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.

There are some missing value which represnet by negative values. If we use them for calculation such as mean, it would cause errors.

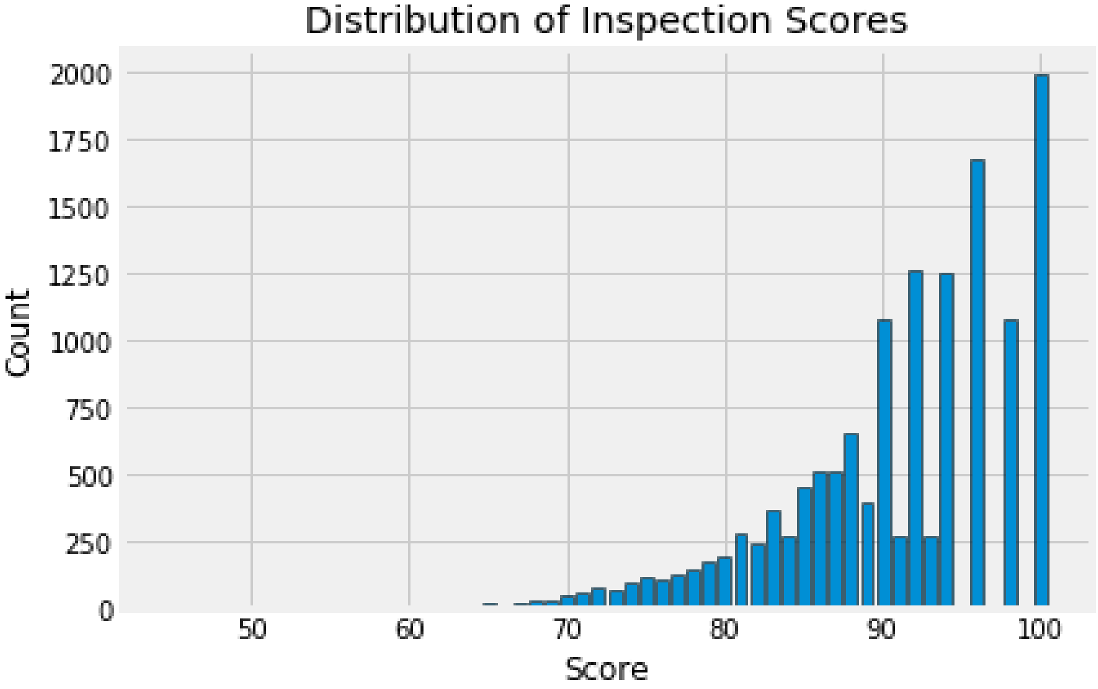
In the cell below, write the name of the restaurant with the lowest inspection scores ever. You can also head to yelp.com and look up the reviews page for this restaurant. Feel free to add anything interesting you want to share.

Lollipot

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.



