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.

Some of the rows in bus appears to be missing actual longitude and latitude data, which is odd, especially if some of the restaurants do have actual longitude and latitude data. It could make locating said restaurants with the missing data harder. Similarly, some of the entries in ins appear to have uneven scores or missing scores of -1. This could make calculating the average score or using the score for comparisons and further analysis difficult.

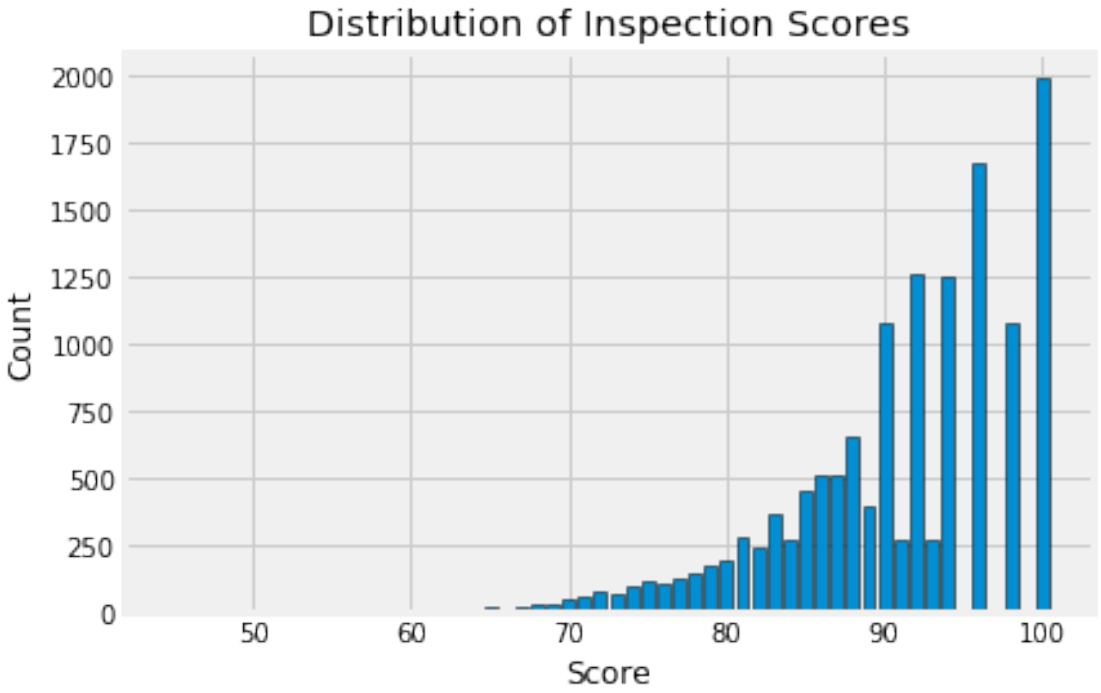
In the cell below, write the name of the restaurant with the lowest inspection scores ever. You can also head to yelp.com and look up the reviews page for this restaurant. Feel free to add anything interesting you want to share.

The restaurant with the lowest inspection scores ever was Lollipot, a hot pot restaurant. Ironically, on Yelp, it has an average star review, which fits with its inspection score.

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.

