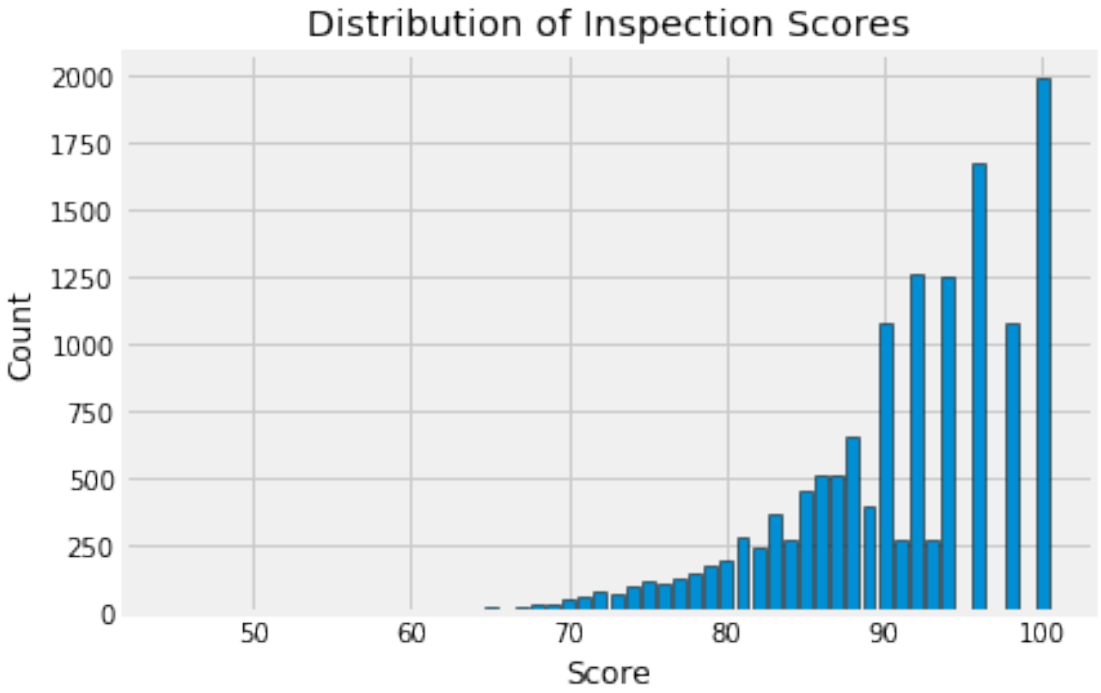
Use the `head` command on your three files again. This time, describe at least one potential problem with the data you see. Consider issues with missing values and bad data.

The latitude and longitude field in the bus.csv have -9999 as values for businesses with no location, which might be problematic when we process our data later.

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.