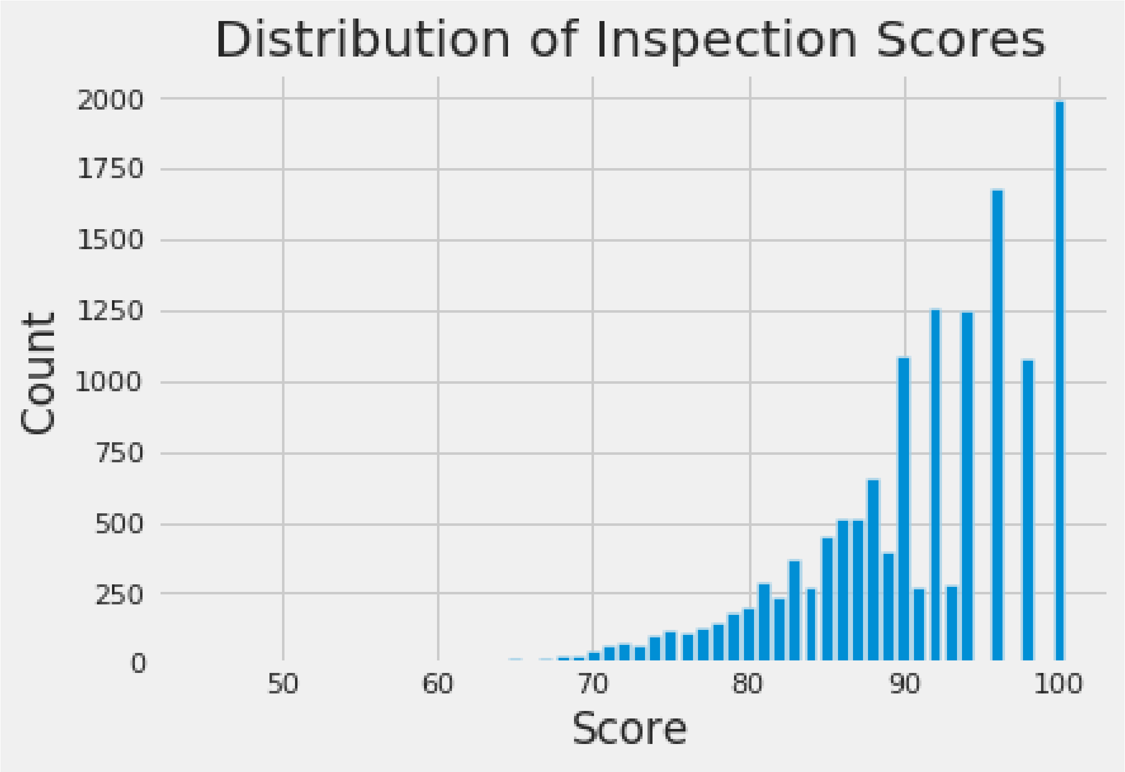
You might find this matplotlib.pyplot tutorial useful. Key syntax that you'll need:

```
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 [76]: plt.bar(ins['score'].value_counts().keys(),ins['score'].value_counts())
         plt.xlabel('Score')
         plt.ylabel('Count')
         plt.title('Distribution of Inspection Scores')
```

```
Out[76]: Text(0.5, 1.0, 'Distribution of Inspection Scores')
```

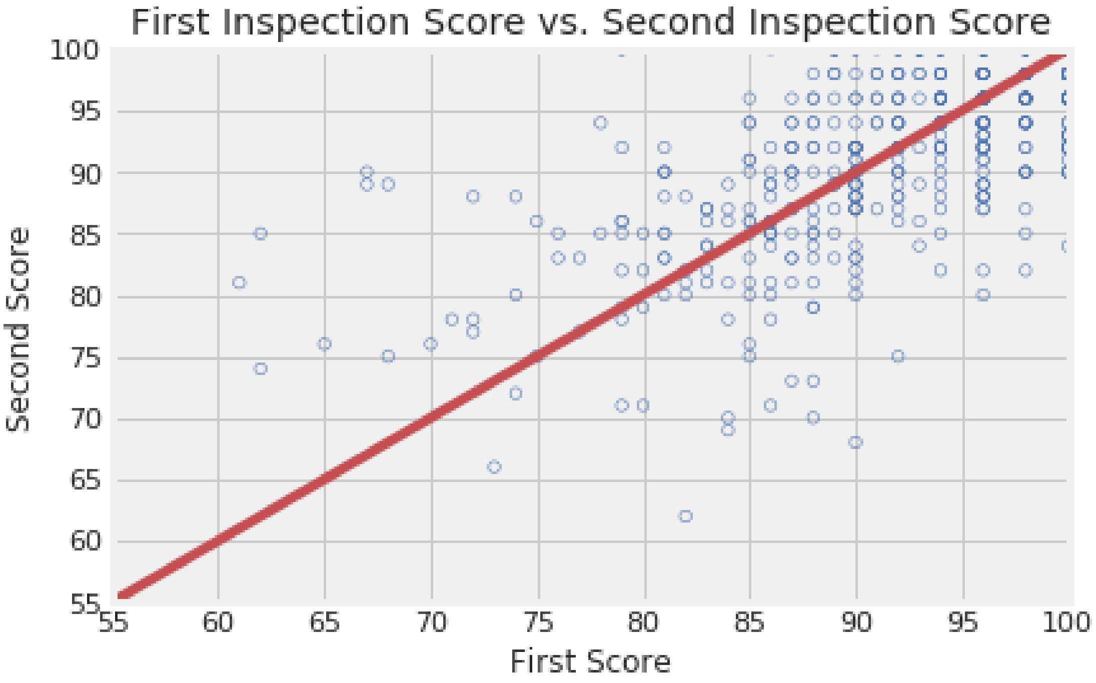


0.1.1 Question 6b

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 highest score for this model is 100 points. We can clearly see that the graph is not symmetrical, but the overall count number is increasing from left to right, with a peak value of 100. There are a few small gaps in the interval of 90-100 that seem unusual, because their quantity has been reduced a lot compare to the trend. In general, most restaurants score very high, it seem that no restaurant has a score lower than 60. This may be due to the fact that the score is too low and no people come leads to closure.