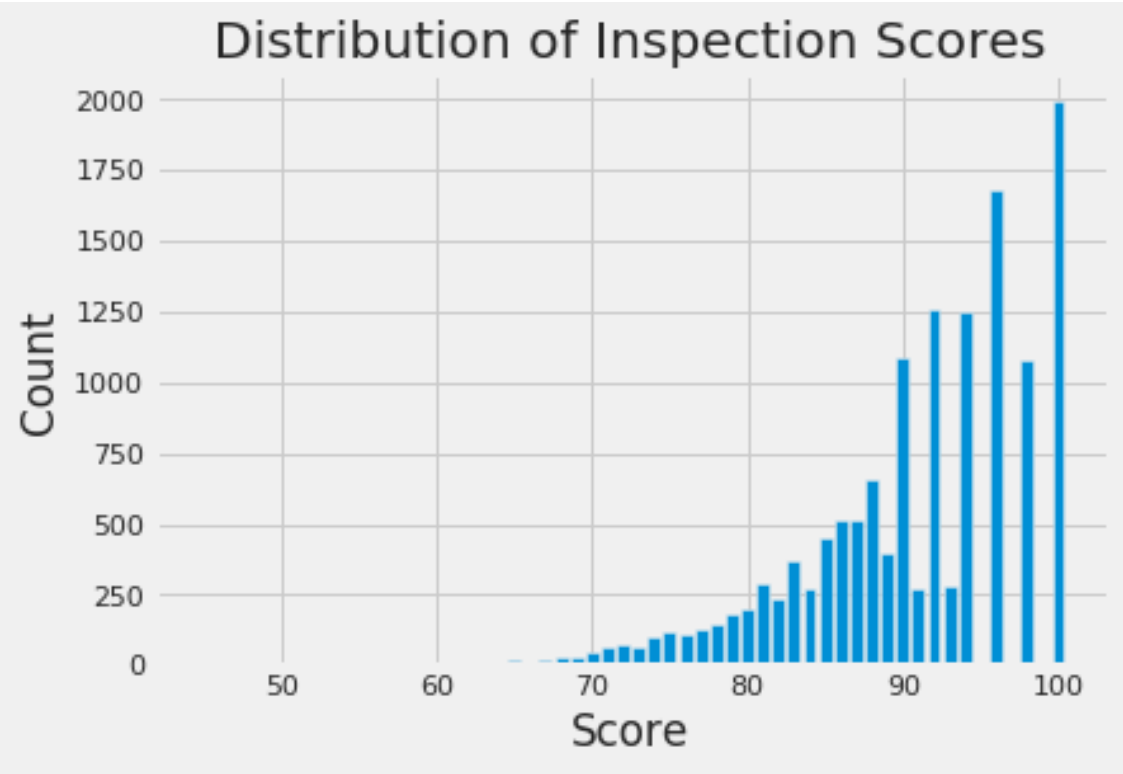


You might find this matplotlib.pyplot tutorial useful. Key syntax that you'll need:

```
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 [430]:
```
scores = ins_named.groupby("score", as_index=False).agg("size").rename(columns={"size":"count
score_counts = plt.bar(scores["score"], scores["count"])
score_counts = plt.xlabel("Score")
score_counts = plt.ylabel("Count")
score_counts = plt.title("Distribution of Inspection Scores");
```

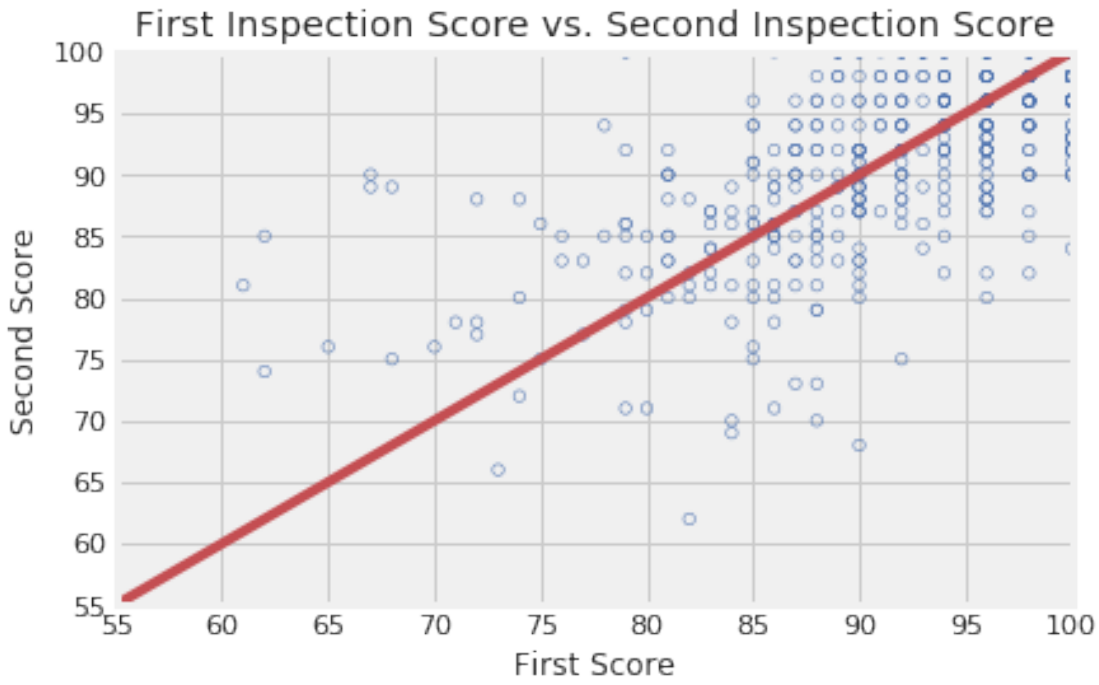


0.1.1 Question 6b

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?

Although not entirely uniformly, the distribution of the inspection scores appears to follow a an increasing/upward trend based on my bar plot. Mostly, it appears that as the score increases, so does its count. The mode appears to be a score of 100 with a count of almost 2000, meaning that almost 2000 restaurants in San Francisco received a full score. The bar plot does not appear symmetric and rather has a pretty prominent right tail. The gaps in between scores is pretty narrow from scoes ranging between 65 to 92-ish. Wider gaps appear between scores of 90 to 100. There are a few anomalous values where a lower number of restaurants appeared to receive scores of 91 and 93 compared to the rest of the scores in that range; there also appears to be another dip in count around a score of 89. These are the most unusual features of this distribution, and based on these observations, I believe that San Francisco could either have a large number of highly-rated restaurants or that the restaurant safety inspectors are a bit generous with their higher scores.

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.

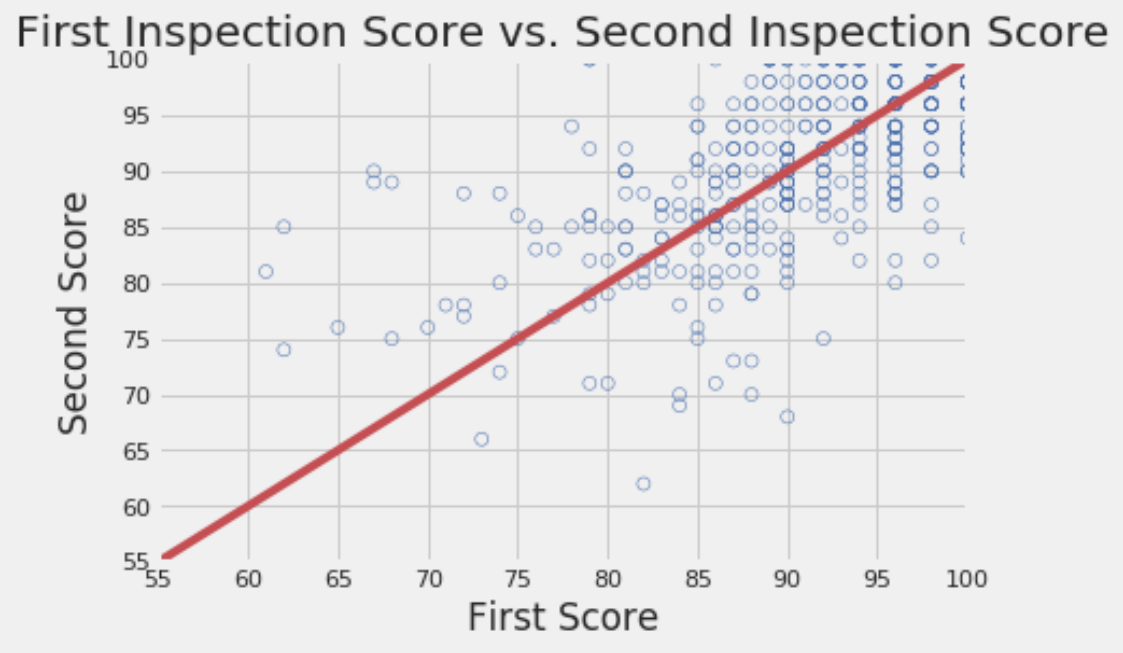
`plt.plot` for the reference line.

`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 [439]: score_pair = scores_pairs_by_business.reset_index()
          score_pair.head()
          score_pair["first"] = [i for i, j in score_pair["score_pair"]]
          score_pair["last"] = [j for i, j in score_pair["score_pair"]]
          score_pair.head()
          plt.xlim(55, 100)
          plt.ylim(55, 100)
          plt.scatter(score_pair["first"], score_pair["last"], facecolors="none", edgecolors="b")
          plt.xlabel("First Score")
          plt.ylabel("Second Score")
          plt.title("First Inspection Score vs. Second Inspection Score")
```

