Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

1 Seattle Section 8 Housing Proj

2 Introduction

2.1 Premise

Public/low income housing has become a somewhat controversia
While many can agree there are two Seattles, or about the existe
"homelessness issue," ideas about how to solve for this problem
for real estate investors, increased police budgets, to "defensive
buildings to stop people from sleeping there) to increasing unemp
back-to-work programs. In the city's political history, candidates h
class disparity and homelessness either larger or smaller, depen
intended audience. In 2015, when this dataset is from, 45 homele
direct result of their homelessness, and the mayor declared a sta
(CITATION: https://www.capitolhillseattle.com/2015/11/45-dead-i
declares-seattle-homelessness-in-a-state-of-emergency/
(https://www.capitolhillseattle.com/2015/11/45-dead-in-2015-may
homelessness-in-a-state-of-emergency/))

Hypothetical Premise: The City of Seattle city government, in par
of Washington's state senate has a new pilot program called Sec
There is a 100 million dollar budget in this pilot program to buy re
value to convert into section 8 housing. This project has 3 main g
the availability of low-income housing 2) to increase property valu
3) to decrease class disparity.

Instead of making parts of the city into the projects, this program
public housing all over the city. This evidence based practice resu
for both the tenants and the properties. Socially, it reduces stigma

Contents

- 6.1 Comparing Renovated Homes with Unrenovated
- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

well.

Each property bought for this project will undergo a renovation to living, and there is a separate budget set aside specifically for rer project, I'm not responsible for balancing the cost of renovating th

I am instead tasked with using machine learning to build a model recommend where to build more public housing, under the cavea renovated. I am looking for properties that will have increased pro future. Certain considerations will need to be taken into account f recommendation, such as the spacing these properties.

While other cities such as Dallas, Texas and Gary Indiana have b from the public with success, this would be on a much larger scal https://www.fastcompany.com/90618596/the-radical-way-cities-are housing (https://www.fastcompany.com/90618596/the-radical-way affordable-housing))

(CITATION FOR OVERVIEW ON SECTION 8 HOUSING: https://www.hud.gov/program_offices/housing/mfh/rfp/s8bkinfo (https://www.hud.gov/program_offices/housing/mfh/rfp/s8bkinfo))

FURTHER READING: On mixed income housing in NYC, where https://case.edu/socialwork/nimc/sites/case.edu.nimc/files/2020-05/Schwartz%20NewYork%202020.pdf (https://case.edu/socialwork/nimc/sites/case.edu.nimc/files/2020-05/Schwartz%20NewYork%202020.pdf)

Why should low-income housing be interspersed throughout (CITATION)

"Proponents of mixed-income housing see it as a tool to address to what has been termed the *culture of poverty*. This phrase deriv physical concentration of poor households in multifamily projects problems for the residents, including joblessness, drug abuse, ar

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

According to Brophy and Smith 6 Cityscape this theory, a mixture reduce the social pathology caused by concentration. Anticipatec income housing include the following:

- The behavior patterns of some lower income residents will be a those of their higher income neighbors. The quality of the living e housing quality alone, leads to upward mobility.

- Nonworking low-income tenants will find their way into the work numbers because of the social norms of their new environment (f work/school every day) and the informal networking with employe

- The crime rate will fall because the higher income households v and better enforced set of ground rules for the community.

- Low-income households will have the benefit of better schools, enhanced safety, enabling them to move themselves and their ch current economic condition.

Many of these anticipated results are subtle and difficult to measu quantifiable result as the effect of mixed incomes on the project's analysis of the effects of mixed-income housing on the behavior c into account the subtleties of human behavior. It is also important between two reasons for these intended benefits. First, if low-inco subsidized to live in developments that are in locations with good rates, and access to jobs, there is some likelihood that the benefi result in the anticipated outcomes described above."

https://www.huduser.gov/periodicals/cityscpe/vol3num2/success.pc (https://www.huduser.gov/periodicals/cityscpe/vol3num2/success.p

2.2 The Stakeholder's Interests:

The city of Seattle has tasked me with ensuring their making goo with an assured outcome. Since this project is using public funds documented (and discussed in media.) The model, furthermore, r

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

understandable to civic workers.

The city is also especially interested in seeing the range of housi
Often in statistics, we focus on the median or mean, however for
to be collecting data on the size of the ranges.

2.3 Why Use Machine Learning?

It is coloquial knowledge that there is class disparity in Seattle. H
gap is between neighborhoods specifically, home by home, prope
I've been tasked to research. Seattle has decided that the best a
one backed by research, science and evidence.

Using machine learning, we can rely on the data itself to ensure
evidence-backed solutions to this problem.

