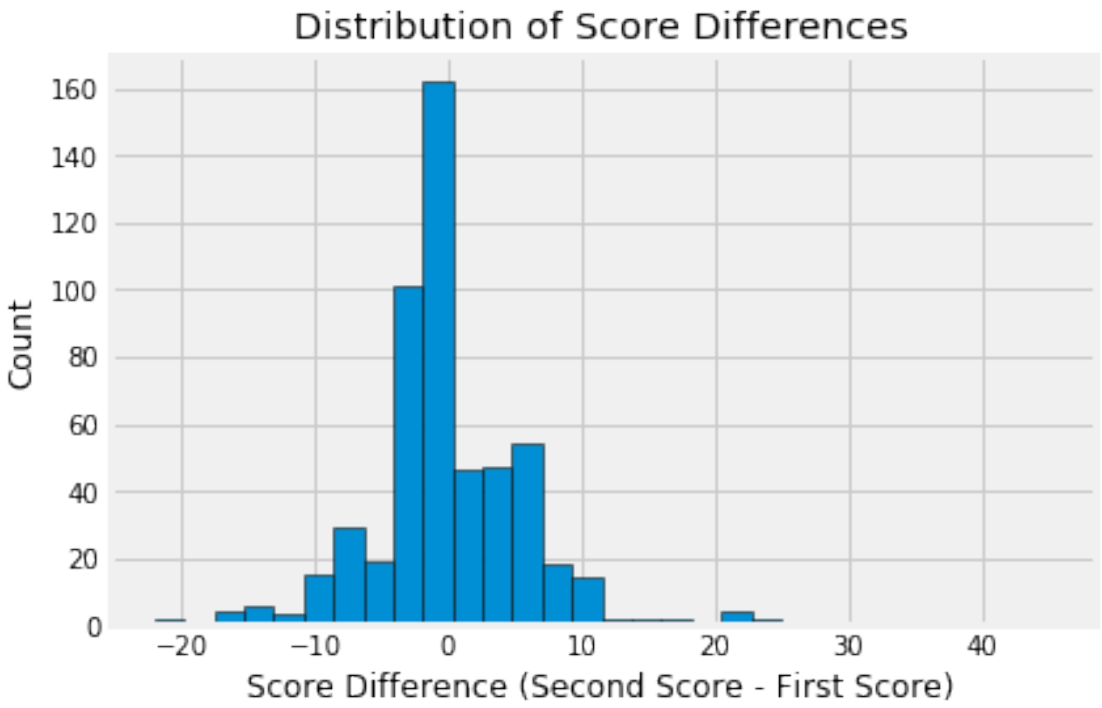
```
Out[111]: Text(0.5, 1.0, 'First Inspection Score vs Second Inspection Score')
```


First Inspection Score vs Second Inspection Score
Second Score
First Score
100
95
90
85
80
75
70
65
60
55
55
60
65
70
75
80
85
90
95
100

0.1.2 Question 7d

Another way to compare the scores from the two inspections is to examine the difference in scores. Subtract the first score from the second in `scores_pairs_by_business`. Make a histogram of these differences in the scores. We might expect these differences to be positive, indicating an improvement from the first to the second inspection.

The histogram should look like this:



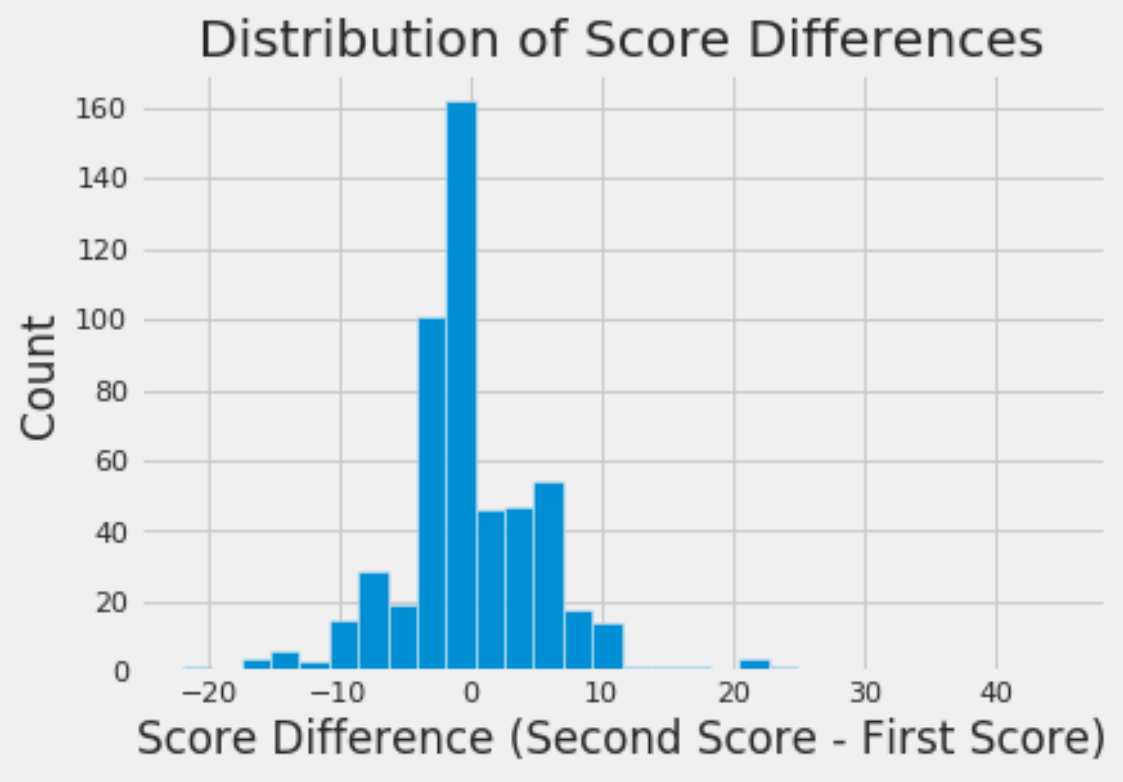
Hint: Use `second_score` and `first_score` created in the scatter plot code above.

Hint: Convert the scores into numpy arrays to make them easier to deal with.

Hint: Use `plt.hist()` Try changing the number of bins when you call `plt.hist()`.

```
In [123]: score_diff = y - x
          plt.hist(score_diff, bins = 30)
          plt.xlabel('Score Difference (Second Score - First Score)')
          plt.ylabel('Count')
          plt.title('Distribution of Score Differences')
```

Out[123]: Text(0.5, 1.0, 'Distribution of Score Differences')



0.1.3 Question 7e

If restaurants' scores tend to improve from the first to the second inspection, what do you expect to see in the scatter plot that you made in question 7c? What do you oberve from the plot? Are your observations consistent with your expectations?

Hint: What does the slope represent?

If restaurants' scores tend to improve from the first to the second inspection, I would expect to see a greater slope in the scatter plot. The slope represents the rate of change of second score relative to first score. Therefore if we see a greater increase in the second score relative to the first score, the slope would increase and become steeper. From the plot, the improvement tends to small, with data points clustering near the y = x reference line, which indicates that the difference between first and second score is small. Thus, my observations are not consistent with my expectations.