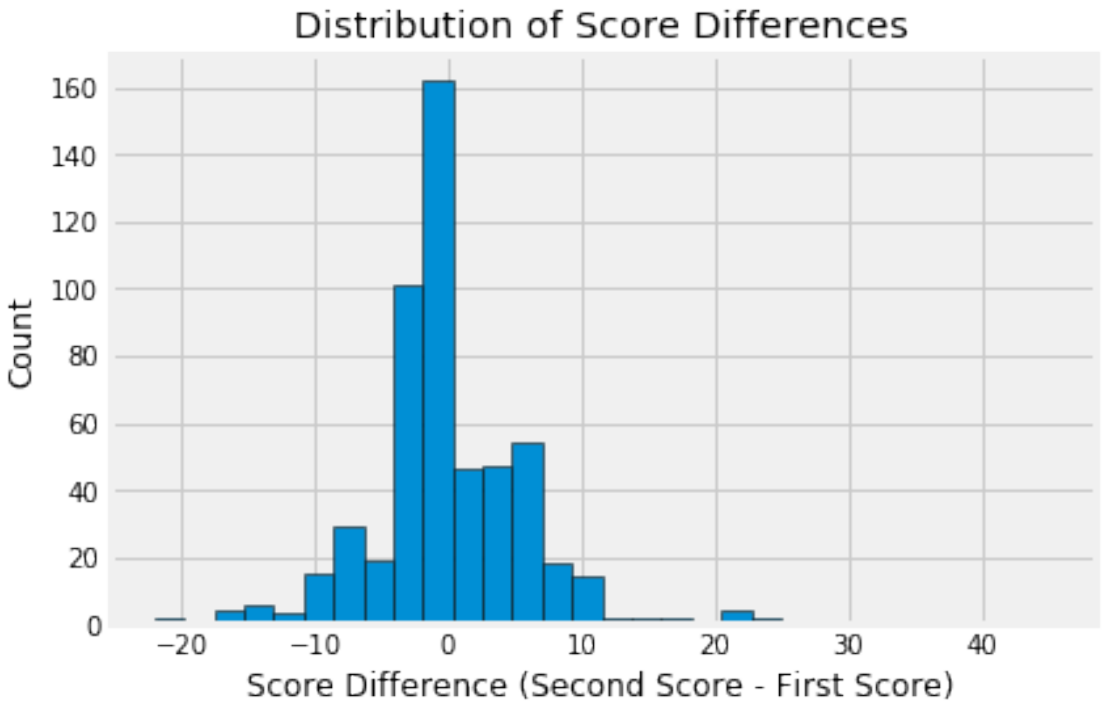
```
x = range(55, 105, 5)
y = range(55, 105, 5)
plt.plot(x, y, color="r")
plt.show()
```



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()`.