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.

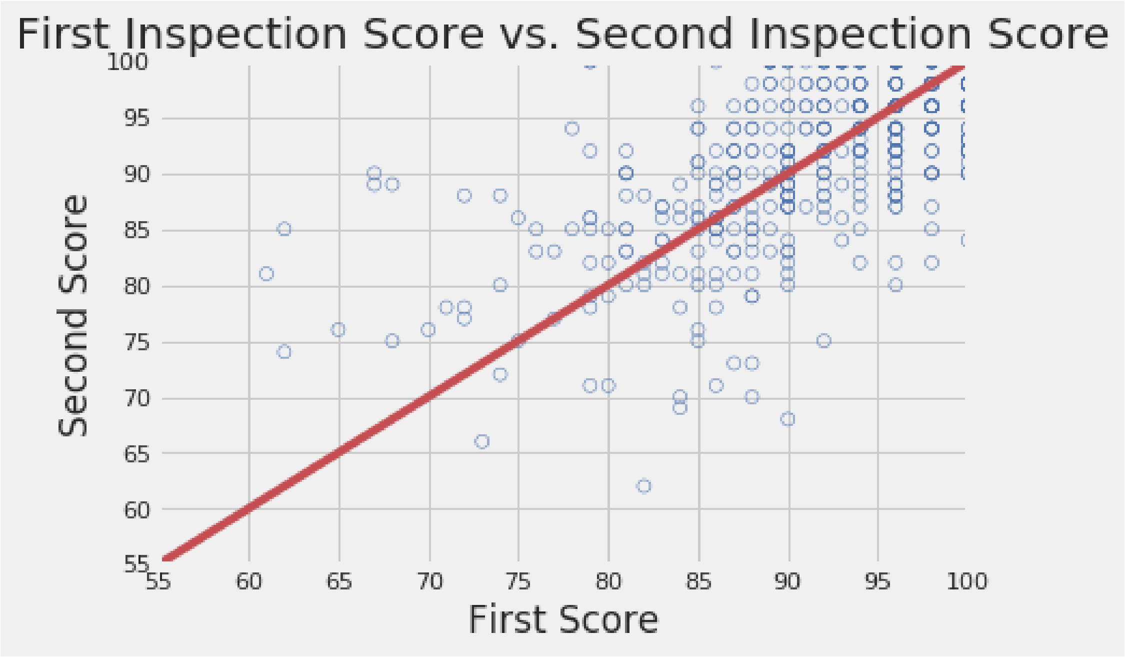
`plt.plot` for the reference line.