Machine Learning is being implemented here to reduce the politi
solution. However, ML can only be as free of bias as the impleme

2.3.1 Why I Was Highered to Oversee the Projec

In order to counter the mechanisms were ML can propogate racis
(CITATION: https://www.nytimes.com/2021/03/15/technology/arti
google-bias.html (https://www.nytimes.com/2021/03/15/technolog
google-bias.html)) Seattle chose me specifically as a Data Scient
experience working with homeless folks, at a housing-forward no

3 Loading and Inspecting the Da

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [1]:

```
# importing the libraries to use, the basics, the libraries
# geopy for distance data collection and selenium for scrap

import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

import statsmodels.api as sm
from statsmodels.stats.outliers_influence import variance_i
import scipy.stats as stats
from sklearn.preprocessing import StandardScaler

## geopy must be installed first
from geopy import distance

## for scraping distances
from selenium import webdriver
from webdriver_manager.chrome import ChromeDriverManager
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Keys

import requests, json

import folium

%matplotlib inline

palette = sns.color_palette("colorblind")
sns.set_style('white')
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [2]:

```
# loading the main CSV

data = pd.read_csv('Data/kc_house_data.csv')
```

In [3]:

```
# inspecting the data

data.head()
```

	id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot
0	7129300520	10/13/2014	221900.0	3	1.00	1180	5650
1	6414100192	12/9/2014	538000.0	3	2.25	2570	7242
2	5631500400	2/25/2015	180000.0	2	1.00	770	10000
3	2487200875	12/9/2014	604000.0	4	3.00	1960	5000
4	1954400510	2/18/2015	510000.0	3	2.00	1680	8080

5 rows × 21 columns

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [4]:

```
data.info()

# taking a look at the raw data
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 21597 entries, 0 to 21596
Data columns (total 21 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   id             21597 non-null  int64
 1   date           21597 non-null  object
 2   price          21597 non-null  float64
 3   bedrooms       21597 non-null  int64
 4   bathrooms      21597 non-null  float64
 5   sqft_living    21597 non-null  int64
 6   sqft_lot       21597 non-null  int64
 7   floors         21597 non-null  float64
 8   waterfront     19221 non-null  object
 9   view           21534 non-null  object
 10  condition      21597 non-null  object
 11  grade          21597 non-null  object
 12  sqft_above     21597 non-null  int64
 13  sqft_basement  21597 non-null  object
 14  yr_built       21597 non-null  int64
 15  yr_renovated   17755 non-null  float64
 16  zipcode        21597 non-null  int64
 17  lat            21597 non-null  float64
 18  long           21597 non-null  float64
 19  sqft_living15  21597 non-null  int64
 20  sqft_lot15     21597 non-null  int64
dtypes: float64(6), int64(9), object(6)
memory usage: 3.5+ MB
```

Of note: waterfront, and year renovated have significant data mis
missing only a bit of data and those null values can probably be f
makes the most sense given the context.

Contents

6.1 Comparing Renovated Homes with Unrenovated
6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [5]:
```
data.isna().sum()
```

```
id                 0
date               0
price              0
bedrooms           0
bathrooms          0
sqft_living        0
sqft_lot           0
floors             0
waterfront      2376
view              63
condition          0
grade              0
sqft_above         0
sqft_basement      0
yr_built           0
yr_renovated    3842
zipcode            0
lat                0
long               0
sqft_living15      0
sqft_lot15         0
dtype: int64
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [6]:

```
data.describe()
```

	id	price	bedrooms	bathrooms	sqft_living	
count	2.159700e+04	2.159700e+04	21597.000000	21597.000000	21597.000000	2.15
mean	4.580474e+09	5.402966e+05	3.373200	2.115826	2080.321850	1.50
std	2.876736e+09	3.673681e+05	0.926299	0.768984	918.106125	4.14
min	1.000102e+06	7.800000e+04	1.000000	0.500000	370.000000	5.20
25%	2.123049e+09	3.220000e+05	3.000000	1.750000	1430.000000	5.04
50%	3.904930e+09	4.500000e+05	3.000000	2.250000	1910.000000	7.61
75%	7.308900e+09	6.450000e+05	4.000000	2.500000	2550.000000	1.06
max	9.900000e+09	7.700000e+06	33.000000	8.000000	13540.000000	1.65

Of note--the average house has 3 bedrooms, 2 bathrooms, whic
we are already looking to make public. The average house was b
the average house on the market 52 years old. The average renc
years ago, which is older than the standard property is old, which
information on renovations must be explored in greater detail.