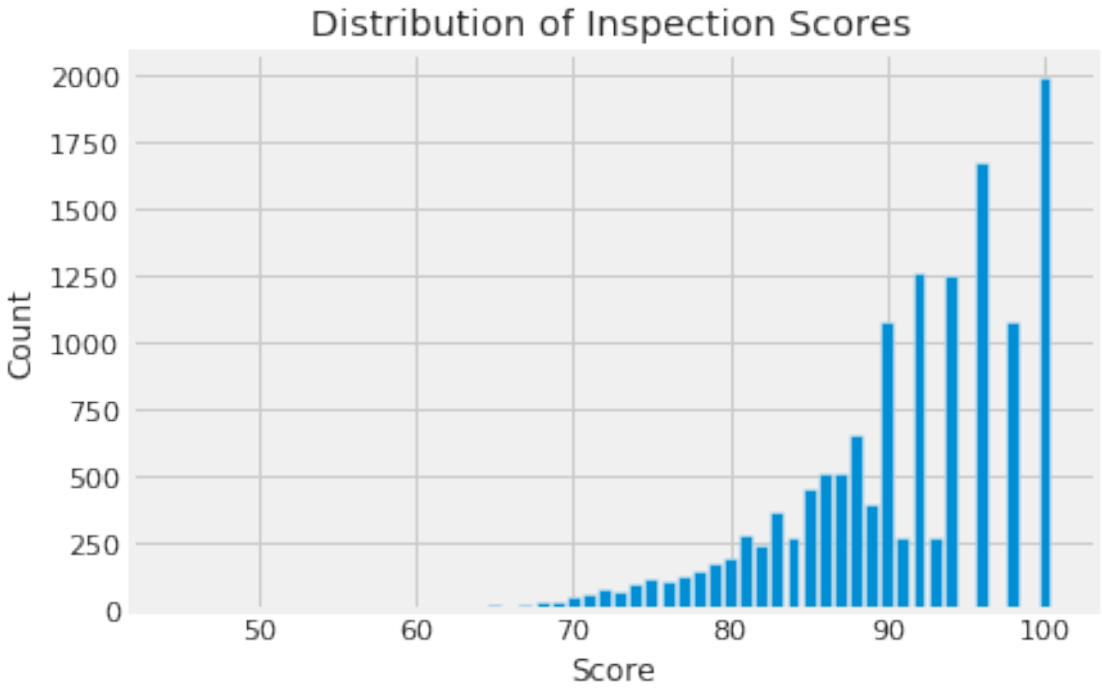
You might find this matplotlib.pyplot tutorial useful. Key syntax that you'll need:

```
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 [74]:
```
graph_series = ins[ins['score'] != - 1]['score'].value_counts().sort_values()
plt.bar(graph_series.index, graph_series.values)
plt.xlabel('Score')
plt.ylabel('Count')
plt.title('Distribution of Inspection Scores')
```

Out[74]:
```
Text(0.5, 1.0, 'Distribution of Inspection Scores')
```



0.1.1 Question 6b

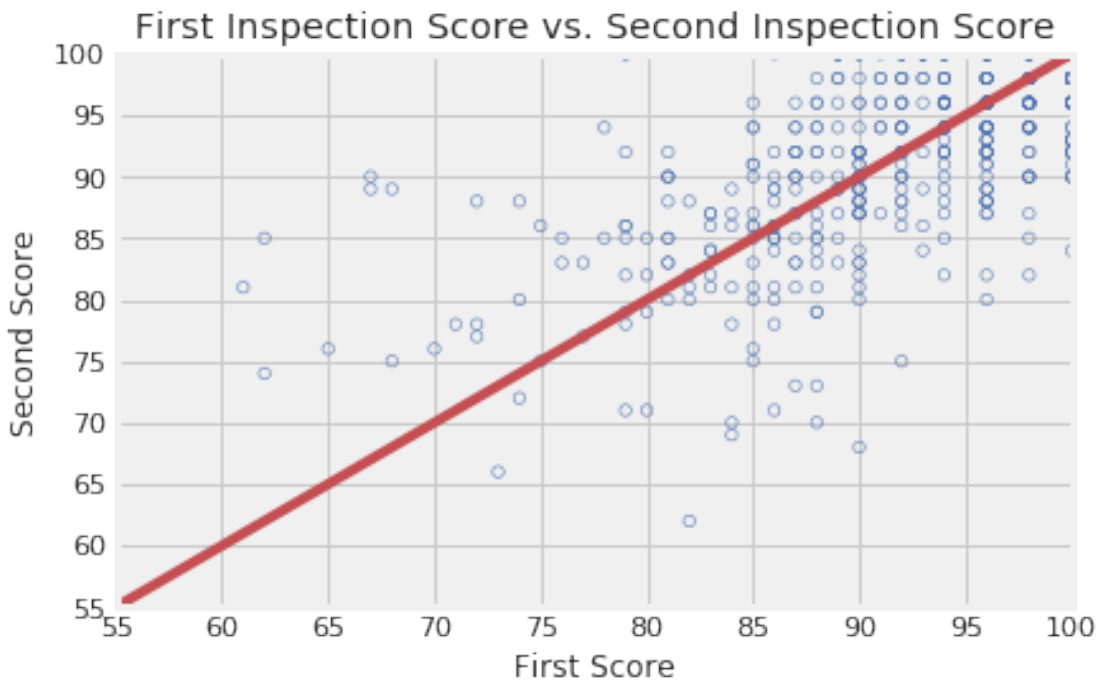
Describe the qualities of the distribution of the inspections scores based on your bar plot. Consider the mode(s), symmetry, tails, gaps, and anomalous values. Are there any unusual features of this distribution? What do your observations imply about the scores?

The mode is a score of 100. The graph is asymmetric and has a tail around $x = 85$, where the graph tapers off towards the left. The distribution has some unusual dips from 90-100 where the count oscillates frequently from high to low. From my observations, the majority of restaurants with a known score had a score from 86-100.

Use the cell above to identify the restaurant with the lowest inspection scores ever. Be sure to include the name of the restaurant as part of your answer in the cell below. 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 worst score (score of 45) is Lollipot.

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.

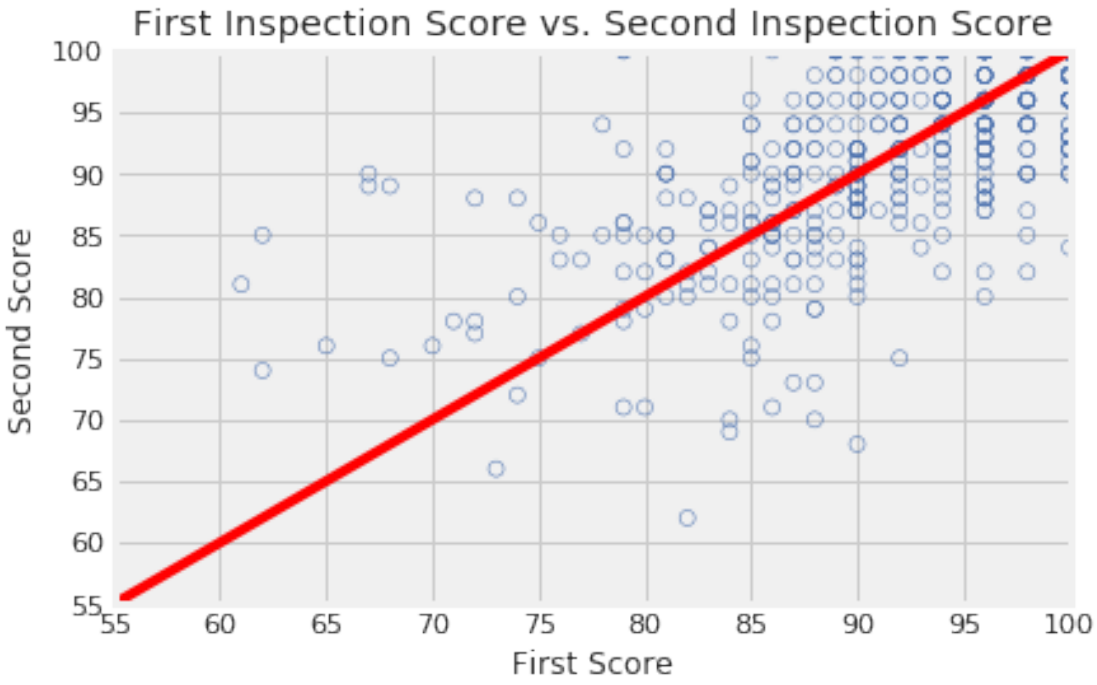
`plt.plot` for the reference line.

`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]: x_points = []
         y_points = []
         for score_pair in scores_pairs_by_business['score_pair']:
             x_points.append(score_pair[0])
             y_points.append(score_pair[1])
         plt.axis(xmin = 55, xmax = 100, ymin = 55, ymax = 100)
         plt.plot([55,100],[55,100],color='red')
         plt.scatter(x_points, y_points, facecolors = 'none', edgecolors = 'b')
         plt.xlabel('First Score')
         plt.ylabel('Second Score')
         plt.title('First Inspection Score vs. Second Inspection Score')
```