`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 [85]: first_score, second_score = zip(*scores_pairs_by_business['score_pair'])
         plt.scatter(first_score, second_score, facecolors='none', edgecolors = 'b')
         plt.plot([55,100],[55,100], 'r-')
         plt.xlabel('First Score')
         plt.ylabel('Second Score')
         plt.title('First Inspection Score vs. Second Inspection Score')
         plt.axis([55, 100, 55, 100])
```

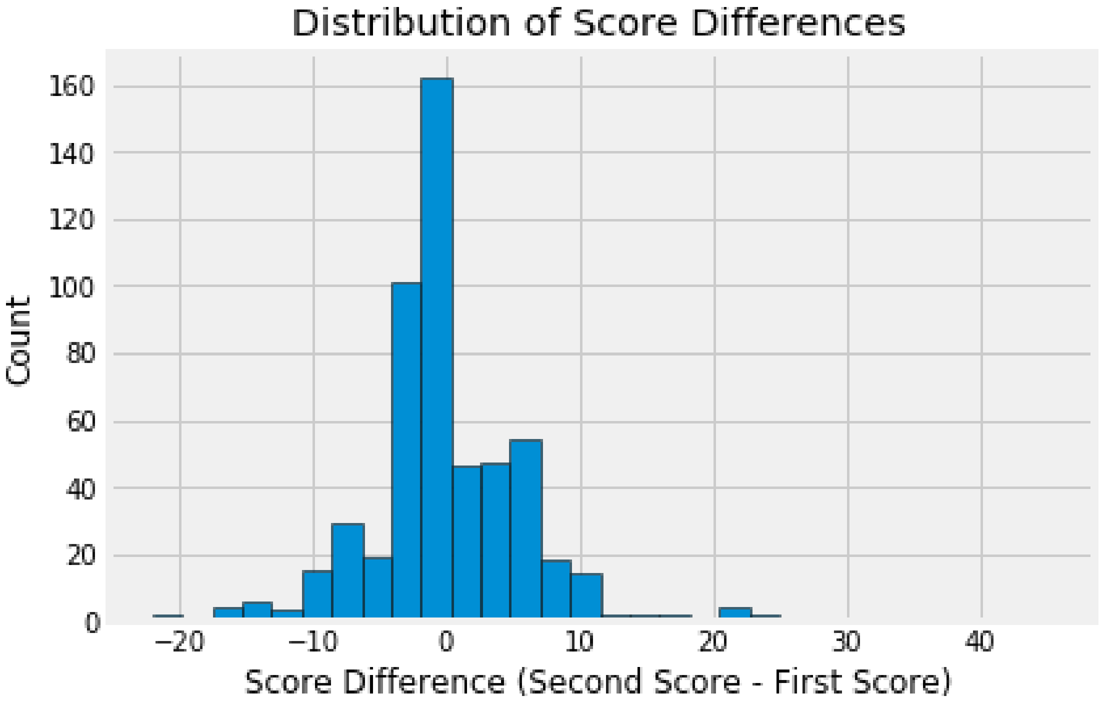
```
Out[85]: [55, 100, 55, 100]
```


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

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:

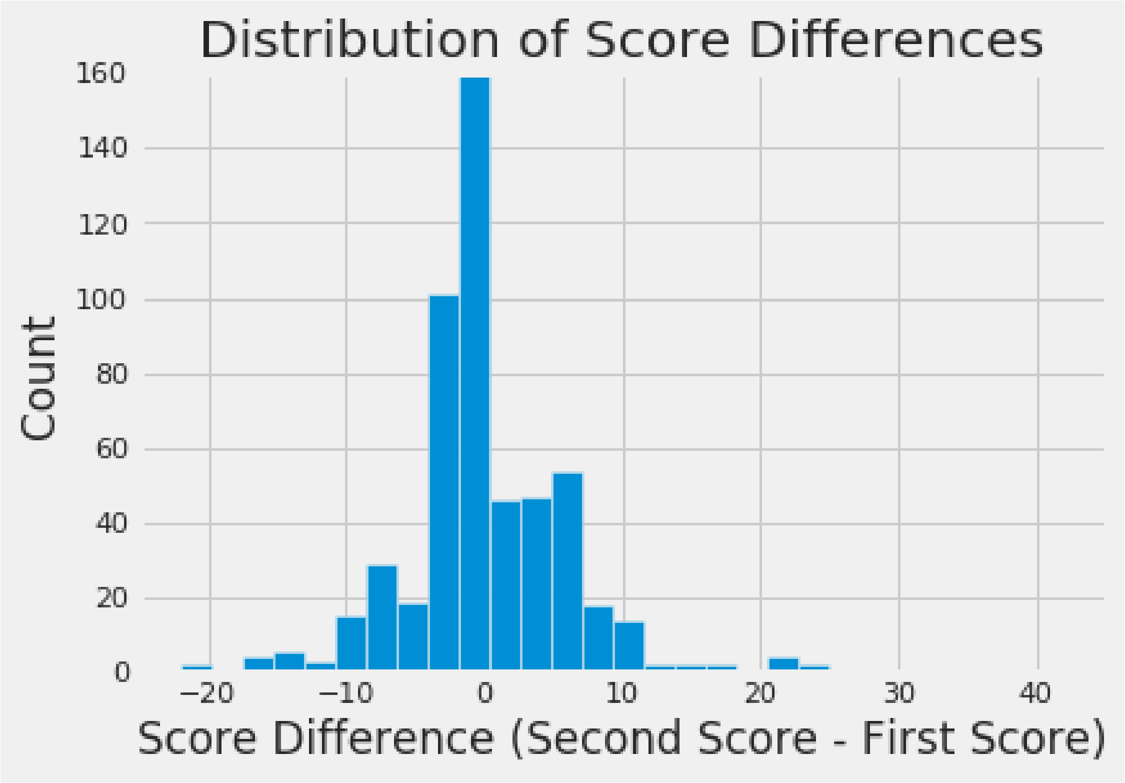


Hint: Use `second_score` and `first_score` created in the scatter plot code above.

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 [86]: plt.hist(np.array(second_score) - np.array(first_score), bins = 30)
         plt.axis([-25, 45, 0, 160])
         plt.xlabel('Score Difference (Second Score - First Score)')
         plt.ylabel('Count')
         plt.title('Distribution of Score Differences');
```