In [7]:

```
data.columns

# looking to see what columns are in the dataset
```

```
Index(['id', 'date', 'price', 'bedrooms', 'bathrooms', 'sqft_living',
       'sqft_lot', 'floors', 'waterfront', 'view', 'condition', 'grade',
       'sqft_above', 'sqft_basement', 'yr_built', 'yr_renovated', 'zipcode
       'lat', 'long', 'sqft_living15', 'sqft_lot15'],
      dtype='object')
```

Contents

- 6.1 Comparing Renovated Homes with Unrenovated
- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [8]:

```
sns.set_style('white')

data.hist(figsize=(20,20));

# let's look at the distributions of some of these columns
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

4 Cleaning and Preparing the Da

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

4.0.1 Duplicates

In [9]:

```
data['id'].value_counts().head()
# let's take a look to see if there are duplicates
```

```
795000620     3
1825069031    2
2019200220    2
7129304540    2
1781500435    2
Name: id, dtype: int64
```

Some of these properties are listed three times, that will throw off
remove the duplicates.

In [10]:

```
data.drop_duplicates(subset='id', inplace=True)

# this will get rid of duplicates
```

4.0.2 Price

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [11]:

```
sns.distplot( a=data["price"]);

# this will show a histogram for the price distribution
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
y:2551: FutureWarning: `distplot` is a deprecated function and will be rem
n. Please adapt your code to use either `displot` (a figure-level function
y) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Observations: The distribution of price in this dataset has a very l
expensive houses, and the data is leptokurtic (taller and skinnier)
distribution.

Since this project is specifically about buying a lot of properties, I
on properties that are priced at 2.5 million and below, which is wh
data is. This will prevent all the calculations from being dragged i
price brackets for the city, and for the tenants of the new housing

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [12]:

```
data = data.loc[data['price'] < 2500000.0]

# this will get rid of all properties in the data set above
```

In [13]:

```
sns.distplot( a=data["price"])

# this will show what the distribution of prices looks like
# that remains
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
y:2551: FutureWarning: `distplot` is a deprecated function and will be rem
n. Please adapt your code to use either `displot` (a figure-level function
y) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

```
<AxesSubplot:xlabel='price', ylabel='Density'>
```

4.0.3 Sqft_living

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

Square footage of living space in the home-from the data.

In [14]:

```
sns.distplot( a=data["sqft_living"]);

# Taking a look at the distribution of this variable
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
y:2551: FutureWarning: `distplot` is a deprecated function and will be rem
n. Please adapt your code to use either `displot` (a figure-level function
y) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Contents

6.1 Comparing Renovated Homes with Unrenovated
6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [15]:

```
sns.regplot(data = data, x = 'price', y = 'sqft_living',
            line_kws={"color":"r","alpha":0.3,"lw":5})
plt.show()

# Visualizing how sqft_living affects price
```

I'm going to get rid of the outliers in the data set, the homes we're public housing will not be these outliers.

In [16]:

```
data = data.loc[data['sqft_living'] < 8000]
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [17]:

```
sns.distplot( a=data["sqft_living"]);
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
y:2551: FutureWarning: `distplot` is a deprecated function and will be rem
n. Please adapt your code to use either `displot` (a figure-level function
y) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [18]:

```
sns.regplot(data = data, x = 'price', y = 'sqft_living',
            line_kws={"color":"r","alpha":0.3,"lw":5})
plt.show()
```

4.0.4 Sqft_above

Square footage of house apart from basement-from the data.

Contents

- 6.1 Comparing Renovated Homes with Unrenovated
- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [19]:

```
sns.distplot( a=data["sqft_above"]);
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
y:2551: FutureWarning: `distplot` is a deprecated function and will be rem
n. Please adapt your code to use either `displot` (a figure-level function
y) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [20]:

```
sns.regplot(data = data, x = 'price', y = 'sqft_above',
            line_kws={"color":"r","alpha":0.3,"lw":5})
plt.show()
```

4.0.5 Sqft_lot

Square footage of the lot-from the data.

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [21]:

```
sns.distplot( a=data["sqft_lot"]);
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
y:2551: FutureWarning: `distplot` is a deprecated function and will be rem
n. Please adapt your code to use either `displot` (a figure-level function
y) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [22]:

```
sns.regplot(data = data, x = 'price', y = 'sqft_lot',
            line_kws={"color":"r","alpha":0.3,"lw":5})
plt.show()
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [23]:

```
data['sqft_lot'].describe()
```

```
count    2.131700e+04
mean     1.508035e+04
std      4.156528e+04
min      5.200000e+02
25%      5.030000e+03
50%      7.590000e+03
75%      1.060000e+04
max      1.651359e+06
Name: sqft_lot, dtype: float64
```

In [24]:

```
data = data.loc[data['sqft_lot'] < 100000]
```

In [25]:

```
sns.distplot( a=data["sqft_lot"]);
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
y:2551: FutureWarning: `distplot` is a deprecated function and will be rem
n. Please adapt your code to use either `displot` (a figure-level function
y) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [26]:

```
sns.regplot(data = data, x = 'price', y = 'sqft_lot',
            line_kws={"color":"r","alpha":0.3,"lw":5})
plt.show()
```