0.1.4 Question 7f

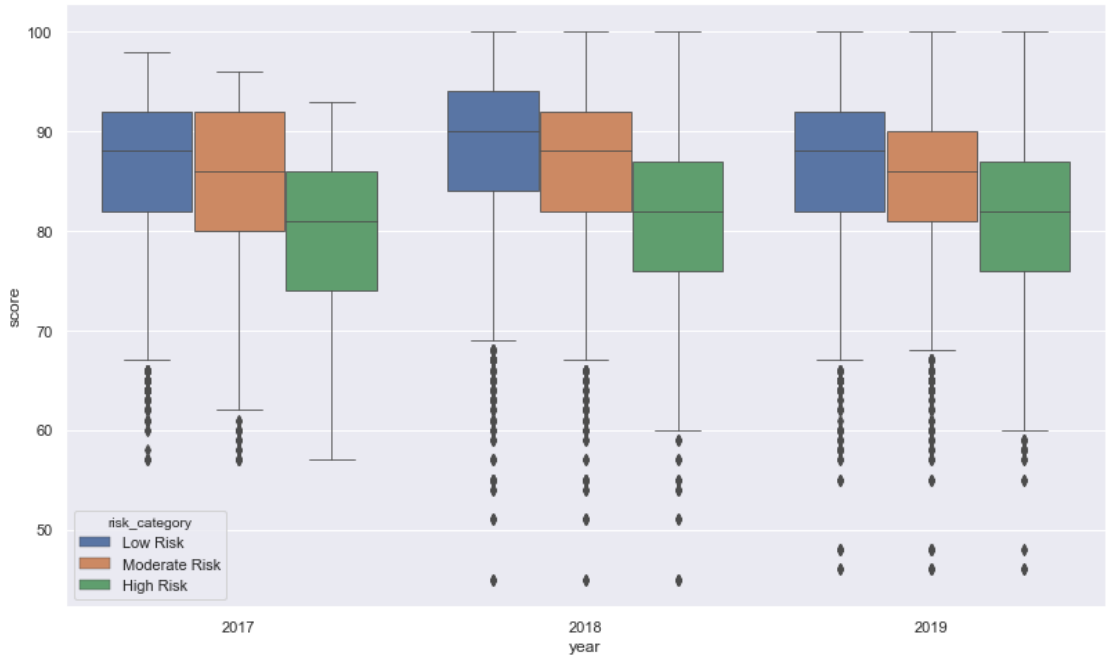
If a restaurant's score improves from the first to the second inspection, how would this be reflected in the histogram of the difference in the scores that you made in question 7d? What do you oberve from the plot? Are your observations consistent with your expectations? Explain your observations in the language of Statistics: for instance, the center, the spread, the deviation etc.

If a restaurant's score improves from the first to the second inspection, the center of the histogram would be on the right side of 0, indicating an increase in score. The mode is currently below 0 and if we expect an improvement, the mode should be positive. In terms of spread, the range of the score differences should also be positive, indicating an improvement instead of deterioation, which we should expect a negative value. From the plot, the range of score difference is from -20 to positive 20, with greater concentration around 0, meaning no or minimal improvement. Hence, I would say that my observations are not consistent with your expectations.

0.1.5 Question 7g

To wrap up our analysis of the restaurant ratings over time, one final metric we will be looking at is the distribution of restaurant scores over time. Create a side-by-side boxplot that shows the distribution of these scores for each different risk category from 2017 to 2019. Use a figure size of at least 12 by 8.

The boxplot should look similar to the sample below. Make sure the boxes are in the correct order!