Distribution of Score Differences
Count
0
20
40
60
80
100
120
140
160
−20
−10
0
10
20
30
40
Score Difference (Second Score - First Score)

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?

We know that the slope of the reference line is 1, which means that the points lower than the reference line means that the second time the score is lower than the first time, and the points higher than the reference line means the second time is higher than the first time. If restaurants' scores tend to improve from the first to the second inspection, the second point will appear above the reference line. I found that the number of points above and below the reference line is about the same. This is what I expect.

0.1.4 Question 7f

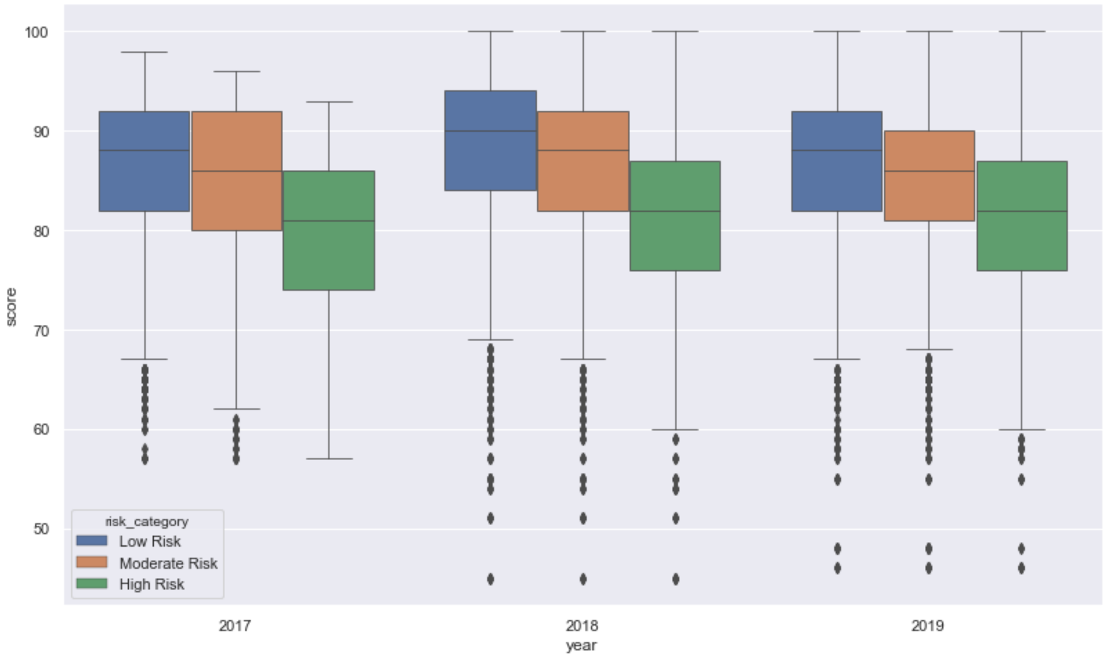
If a restaurant's score improves from the first to the second inspection, how would this be reflected in the histogram of the difference in the scores that you made in question 7d? What do you oberve from the plot? Are your observations consistent with your expectations? Explain your observations in 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 hole graph should shift right, the peak will be to the right of the x-axis center(in the positive side) and the total amount of quantity in the right side(positive number) of the x-axis center(0) is larger than the left side(negative number). From the figure we can see that the peak appears near 0. In addition, in the interval no longer near 0, the deviation seems to be relatively large. Half to the left of 0, half to the right. Therefore, half restaurant improved, and half got worse. This is exactly the conclusion we reached in the previous question, so it meets my 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!