4.0.6 Sqft_basement

In [27]:

```
data['sqft_basement'].value_counts().head()
```

```
0.0      12374
?          437
600.0      214
700.0      204
500.0      204
Name: sqft_basement, dtype: int64
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [28]:

```
data['sqft_basement'].describe()
```

```
count     20855
unique      288
top         0.0
freq      12374
Name: sqft_basement, dtype: object
```

In [29]:

```
data.loc[data['sqft_basement'] == '?','sqft_basement'] = 0.0

#? doesn't make sense as a value. I will replace it with th
```

Contents

6.1 Comparing Renovated Homes with Unrenovated
6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [30]:

```

data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 20855 entries, 0 to 21596
Data columns (total 21 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   id             20855 non-null  int64  
 1   date           20855 non-null  object 
 2   price          20855 non-null  float64
 3   bedrooms       20855 non-null  int64  
 4   bathrooms      20855 non-null  float64
 5   sqft_living    20855 non-null  int64  
 6   sqft_lot       20855 non-null  int64  
 7   floors         20855 non-null  float64
 8   waterfront     18559 non-null  object 
 9   view           20795 non-null  object 
 10  condition      20855 non-null  object 
 11  grade          20855 non-null  object 
 12  sqft_above     20855 non-null  int64  
 13  sqft_basement  20855 non-null  object 
 14  yr_built       20855 non-null  int64  
 15  yr_renovated   17129 non-null  float64
 16  zipcode        20855 non-null  int64  
 17  lat            20855 non-null  float64
 18  long           20855 non-null  float64
 19  sqft_living15  20855 non-null  int64  
 20  sqft_lot15     20855 non-null  int64  
dtypes: float64(6), int64(9), object(6)
memory usage: 3.5+ MB
```

In [31]:

```
data['sqft_basement'] = data['sqft_basement'].astype(float)
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [32]:

```
sns.regplot(data = data, x = 'price', y = 'sqft_basement',
            line_kws={"color":"r","alpha":0.3,"lw":5})
plt.show()
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [33]:

```
data['sqft_basement'].hist()
```

```
<AxesSubplot:>
```

I'm going to turn this into a binary variable: 0 if the property has n
(sqft_basement at 0,) and 1 if the property has a basement. Ther
priced homes without a basement, and over 50% of homes don't

In [34]:

```
data.loc[data['sqft_basement'] != 0,'sqft_basement'] = 1
```

In [35]:

```
data['sqft_basement'].value_counts().head()
```

```
0.0    12811
1.0     8044
Name: sqft_basement, dtype: int64
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [36]:

```
sns.boxplot(data = data, x="sqft_basement", y="price", pale
```

```
<AxesSubplot:xlabel='sqft_basement', ylabel='price'>
```

Observations: the minimum quartile price, average price and ma
are all higher in properties with basements, as to be expected.

Square Foot Living 15 and Square Lot 15

It is noteworthy to have the square footage of the living space an
15 neighbors. These following two metrics are especially interest
Traditional public housing, in big blocky complexes, would have r
the following two variables than in mixed income public housing r

FURTHER WORK:

None of the properties in this data set are public housing properti
compare how the relationships to the closest 15 neighboring prop
both public and non-public housing, I'd need to collect more data

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

4.0.7 Sqft_living15

The square footage of interior housing living space for the neares
the data.

In [37]:

```
sns.distplot( a=data["sqft_living15"]);
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
y:2551: FutureWarning: `distplot` is a deprecated function and will be rem
n. Please adapt your code to use either `displot` (a figure-level function
y) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
  - 7.1 Getting Infrastucture Data
    - 7.1.1 Downtown
    - 7.1.2 Hospitals
    - 7.1.3 Police Stations
    - 7.1.4 Schools
  - 7.2 Finding Distance for Infrastructure
  - 7.3 Exploring Distance Data
    - 7.3.1 Downtown
    - 7.3.2 Hospitals
    - 7.3.3 Police Stations
    - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [38]:

```
sns.regplot(data = data, x = 'price', y = 'sqft_living15',
            line_kws={"color":"r","alpha":0.3,"lw":5})
plt.show()
```

4.0.8 Sqft_lot15

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [39]:

```
sns.regplot(data = data, x = 'price', y = 'sqft_lot15',
            line_kws={"color":"r","alpha":0.3,"lw":5})
plt.show()
```

I'm going to get rid of the upper valued outliers here, which woulc
where the 15 closest neighbors have a *lot* of square footage. The
neighborhoods and will throw off the data. I want to be conservat

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [40]:

```
data = data.loc[data['sqft_lot15'] < 400000]
```

In [41]:

```
sns.regplot(data = data, x = 'price', y = 'sqft_lot15',
            line_kws={"color":"r","alpha":0.3,"lw":5})
plt.show()
```

4.0.9 Waterfront

In [42]:

```
data['waterfront'].value_counts()
```

```
NO     18447
YES      112
Name: waterfront, dtype: int64
```

In [43]:

```
data['waterfront'].isna().sum()
```

```
2296
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [44]:

```
data['waterfront'].fillna(value = "NO", inplace = True)
```

In [45]:

```
data['waterfront'].replace(['NO', 'YES'],
                           [0, 1], inplace=True)
#this will make the value numeric and therefore able to be
#the model
```

Because the average value is "no", I'm choosing to fill null values
waterfront view would be a boon to the property, the risk involved
representing a property with a waterfront to have no waterfront, a
selected for public housing has a surprise positive effect for the fo
there. By contrast, if I filled these values with "yes" and they didn
this would falsely inflate the assigned value of that property, and
were selected because of that false inflated value, that would act
folks who lived there.

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [46]:

```
sns.boxplot(data = data, x = 'waterfront', y = 'price', pal
```

```
<AxesSubplot:xlabel='waterfront', ylabel='price'>
```

This is a clear visualization that represents:

1) homes with a waterfront have a much wider spread for their av in the upper limits of how pricey those homes can be. But in gene are more expensive. 2) there are plenty outlying examples of hor that are more expensive.

4.0.10 Bedrooms

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [47]:

```
data['bedrooms'].hist()
# taking a look at the distribution of bedrooms in this dat
```

```
<AxesSubplot:>
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [48]:

```
data['bedrooms'].value_counts()
```

```
3     9512
4     6649
2     2685
5     1517
6      248
1      186
7       35
8       12
9        6
10       3
11       1
33       1
Name: bedrooms, dtype: int64
```

Since properties where there are more than 9 bedrooms may be public housing anyway, and because most of the data is outside t dropping properties that have more than 9 bedrooms.

In [49]:

```
data = data.loc[data['bedrooms'] <= 9]
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [50]:

```
sns.boxplot(data = data, x="bedrooms", y="price", palette =
# we can see from the figure that as bedrooms increase, pri
# increases, and also we can get a better sense for the dis
# outliers by bedroom.
```

```
<AxesSubplot:xlabel='bedrooms', ylabel='price'>
```

This plot shows 1) that homes show a steady increase in minimu
increase 2) that there are very many outliers in homes that have
but even with those outliers, there is also a steady increase in ma
bedrooms increase as well.

5 Transforming Categorical Data

5.0.1 Condition

CITATION: https://info.kingcounty.gov/assessor/esales/Glossary.asp

Contents

- 6.1 Comparing Renovated Homes with Unrenovated
- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

(https://info.kingcounty.gov/assessor/esales/Glossary.aspx?type=r#c

Relative to age and grade. Coded 1-5.

1 = Poor- Worn out. Repair and overhaul needed on painted surf plumbing, heating and numerous functional inadequacies. Exces maintenance and abuse, limited value-in-use, approaching aban reconstruction; reuse or change in occupancy is imminent. Effect of the scale regardless of the actual chronological age.

2 = Fair- Badly worn. Much repair needed. Many items need refir deferred maintenance obvious, inadequate building utility and sy the life expectancy and increasing the effective age.

3 = Average- Some evidence of deferred maintenance and norm age in that a few minor repairs are needed, along with some refir components still functional and contributing toward an extended l Effective age and utility is standard for like properties of its class

4 = Good- No obvious maintenance required but neither is everyt and utility are above the standard and the overall effective age w typical property.

5 = Very Good- All items well maintained, many having been ove as they have shown signs of wear, increasing the life expectancy effective age with little deterioration or obsolescence evident with utility.

Based on the description as given by the data set, this is a good into continuous data. A house in "very good" condition could get in the future if the property doesn't receive proper maintenance. l "fair" condition could be renovated and made into a property with

Based on the description as given by the data set, this is a good into continuous data. A house in "very good" condition could get in the future if the property doesn't receive proper maintenance. l "fair" condition could be renovated and made into a property with

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [51]:

```
data['condition'].replace(['Poor', 'Fair', 'Average', 'Good
                          [0, 1, 2, 3, 4], inplace=True)

# making these values continuous
```

In [52]:

```
data['condition'].value_counts()

# looking at the spread of the data
```

```
2    13519
3     5502
4     1653
1      149
0       27
Name: condition, dtype: int64
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [53]:

```
sns.distplot( a=data["condition"], hist=True, kde=False, ru

# visualizing the spread of the data
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
y:2551: FutureWarning: `distplot` is a deprecated function and will be rem
n. Please adapt your code to use either `displot` (a figure-level function
y) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

```
<AxesSubplot:xlabel='condition'>
```

Observations: there is relatively no representation of homes with "fair" in this dataset. A majority of homes are in fair condition, witl condition and even less in "very good" condition.