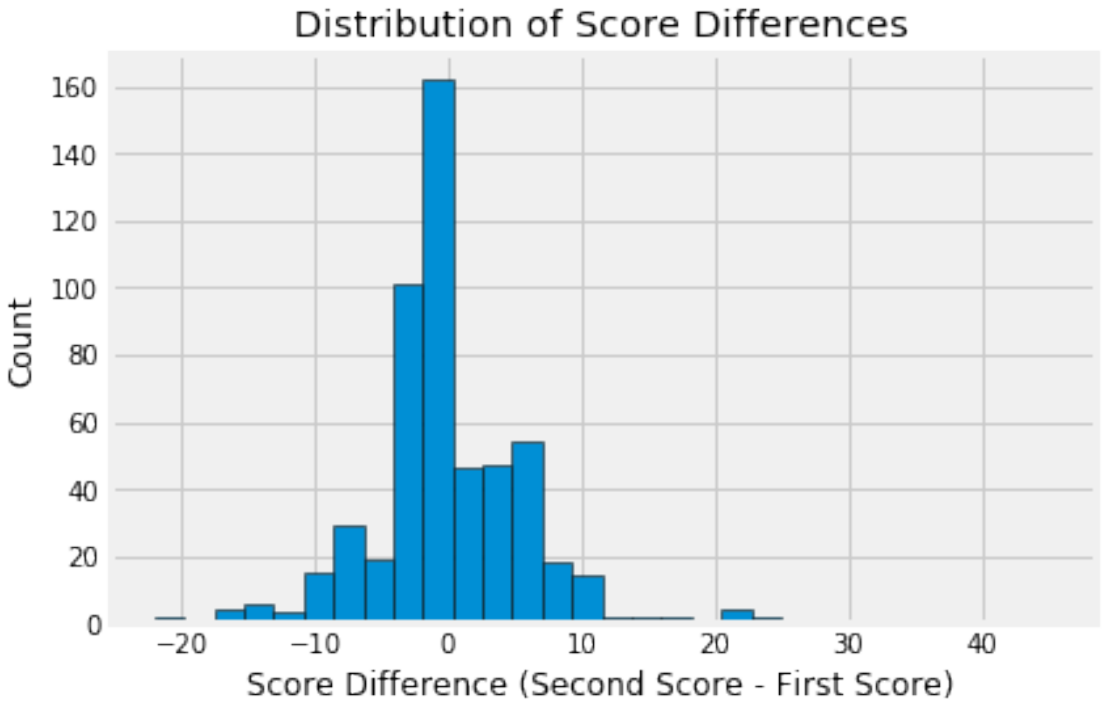
Out[85]: Text(0.5, 1.0, 'First Inspection Score vs. Second Inspection Score')