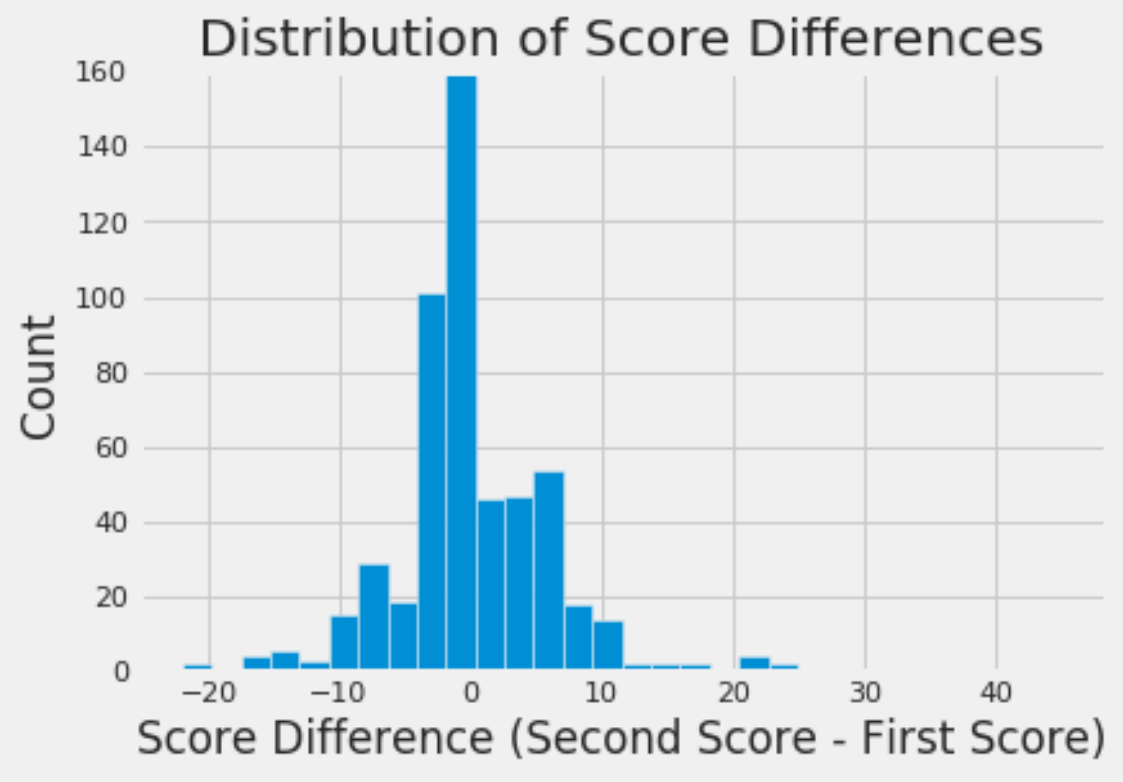
```
In [440]: score_pair.head()
```

```
Out[440]:    bid score_pair  first  last
          0   48   (94, 87)     94    87
```

```
1   66   (98, 98)     98    98
2  146   (81, 90)     81    90
3  184   (90, 96)     90    96
4  273   (83, 84)     83    84
```

In [441]:
```
diff = score_pair["last"].values - score_pair["first"].values
plt.ylim(0, 160)
plt.hist(diff, bins=30)
plt.xlabel("Score Difference (Second Score - First Score)")
plt.ylabel("Count")
plt.title("Distribution of Score Differences")
```