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fc
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Loc
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [54]:

```
sns.boxplot(data = data, x = 'condition', y = 'price', pale
plt.show();
```

Observations: The price of homes in poor condition and fair cond
a lower price than homes in better condition. The minimum price
it's condition increases in value, as well as the upper limit price fc

5.0.2 Grade

CITATION: https://info.kingcounty.gov/assessor/esales/Glossary.asp
(https://info.kingcounty.gov/assessor/esales/Glossary.aspx?type=r#c

1-3 Falls short of minimum building standards. Normally cabin or

4 Generally older, low quality construction. Does not meet code.

5 Low construction costs and workmanship. Small, simple desigr

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

6 Lowest grade currently meeting building code. Low quality mat
designs.

7 Average grade of construction and design. Commonly seen in
divisions.

8 Just above average in construction and design. Usually better
exterior and interior finish work.

9 Better architectural design with extra interior and exterior desig

10 Homes of this quality generally have high quality features. Fin
more design quality is seen in the floor plans. Generally have a la

11 Custom design and higher quality finish work with added ame
bathroom fixtures and more luxurious options.

12 Custom design and excellent builders. All materials are of the
conveniences are present.

13 Generally custom designed and built. Mansion level. Large an
cabinet work, wood trim, marble, entry ways etc.

In [55]:

```
data['grade'].value_counts()
```

```
7 Average        8762
8 Good           5928
9 Better         2505
6 Low Average    1967
10 Very Good     1046
11 Excellent      328
5 Fair            227
12 Luxury          54
4 Low              27
13 Mansion          5
3 Poor              1
Name: grade, dtype: int64
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [56]:

```
data['grade'].replace(['3 Poor', '4 Low', '5 Fair', '6 Low A
                       '7 Average', '8 Good', '9 Better', '1
                       '11 Excellent', '12 Luxury', '13 Mans
                        [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10],
```

In [57]:

```
data['grade'].value_counts()
```

```
4     8762
5     5928
6     2505
3     1967
7     1046
8      328
2      227
9       54
1       27
10       5
0        1
Name: grade, dtype: int64
```

For our purposes, it doesn't make sense to keep any Luxury or M
the data set, since those kinds of properties would need to be rer
better use of the space, and would effectively be making these pr
It also doesn't make sense to keep the single property in the poo
assume that property will be on the list, and take it out of the data
to drop those rows.

Likewise it doesn't make sense to keep homes in this data set tha
because these homes would need essential building done, such
and water. While we are looking to do renovations on properties,
distinction between building and renovating. Therefore I'll be drop
or below.

FURTHER WORK:

Contents

- 6.1 Comparing Renovated Homes with Unrenovated
- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

I would like to come back to this variable in future works and do f
Homes currently at a grade of 6 or above meet building codes tha
place. It may be beneficial to come back and raise the minimum g
data set to 6, to lower the cost of renovations needed to make the
public housing. All homes not currently up to code would need to
in renovations, should they be selected for the Section 8 Expansi

In [58]:

```
poor = data.loc[data['grade'] == 0]

# labeling this single property for the city to buy, buldoz
# make into a small public park
```

In [59]:

```
data = data.loc[data['grade'] < 9]
data = data.loc[data['grade'] >= 3]
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [60]:

```
sns.distplot( a=data["grade"], hist=True, kde=False, rug=Fa
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
y:2551: FutureWarning: `distplot` is a deprecated function and will be rem
n. Please adapt your code to use either `displot` (a figure-level function
y) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
  - 7.1 Getting Infrastucture Data
    - 7.1.1 Downtown
    - 7.1.2 Hospitals
    - 7.1.3 Police Stations
    - 7.1.4 Schools
  - 7.2 Finding Distance for Infrastructure
  - 7.3 Exploring Distance Data
    - 7.3.1 Downtown
    - 7.3.2 Hospitals
    - 7.3.3 Police Stations
    - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [61]:

```
sns.regplot(data = data, x = 'price', y = 'grade',
            line_kws={"color":"r","alpha":0.7,"lw":5})
plt.show()
```

Observations: As grade improves, so does price, this is what we'

5.0.3 View

FROM THE DATASET: Includes views of Mt. Rainier, Olympics, C Seattle Skyline, Puget Sound, Lake Washington, Lake Sammami creek, and other. Represents quality of view from house.

Since most properties have "none" listed as the view, and this ca start with, I am going to convert it to a binary column. Homes wit views will have a value of 1, all other homes will have a value of (

Contents

6.1 Comparing Renovated Homes with Unrenovated
6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [62]:

```
data['view'].value_counts()
```

```
NONE         18569
AVERAGE        883
GOOD           455
FAIR           316
EXCELLENT      253
Name: view, dtype: int64
```

In [63]:

```
data['view'].replace(['NONE', 'FAIR', 'AVERAGE', 'GOOD', 'E
                      [0, 0, 0, 1, 1], inplace=True)