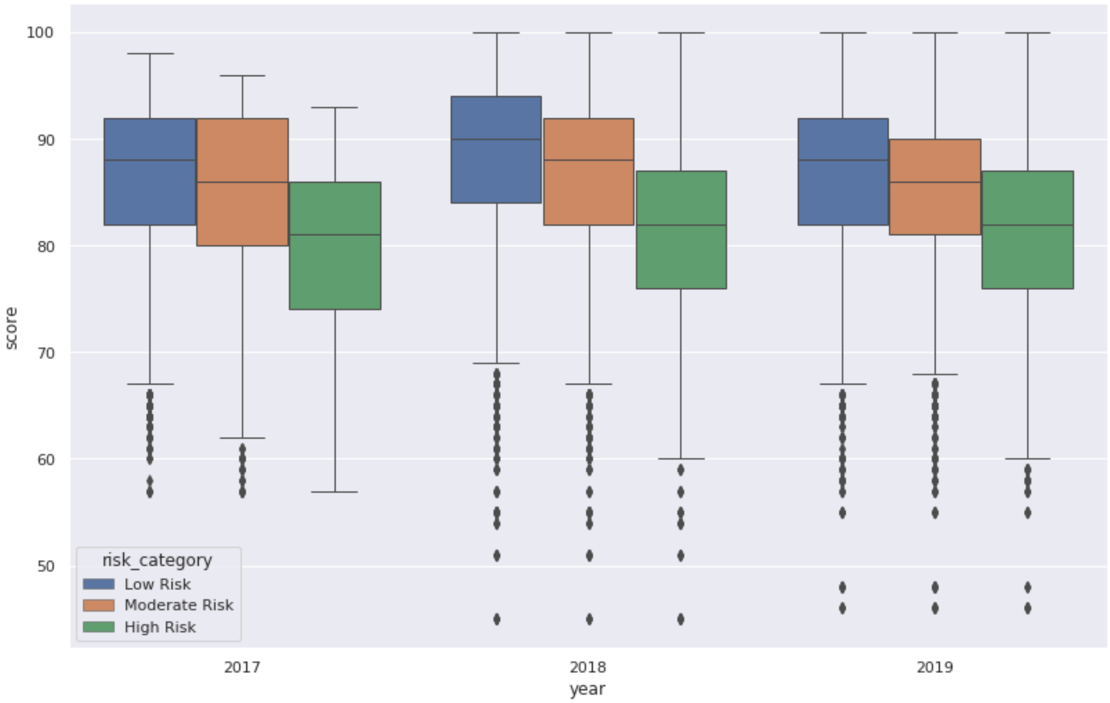
Hint: Use `sns.boxplot()`. Try taking a look at the first several parameters. The documentation is linked here!

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

```
In [87]: # Do not modify this line
         sns.set()

         tables = ins[(ins['year']>= 2017) & (ins['year']<=2019)]
         tables = pd.merge(tables, ins2vio, how = 'left', left_on = "iid", right_on = "iid").merge(vio,
         plt.figure(figsize = (12,8))
         sns.boxplot(x="year", y="score", hue="risk_category", data=tables, hue_order = ["Low Risk", "M
```

```
Out[87]: <matplotlib.axes._subplots.AxesSubplot at 0x7f0b1d32de20>
```


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 [88]: data = pd.merge(ins, bus[['name', 'address','bid']], how = 'left', on = 'bid')
         data = pd.merge(data ,ins2vio, how = 'left', left_on='iid', right_on='iid') .merge(vio, on = '
         data = data.groupby(['risk_category','year']).mean().rename(columns = {"score": "mean score"})
         data
         # I want to know the relationship between risk_category and score. As a reasult, it can be cle
         # from high to low is Low Risk, Moderate Risk, High Risk. This fits what I thought
```

```
Out[88]:                               mean score
         risk_category year
```