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.

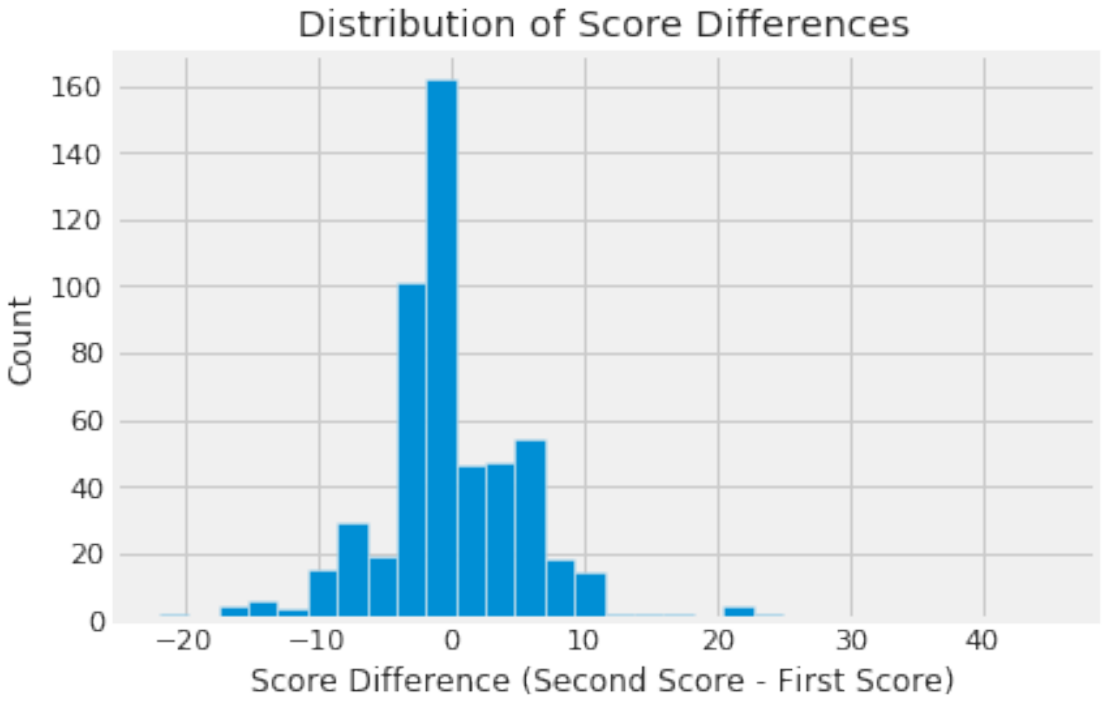
Hint: Convert the scores into numpy arrays to make them easier to deal with.