# replacing the values to binary values
```

In [64]:

```
data['view'].fillna(value=0,inplace=True)

# filling in n/a values with 0
```

In [65]:

```
data['view'].value_counts()

# taking a look at the distribution now
```

```
0.0    19828
1.0      708
Name: view, dtype: int64
```

Contents

6.1 Comparing Renovated Homes with Unrenovated
6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [66]:

```
data['view'].hist()

# it's easier to see visually
```

```
<AxesSubplot:>
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [67]:

```
sns.boxplot(data = data, x = 'view', y = 'price', palette =
plt.show();
```

Observations: We can see here that having a view has a clear in
the home, and there is a much larger spread in homes with a vie
Likewise, we can see that homes without a view are generally les
homes with a view but there are very many outlying cases where

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [68]:

```
data.isna().sum()

#making sure there are no remaining columns with missing va
```

```
id                 0
date               0
price              0
bedrooms           0
bathrooms          0
sqft_living        0
sqft_lot           0
floors             0
waterfront         0
view               0
condition          0
grade              0
sqft_above         0
sqft_basement      0
yr_built           0
yr_renovated    3658
zipcode            0
lat                0
long               0
sqft_living15      0
sqft_lot15         0
dtype: int64
```

5.1 Year Renovated

Because we will be renovating each property selected for this pilo
important to understand the effects renovating a property will hav

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fc
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Loc
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [69]:

```
data['yr_renovated'].hist()
# taking a look at the distributions of homes renovated vs
```

```
<AxesSubplot:>
```

This plot doesn't tell us much except that the majority of homes ir never been renovated.

In [70]:

```
data['yr_renovated'].mean()
```

```
82.43672236046925
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [71]:

```
data['yr_renovated'].isna().sum()/data.shape[0]

# 17% of the data in this column is missing, and in this co
```

```
0.17812621737436696
```

In [72]:

```
data['yr_renovated'].fillna(value = 0, inplace = True)
```

I'm filling the NA values with 0, as the vast majority of homes hav
renovated in this dataset.

6 Analyzing Property Timelines

Because I'm now finished cleaning and preparing the data, I can
data exists.

6.1 Comparing Renovated Homes with Homes

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [73]:

```
# the mean price for a home not renovated in this data set

no_reno = data.loc[data['yr_renovated'] == 0]
no_reno_stats = no_reno.mean()
no_reno_stats
```

```
id               4.648427e+09
price            5.178077e+05
bedrooms         3.373759e+00
bathrooms        2.102563e+00
sqft_living      2.047173e+03
sqft_lot         1.018782e+04
floors           1.496295e+00
waterfront       4.082867e-03
view             3.165482e-02
condition        2.419174e+00
grade            4.651394e+00
sqft_above       1.762171e+03
sqft_basement    3.855537e-01
yr_built         1.972364e+03
yr_renovated     0.000000e+00
zipcode          9.807812e+04
lat              4.756143e+01
long            -1.222163e+02
sqft_living15    1.972658e+03
sqft_lot15       9.898066e+03
dtype: float64
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [74]:

```
yes_reno = data.loc[data['yr_renovated'] != 0].copy()
yes_reno_stats = yes_reno.mean()
yes_reno_stats
```

```
id               4.514145e+09
price            7.200811e+05
bedrooms         3.440459e+00
bathrooms        2.268651e+00
sqft_living      2.248066e+03
sqft_lot         1.052812e+04
floors           1.502869e+00
waterfront       3.299857e-02
view             1.147776e-01
condition        2.216643e+00
grade            4.714491e+00
sqft_above       1.816657e+03
sqft_basement    5.007174e-01
yr_built         1.939235e+03
yr_renovated     1.996222e+03
zipcode          9.809764e+04
lat              4.758326e+01
long            -1.222680e+02
sqft_living15    1.947396e+03
sqft_lot15       9.874834e+03
dtype: float64
```

FURTHER WORK:

I'd like to come back to these statistics and do a much further ana
differences between the average renovated home vs non-renova
variables besides price. IE does the average renovated home ha
or neighbors with more living space than non-renovated counterp
only be comparing the prices.

I'd also like to come back to the data and isolate non-renovated h
up in the last 10 years from all other non-renovated homes and c
groups. However, in my initial research, I have only had the resou

Contents

- 6.1 Comparing Renovated Homes with Unrenovated
- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

inspection of renovated VS non-renovated.

In [75]:

```
reno_differences = yes_reno_stats - no_reno_stats
reno_difference = reno_differences['price']
reno_difference
```

```
202273.36808995338
```

Of note: this is the exact calculated difference in average price be have been renovated vs. homes that have not been renovated, ir so far cleaned and prepared. While this statistic has not been sca as bedrooms or living space for example, this statistic does indica that were renovated in this data set where $201,941 more valuab

In [76]:

```
average_yes_reno_price = yes_reno_stats['price']
average_yes_reno_price
```

```
720081.1162123386
```

In [77]:

```
budget = 100000000
baseline_yes_reno = budget / average_yes_reno_price
baseline_yes_reno
```

```
138.87324323404678
```

The average non renovated home in the remaining data after cle is priced at $719,794. If we were to hypothetically spend our enti average renovated home, we'd be able to buy 138 properties wit budget.