Out[441]:
```
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 expect to see most of the points in the scatter plot that I made in question 7c clustering perhaps towards the top leftmost quadrants of the plot and more scores to the left of the diagonal x=y, because you'd expect the y-value, aka the second score, to be greater than the first score, aka the x-value. Since the slope basically represents all the scores that remained the same for both inspections, more scores would be expected to be on the left side of the diagonal if they'd improved dramatically. However, that is not the case with the scatter plot. In actuality, more scores are clustered in the top righthand corner and along the slope. My observations are not consistent with my expectations. It does not seem that the scores improved by that much. It is highly likely that the first scores for most restaurants were already decently high (on the righthand side of the x-axis) and improved just enough to rise on the y-axis as well

---

0.1.4 Question 7f

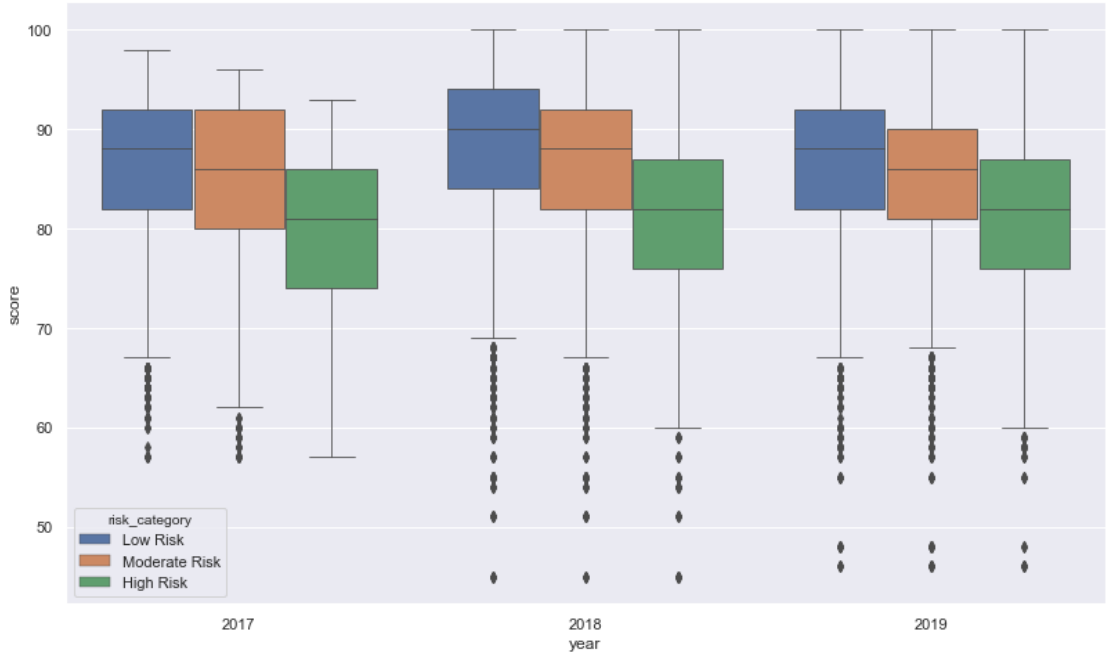
If a restaurant's score improves from the first to the second inspection, how would this be reflected in the histogram of the difference in the scores that you made in question 7d? What do you oberve from the plot? Are your observations consistent with your expectations? Explain your observations in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

If restaurants' scores tend to improve from the first to the second inspection, I would expect to see more bars on the righthand side of the histogram of the difference in the scores that I made in question 7d. This would imply that there was a general greater positive score difference (aka improvement) from the first inspection to the second, thus making it right-tailed. However, this is not the case. The histogram does appear to be centered slightly around 0, ranges from around -25 to around 25, and is generally symmetric-seeming. This implies that there is an equal number, or close to equal number, of restaurants whose scores increased and restaurants whose scores decreased among inspections. Thus, my observations are not consistent with 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!

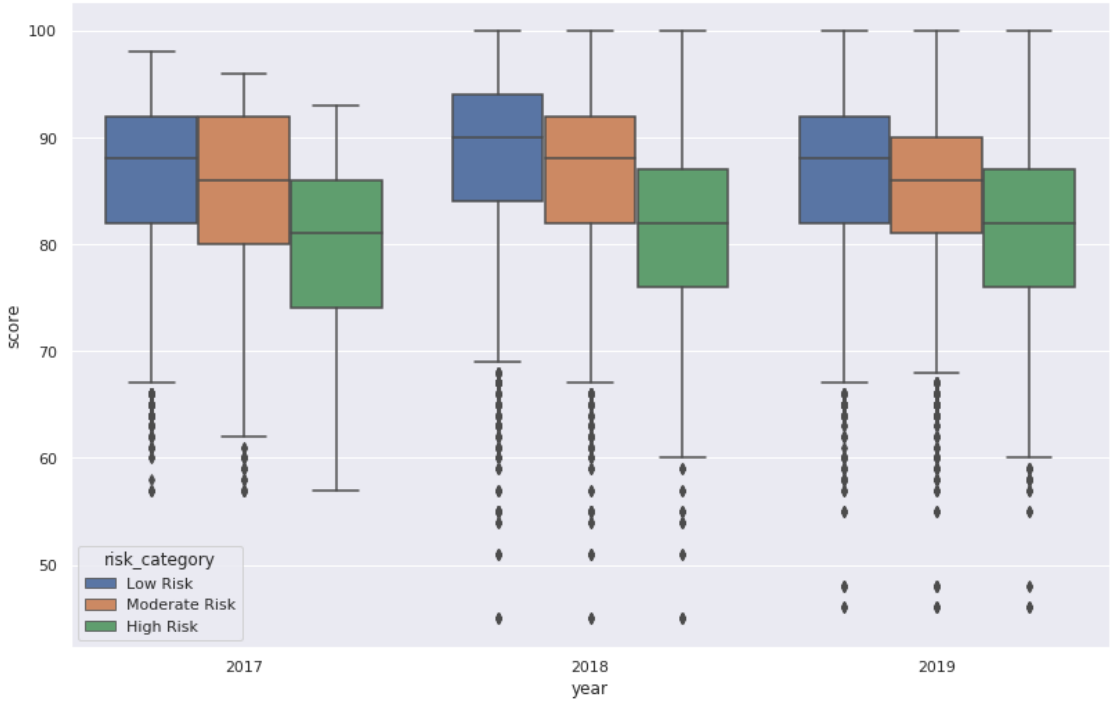


Hint: Use `sns.boxplot()`. Try taking a look at the first several parameters. The documentation is linked here!

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