```
High Risk     2016   81.428571
              2017   79.980589
              2018   80.724436
              2019   80.726178
Low Risk      2016   87.483525
              2017   86.852113
              2018   87.476642
              2019   86.740297
Moderate Risk 2016   85.567427
              2017   85.060494
              2018   85.815690
              2019   85.252673
```

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.

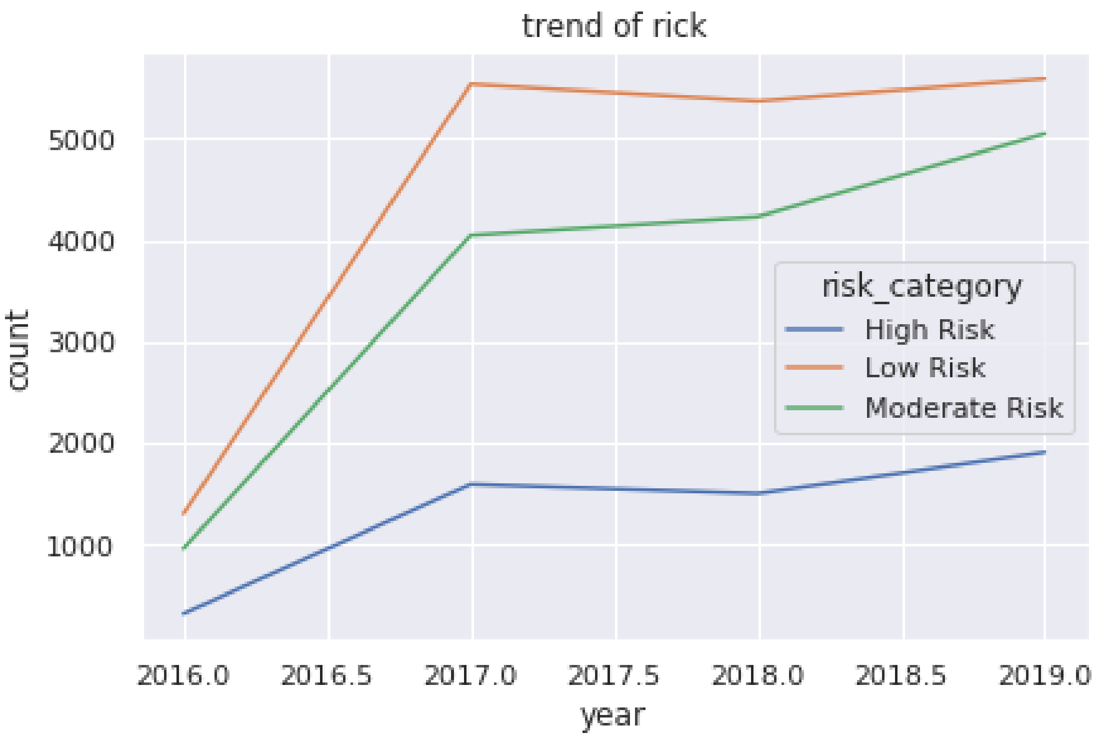
We will lean towards being generous with the grading. We might also either discuss in discussion or post on Piazza some examplar analysis you have done (with your permission)!

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 [89]: # YOUR DATA PROCESSING AND PLOTTING HERE
         graph = pd.merge(ins, bus[['name', 'address','bid']], how = 'left', on = 'bid')
         graph = pd.merge(graph ,ins2vio, how = 'left', left_on='iid', right_on='iid') .merge(vio, on =
         graph = graph.groupby(['year','risk_category']).count().rename(columns = {"score": "count"})
         graph['count'].unstack().plot(ylabel = 'count', title = 'trend of rick')
         # I want to know the trend of Low Risk, Moderate Risk, High Risk in recent years.
         # We can see from the figure that the rapid growth from 2016 to 2017.
         # From 2017 to 2019, the whole graph rose steadily, among which High Risk decreased from 2017
```

```
Out[89]: <matplotlib.axes._subplots.AxesSubplot at 0x7f0b1d2d4b80>
```


trend of rick
5000
4000
3000
2000
1000
count
2016.0
2016.5
2017.0
2017.5
2018.0
2018.5
2019.0
year
risk_category
High Risk
Low Risk
Moderate Risk