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

```
In [86]: differences = []
         for score_pair in scores_pairs_by_business['score_pair']:
             differences.append(score_pair[1] - score_pair[0])
         plt.hist(differences, bins = 30)
         plt.xlabel('Score Difference (Second Score - First Score)')
```

```
plt.ylabel('Count')
plt.title('Distribution of Score Differences')
```

Out[86]: 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 2c? What do you oberve from the plot? Are your observations consistent with your expectations?

Hint: What does the slope represent?

If scores tend to improve from the first to second inspection, most of the data points should lie above the reference line since this indicates an improvement from the first score. A lot of data points fall below the reference line, which indicate that some restaurants had scores that went down in the second inspection. My observations are consistent with my expectations as some restaurants had their scores go down from their first inspection.

0.1.4 Question 7f

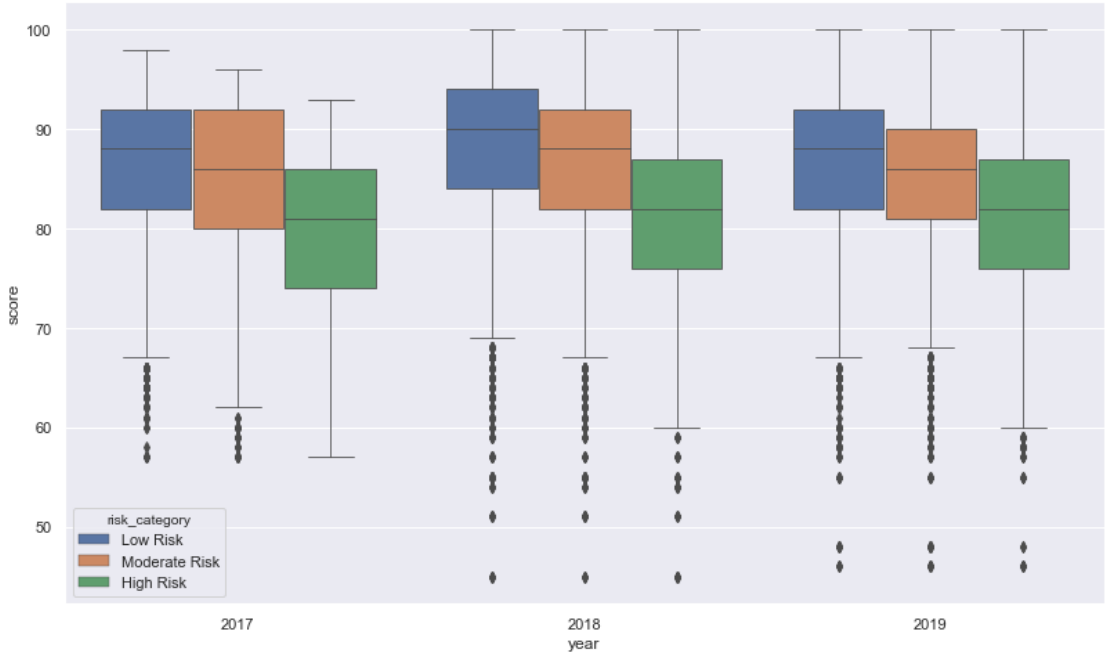
If a restaurant's score improves from the first to the second inspection, how would this be reflected in the histogram of the difference in the scores that you made in question 8d? What do you oberve from the plot? Are your observations consistent with your expectations? Explain your observations in the language of Statistics: for instance, the center, the spread, the deviation etc.