```
In [445]: ins_vio = pd.merge(vio, ins2vio, how="inner", on="vid")
          ins_named_vio = pd.merge(ins_named, ins_vio, how="inner", on="iid")
          ins_named_vio = ins_named_vio[ins_named_vio["year"] > 2016]
          # Do not modify this line
          sns.set()
          ax = plt.figure(figsize=(12,8))
          ax = sns.boxplot(x="year", y="score", hue="risk_category", data=ins_named_vio, hue_order=["Lo
          ax
```

Out[445]: <matplotlib.axes._subplots.AxesSubplot at 0x7fb701000490>



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 [448]: ins_named_by_bid = ins_named.groupby("bid", as_index=False).agg("size").rename(columns={"size
          ins_named_by_bid = ins_named_by_bid[ins_named_by_bid["count"]==5]
          ins_named_by_bid_and_risk = pd.merge(ins_named_by_bid, ins_named, how="inner", on="bid")
          ins_named_by_bid_and_risk = pd.merge(ins_named_by_bid_and_risk, ins2vio, how="left", on="iid"
          ins_named_by_bid_and_risk = pd.merge(ins_named_by_bid_and_risk, vio, how="left", on="vid")
          ins_named_by_bid_and_risk = ins_named_by_bid_and_risk.drop_duplicates(subset ="iid", keep = "
          ins_named_by_bid_and_risk.head(20)

          # I wanted to see how risk category classification changed across the course of many inspecti
          # the sake of this analysis, I chose restaurants that had had an arbitrary number of five ins
```

```
# just to narrow down the data a bit. For example, it does appear that as the score generally
# For example, the restaurant Moulin Rouge generally improved its score from 81 to 96 over th
# and went from High Risk to generally staying classed a Low Risk.However, this is not always
# next restaurant, Hamburger Haven, whose scores bounced around the 80s and whose classificat
# and Low Risk. I believe that studying the risk categories in conjunction with the inspectio
# can tell an interesting story about its general state and improvement, or lack there of. Co
```

Out[448]:

	bid	count	iid	date	score
0	302	5	302_20170728	07/28/2017 12:00:00 AM	81
5	302	5	302_20180130	01/30/2018 12:00:00 AM	96
7	302	5	302_20180709	07/09/2018 12:00:00 AM	94
10	302	5	302_20190306	03/06/2019 12:00:00 AM	94
12	302	5	302_20190917	09/17/2019 12:00:00 AM	96
14	542	5	542_20161006	10/06/2016 12:00:00 AM	86
18	542	5	542_20170630	06/30/2017 12:00:00 AM	80
22	542	5	542_20180612	06/12/2018 12:00:00 AM	88
27	542	5	542_20190131	01/31/2019 12:00:00 AM	84
32	542	5	542_20190816	08/16/2019 12:00:00 AM	84
38	551	5	551_20161107	11/07/2016 12:00:00 AM	81
43	551	5	551_20170621	06/21/2017 12:00:00 AM	76
49	551	5	551_20180612	06/12/2018 12:00:00 AM	90
53	551	5	551_20190213	02/13/2019 12:00:00 AM	89
56	551	5	551_20190918	09/18/2019 12:00:00 AM	94
58	792	5	792_20161017	10/17/2016 12:00:00 AM	90
61	792	5	792_20170713	07/13/2017 12:00:00 AM	85
64	792	5	792_20180123	01/23/2018 12:00:00 AM	96
66	792	5	792_20181012	10/12/2018 12:00:00 AM	94
69	792	5	792_20190626	06/26/2019 12:00:00 AM	96

	type	timestamp	year	Missing	Score	name
0	Routine - Unscheduled	2017-07-28	2017		False	MOULIN ROUGE
5	Routine - Unscheduled	2018-01-30	2018		False	MOULIN ROUGE
7	Routine - Unscheduled	2018-07-09	2018		False	MOULIN ROUGE
10	Routine - Unscheduled	2019-03-06	2019		False	MOULIN ROUGE
12	Routine - Unscheduled	2019-09-17	2019		False	MOULIN ROUGE
14	Routine - Unscheduled	2016-10-06	2016		False	Hamburger Haven
18	Routine - Unscheduled	2017-06-30	2017		False	Hamburger Haven
22	Routine - Unscheduled	2018-06-12	2018		False	Hamburger Haven
27	Routine - Unscheduled	2019-01-31	2019		False	Hamburger Haven
32	Routine - Unscheduled	2019-08-16	2019		False	Hamburger Haven
38	Routine - Unscheduled	2016-11-07	2016		False	Gordo Taqueria #1
43	Routine - Unscheduled	2017-06-21	2017		False	Gordo Taqueria #1
49	Routine - Unscheduled	2018-06-12	2018		False	Gordo Taqueria #1
53	Routine - Unscheduled	2019-02-13	2019		False	Gordo Taqueria #1
56	Routine - Unscheduled	2019-09-18	2019		False	Gordo Taqueria #1
58	Routine - Unscheduled	2016-10-17	2016		False	Boudin Bakery
61	Routine - Unscheduled	2017-07-13	2017		False	Boudin Bakery
64	Routine - Unscheduled	2018-01-23	2018		False	Boudin Bakery
66	Routine - Unscheduled	2018-10-12	2018		False	Boudin Bakery
69	Routine - Unscheduled	2019-06-26	2019		False	Boudin Bakery