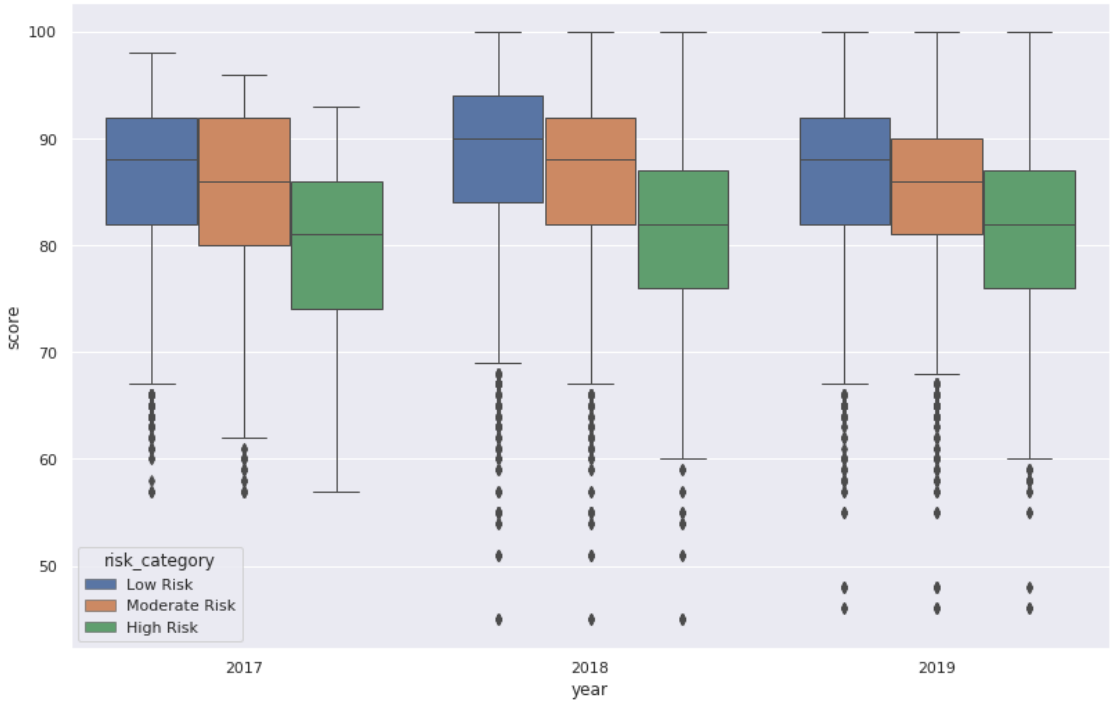


Hint: Use `sns.boxplot()`. Try taking a look at the first several parameters. The documentation is linked here!

Hint: Use `plt.figure()` to adjust the figure size of your plot.

```
In [127]: # Do not modify this line
          sns.set()
          ins_vid2 = ins.merge(ins2vio, how = 'left', on = 'iid')
          ins_vid2 = ins_vid2[ins_vid2['year'] > 2016]
          vio_n = vio[vio['vid'].isin(ins_vid2['vid'])].merge(ins_vid2.iloc[:, [2, 6, 8]], how = 'left'
          plt.figure(figsize=[12, 8])
          sns.boxplot(x = 'year', y = 'score', hue = 'risk_category', data = vio_n, linewidth=1, hue_or
```

```
Out[127]: <matplotlib.axes._subplots.AxesSubplot at 0x7f33d68a1400>
```


100
90
80
70
60
50
score
risk_category
Low Risk
Moderate Risk
High Risk
2017
2018
2019
year

1 8: Open Ended Question

1.1 Question 8a

1.1.1 Compute Something Interesting

Play with the data and try to compute something interesting about the data. Please try to use at least one of groupby, pivot, or merge (or all of the above).

Please show your work in the cell below and describe in words what you found in the same cell. This question will be graded leniently but good solutions may be used to create future homework problems.

1.1.2 Grading

Since the question is more open ended, we will have a more relaxed rubric, classifying your answers into the following three categories:

- **Great** (4 points): Uses a combination of pandas operations (such as groupby, pivot, merge) to answer a relevant question about the data. The text description provides a reasonable interpretation of the result.
- **Passing** (1-3 points): Computation is flawed or very simple. The text description is incomplete but makes some sense.
- **Unsatisfactory** (0 points): No computation is performed, or a computation with completely wrong results.

Please have both your code and your explanation in the same one cell below. Any work in any other cell will not be graded.

In [142]:
```
#YOUR CODE HERE

new_ins = ins.merge(bus.iloc[:, [0, 1, 9]], on = 'bid', how = 'left')
new_ins = new_ins.merge(ins2vio).merge(vio)
new_pivot = new_ins.pivot_table(index = 'name', columns = 'risk_category', values = 'vid', ag
highest_risk_rest = new_pivot[new_pivot['High Risk'] == max(new_pivot['High Risk'].values)].i
#YOUR EXPLANATION HERE (in a comment)
# What are the names of the restaurants with the highest count of high risk violations?
# First merge the ins dataframe with the bus dataframe to include the name of the businesses.
# Then merge the created dataframe with vio table to include the vid, resulting in the table
```

```
# Pivot the table to show the number of violations for each business under the categories hig
# Finally, select the dataframe with only the maximum number of high risk violations. Get the
```

1.1.3 Grading

Since the question is more open ended, we will have a more relaxed rubric, classifying your answers into the following three categories:

- **Great** (4 points): The chart is well designed, and the data computation is correct. The text written articulates a reasonable metric and correctly describes the relevant insight and answer to the question you are interested in.
- **Passing** (1-3 points): A chart is produced but with some flaws such as bad encoding. The text written is incomplete but makes some sense.
- **Unsatisfactory** (0 points): No chart is created, or a chart with completely wrong results.

We will lean towards being generous with the grading. We might also either discuss in discussion or post on Piazza some examplar analysis you have done (with your permission)!

You should have the following in your answers: * a few visualizations; Please limit your visualizations to 5 plots. * a few sentences (not too long please!)