If a restaurant's score improves from the first to second inspection, this would result in higher frequencies of positive values in the histogram or a center that is a positive value. From observing the plot, it is clear that center of the distribution is at a negative value, indicating many restaurants had their score go down from the first inspection. My observations are consistent with my expectations of how the data will be represented in the histogram.

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!



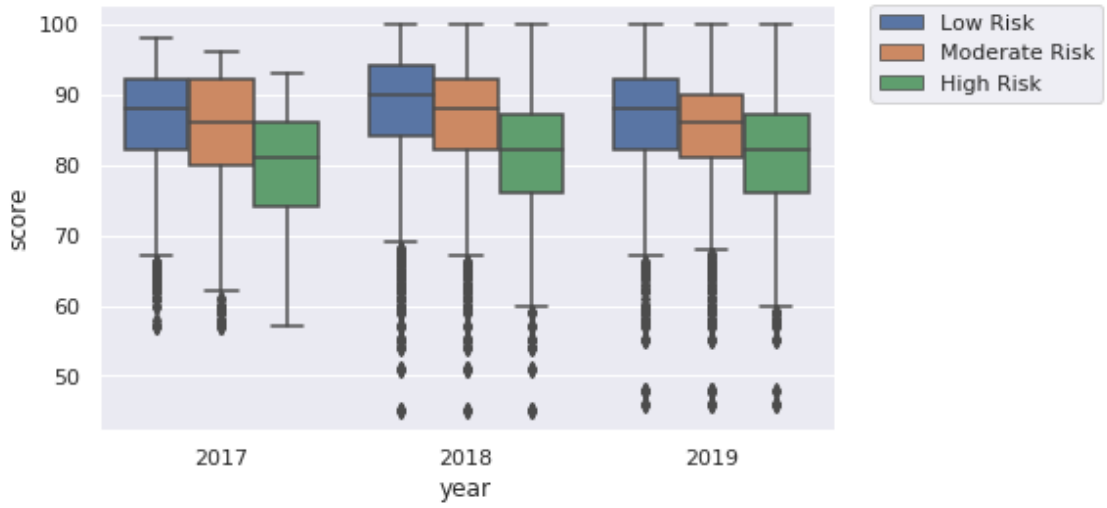
Hint: Use `sns.boxplot()`. Try taking a look at the first several parameters. The documentation is linked here!

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

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

valid_ins = score_count_table[(score_count_table['score'] != -1) & (score_count_table['year']
sns.boxplot(valid_ins['year'], valid_ins['score'], hue=valid_ins['risk_category'], hue_order=[
plt.legend(bbox_to_anchor=(1.05, 1), loc=2, borderaxespad=0.)
```

Out[87]:
```
<matplotlib.legend.Legend at 0x7f569ea24290>
```


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

---

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.

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 [95]:
```
#YOUR CODE HERE
no_number = bus[bus['address'] == 'Off The Grid']
ins_info8 = ins_named.drop(columns=['iid','name','address'])
no_number = pd.merge(no_number, ins_info8, on = ['bid'])
no_number = no_number[no_number['score'] != -1]

print('Off grid mean:', no_number['score'].mean())
print('Valid address mean:', ins_named['score'].mean())

#YOUR EXPLANATION HERE (in a comment)
#Restaurants and businesses off the grid (i.e. food trucks, night markets) had an average scor
#Restaurants and businesses with valid addresses had an average score of 47.23. So, overall, r
#tend to have better inspection scores than restaurants with fixed locations.
```