	…	city	state	postal_code	latitude	longitude	phone_number
0	…	San Francisco	CA	94109	37.786084	-122.417889	-9999

```
5   …  San Francisco    CA       94109  37.786084 -122.417889         -9999
7   …  San Francisco    CA       94109  37.786084 -122.417889         -9999
10  …  San Francisco    CA       94109  37.786084 -122.417889         -9999
12  …  San Francisco    CA       94109  37.786084 -122.417889         -9999
14  …  San Francisco    CA       94118  37.782799 -122.467852         -9999
18  …  San Francisco    CA       94118  37.782799 -122.467852         -9999
22  …  San Francisco    CA       94118  37.782799 -122.467852         -9999
27  …  San Francisco    CA       94118  37.782799 -122.467852         -9999
32  …  San Francisco    CA       94118  37.782799 -122.467852         -9999
38  …  San Francisco    CA       94121  37.782107 -122.483631         -9999
43  …  San Francisco    CA       94121  37.782107 -122.483631         -9999
49  …  San Francisco    CA       94121  37.782107 -122.483631         -9999
53  …  San Francisco    CA       94121  37.782107 -122.483631         -9999
56  …  San Francisco    CA       94121  37.782107 -122.483631         -9999
58  …  San Francisco    CA       94133  37.791924 -122.398588         -9999
61  …  San Francisco    CA       94133  37.791924 -122.398588         -9999
64  …  San Francisco    CA       94133  37.791924 -122.398588         -9999
66  …  San Francisco    CA       94133  37.791924 -122.398588         -9999
69  …  San Francisco    CA       94133  37.791924 -122.398588         -9999

    postal5       vid                                      description  \
0     94109  103109.0        Unclean or unsanitary food contact surfaces
5     94109  103156.0     Permit license or inspection report not posted
7     94109  103144.0   Unapproved or unmaintained equipment or utensils
10    94109  103144.0   Unapproved or unmaintained equipment or utensils
12    94109  103149.0  Wiping cloths not clean or properly stored or …
14    94118  103120.0             Moderate risk food holding temperature
18    94118  103144.0   Unapproved or unmaintained equipment or utensils
22    94118  103157.0  Food safety certificate or food handler card n…
27    94118  103131.0                   Moderate risk vermin infestation
32    94118  103149.0  Wiping cloths not clean or properly stored or …
38    94121  103144.0   Unapproved or unmaintained equipment or utensils
43    94121  103103.0                 High risk food holding temperature
49    94121  103144.0   Unapproved or unmaintained equipment or utensils
53    94121  103150.0                     Improper or defective plumbing
56    94121  103144.0   Unapproved or unmaintained equipment or utensils
58    94133  103132.0                           Improper thawing methods
61    94133  103124.0  Inadequately cleaned or sanitized food contact…
64    94133  103161.0                       Low risk vermin infestation
66    94133  103150.0                     Improper or defective plumbing
69    94133  103144.0   Unapproved or unmaintained equipment or utensils

    risk_category
0       High Risk
5        Low Risk
7        Low Risk
10       Low Risk
12       Low Risk
14  Moderate Risk
18       Low Risk
22       Low Risk
27  Moderate Risk
32       Low Risk
38       Low Risk
```

```
43      High Risk
49       Low Risk
53       Low Risk
56       Low Risk
58  Moderate Risk
61  Moderate Risk
64       Low Risk
66       Low Risk
69       Low Risk

[20 rows x 21 columns]
```

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.

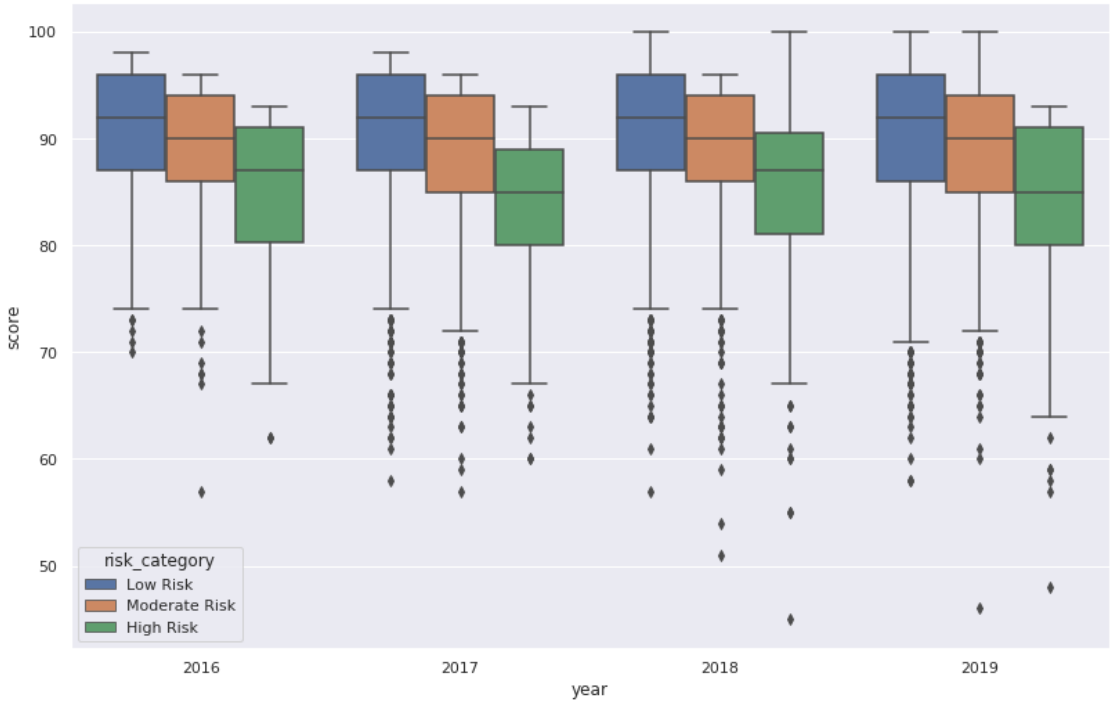
We will lean towards being generous with the grading. We might also either discuss in discussion or post on Piazza some examplar analysis you have done (with your permission)!