Please note that you will only receive support in OH and Piazza for Matplotlib and seaborn questions. However, you may use some other Python libraries to help you create you visualizations. If you do so, make sure it is compatible with the PDF export (e.g., Plotly does not create PDFs properly, which we need for Gradescope).

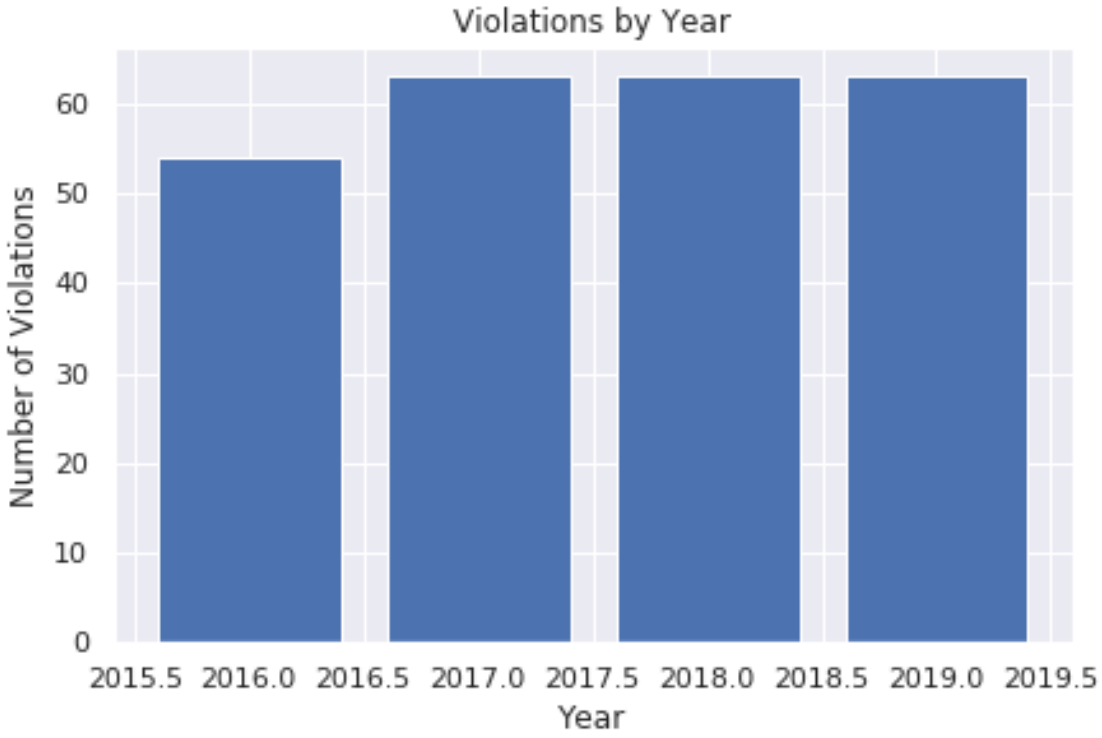
In [228]:
```
# YOUR DATA PROCESSING AND PLOTTING HERE
new_ins2 = ins.merge(ins2vio).merge(vio)
new_pivot2 = new_ins2.pivot_table(index = 'year', columns = 'risk_category', values = 'vid', a

x = list(new_pivot2.index)
height = new_pivot2.sum(axis = 1).values
plot1 = plt.figure(1)
plt.bar(x, height, align='center')
plt.xlabel('Year')
plt.ylabel('Number of Violations')
plt.title('Violations by Year')

x2 = list(new_pivot2.columns)
height2 = new_pivot2.sum(axis = 0).values
plot2 = plt.figure(2)
plt.bar(x2, height2, align='center')
plt.xlabel('Risk Category')
plt.ylabel('Number of Violations')
plt.title('Violations by Risk Category')

# YOUR EXPLANATION HERE (in a comment)
# The first line merges ins and vio to include the year; the second line of code pivots the t
# the number of violations each year by risk category.
# Two bar graphs are illustrated to see the number of violations by year and by risk category
# is the sum of inspections for each of the cateogories (year vs risk category)
```

Out[228]: Text(0.5, 1.0, 'Violations by Risk Category')




Violations by Risk Category
Number of Violations
0
20
40
60
80
100
High Risk
Low Risk
Moderate Risk
Risk Category