```
Off grid mean: 92.16981132075472
Valid address mean: 47.22769380789859
```

1.1.2 Grading

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

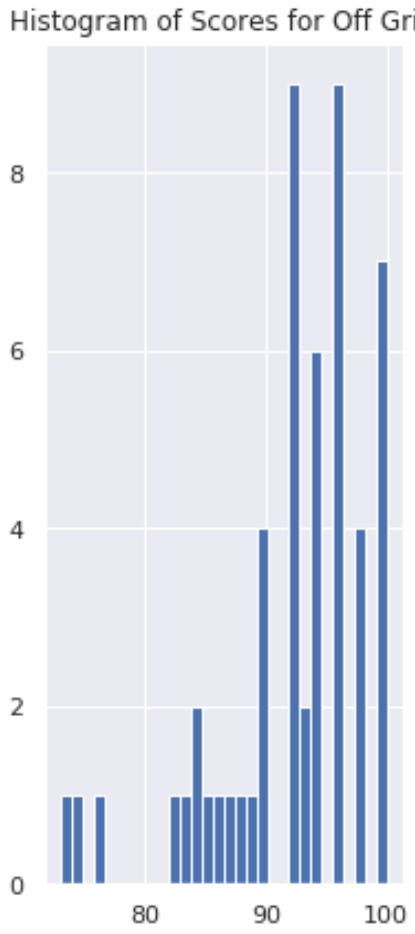
- **Great** (4 points): The chart is well designed, and the data computation is correct. The text written articulates a reasonable metric and correctly describes the relevant insight and answer to the question you are interested in.
- **Passing** (1-3 points): A chart is produced but with some flaws 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.

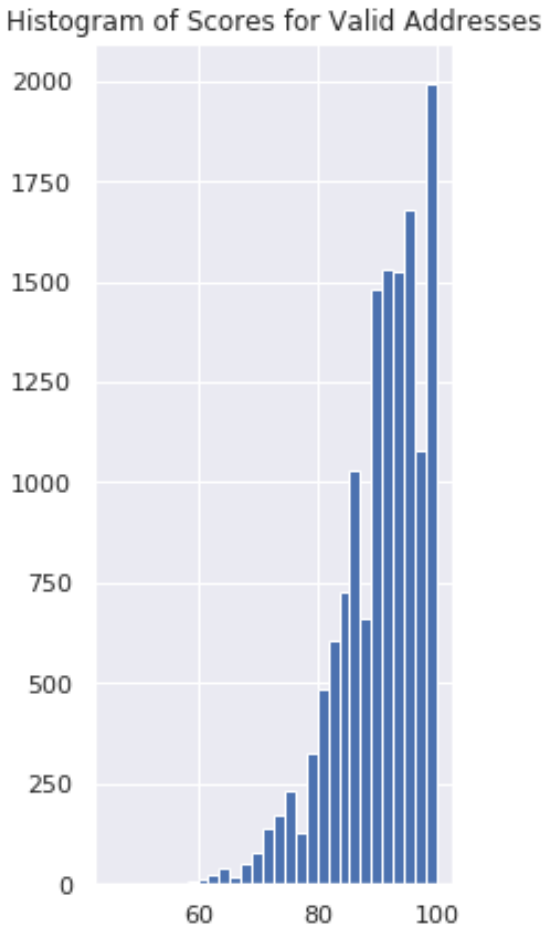
We will lean towards being generous with the grading. We might also either discuss in discussion or post on Piazza some examplar analysis you have done (with your permission)!

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 [125]:
```
# YOUR DATA PROCESSING AND PLOTTING HERE
fig, axs = plt.subplots(1,2,figsize=(7,7))
series1 = no_number['score'].sort_values()
ins_pos = ins_named[ins_named['score'] != -1]
series2 = ins_pos['score'].sort_values()
series1.hist(bins =30, ax=axs[0])
series2.hist(bins =30, ax=axs[1])
axs[0].set_title('Histogram of Scores for Off Grid')
axs[1].set_title('Histogram of Scores for Valid Addresses')
fig.subplots_adjust(wspace=0.7)
# YOUR EXPLANATION HERE (in a comment)
# The question we attempt to answer is do off grid restaurants perform significantly better t
# in inspections?
# The histogram is able to provide us insight as to why the average of off grid restaurants a
# that the sample size of off grid restaurants are much smaller, we are not able to obtain an
# of the entire population, so our results may be skewed. From the given data and visualizati
# that off grid restaurants do not necessarily perform better on average than their counterpa
# may not contain an adequate number of sample points.
```


Histogram of Scores for Off Grid
0
2
4
6
8
80
90
100


Histogram of Scores for Valid Addresses
0
250
500
750
1000
1250
1500
1750
2000
60
80
100