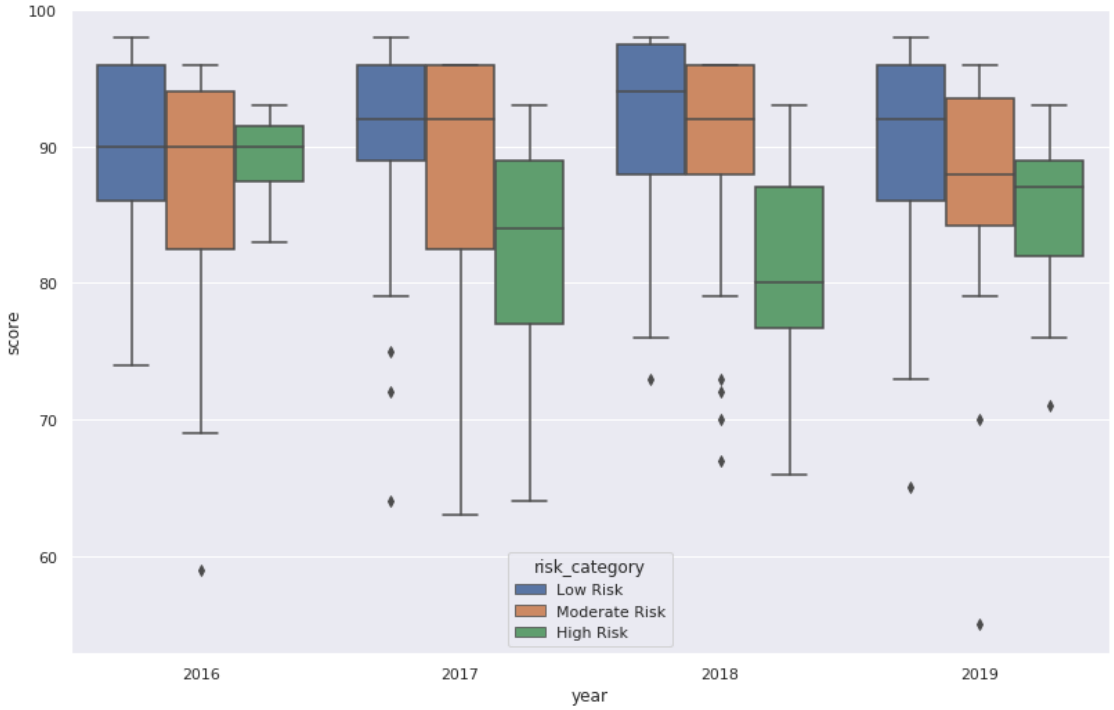
You should have the following in your answers: * a few visualizations; Please limit your visualizations to 5 plots. * a few sentences (not too long please!)

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 [215]:
```
ins_named_by_bid_copy = ins_named.groupby("bid", as_index=False).agg("size").rename(columns={
ins_named_by_bid_copy = ins_named_by_bid_copy[ins_named_by_bid_copy["count"]<5]
ins_named_by_bid_and_risk_copy = pd.merge(ins_named_by_bid_copy, ins_named, how="inner", on="
ins_named_by_bid_and_risk_copy = pd.merge(ins_named_by_bid_and_risk_copy, ins2vio, how="left"
ins_named_by_bid_and_risk_copy = pd.merge(ins_named_by_bid_and_risk_copy, vio, how="left", on=
ins_named_by_bid_and_risk_copy = ins_named_by_bid_and_risk_copy.drop_duplicates(subset ="iid"
bx = plt.figure(figsize=(12,8))
bx = sns.boxplot(x="year", y="score", hue="risk_category", data=ins_named_by_bid_and_risk_cop
bx
cx = plt.figure(figsize=(12,8))
cx = sns.boxplot(x="year", y="score", hue="risk_category", data=ins_named_by_bid_and_risk, hu
cx
# Using an extension of the evaluation of risk category against score from both 7d and 8a, I
# score clusters for restaurants that had been inspected fewer than five times was versus the
# restaurants that had been inspected only five times. Mainly, I wanted to see whether the sc
# change from year to year for the number of inspections. For restaurants inspected fewer tha
# for Low and Moderate Risk tended to stay fairly consistent. For High Risk, the score ranges
# than the ones in 2016 and 2019. Then for restaurants inspected only five times, it changes,
# for the classifications seem to stay consistent over the years. In 2017, for example, the s
# to stretch very far. I cannot be sure if this can be attributed to chance or any inconsiste
# respective decisions for inspections, but either way, it seems to be an odd outlier. This d
# more closely for anything conclusive to come about.
```

Out[215]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fb7085d0f40>
```


100
90
80
70
60
50
score
risk_category
Low Risk
Moderate Risk
High Risk
2016
2017
2018
2019
year


100
90
80
70
60
score
risk_category
Low Risk
Moderate Risk
High Risk
2016
2017
2018
2019
year