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [78]:

```
average_no_reno_price = no_reno_stats['price']
average_no_reno_price
```

```
517807.7481223852
```

In [79]:

```
baseline_no_reno = budget / average_no_reno_price
baseline_no_reno
```

```
193.12186880673084
```

In [80]:

```
baseline_no_reno - baseline_yes_reno
```

```
54.24862557268406
```

The average non renovated home in the remaining data after cle is priced at $517,853. That may sound pretty close to an average fits into the laid out budget for the project about 193 times.

For clarity, if we were only buying average priced non-renovated to buy 54 more homes than if we were only buying average price homes.

To properly compare visually, I will take each home that has beer subtract the average price of a non-renovated home from the sal recenter the visualization's bars to show how above or below eac compared to the average un-renovated home.

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [81]:

```
# make a function where the input is each price for each pr
# been renovated and the output is that price minus the ave
# renovated home

def find_price_adjusted(price):
    price = price - average_no_reno_price
    return price

yes_reno['price'] = yes_reno['price'].map(find_price_adjust
```

To organize the data, and get a sense of timeline, I'm going to be properties into chronological bins. Every 5 years will show very b decade.

FURTHER WORK:

I'd like to come back to this visualization with more time and reso bins every four years, in correlation with the mayoral history of Se

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
  - 7.1 Getting Infrastucture Data
    - 7.1.1 Downtown
    - 7.1.2 Hospitals
    - 7.1.3 Police Stations
    - 7.1.4 Schools
  - 7.2 Finding Distance for Infrastructure
  - 7.3 Exploring Distance Data
    - 7.3.1 Downtown
    - 7.3.2 Hospitals
    - 7.3.3 Police Stations
    - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [82]:

```
# aggregating the data by half-decade

bins = [ 1935, 1940, 1945, 1950, 1955, 1960, 1965, 1970, 19
         1990, 1995, 2000, 2005, 2010, 2015]

labels = ['1940', '1945', '1950', '1955', '1960', '1965', '
          '1985', '1990', '1995', '2000', '2005', '2010', '

cut_data, bin_edges = pd.cut(yes_reno['yr_renovated'], bins
                             labels = labels, retbins = Tr

bin_edges
```

```
array([1935, 1940, 1945, 1950, 1955, 1960, 1965, 1970, 1975, 1980, 1985,
       1990, 1995, 2000, 2005, 2010, 2015])
```

In [83]:

```
yes_reno.head()

#taking a look to make sure aggrigation worked correctly
```

	id	date	price	bedrooms	bathrooms	sqft_living	s
1	6414100192	12/9/2014	20192.251878	3	2.25	2570	7
35	9547205180	6/13/2014	178192.251878	3	2.50	2300	3
95	1483300570	9/8/2014	387192.251878	4	2.50	3300	1
103	2450000295	10/7/2014	572192.251878	3	2.50	2920	8
125	4389200955	3/2/2015	932192.251878	4	2.75	2750	1

5 rows × 21 columns

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
  - 7.1 Getting Infrastucture Data
    - 7.1.1 Downtown
    - 7.1.2 Hospitals
    - 7.1.3 Police Stations
    - 7.1.4 Schools
  - 7.2 Finding Distance for Infrastructure
  - 7.3 Exploring Distance Data
    - 7.3.1 Downtown
    - 7.3.2 Hospitals
    - 7.3.3 Police Stations
    - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [84]:

```
yes_reno['yr_renovated'] = cut_data

# replacing the yr_renovated with the binned data
```

Since I have homes grouped by 5 year bins, it's important that fo
average price instead of a sum value of all the properties in that b
different numbers of properties in each bin. Getting the average p
something comparable.

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [85]:

```
# making sure I get the average per bin, and resetting the

yes_reno_averages = yes_reno.groupby(by='yr_renovated').mea
yes_reno_averages = yes_reno_averages.reset_index()
yes_reno_averages
```

	yr_renovated	id	price	bedrooms	bathrooms	sqft_livi
0	1940	1.451350e+09	-139407.748122	2.000000	1.000000	1150.0000
1	1945	4.448326e+09	-88557.748122	2.750000	1.687500	1607.5000
2	1950	3.606523e+09	-146841.081456	1.333333	1.166667	963.33333
3	1955	4.577260e+09	-59474.414789	3.166667	1.458333	1586.6666
4	1960	5.478101e+09	17870.251878	3.100000	1.500000	1814.5000
5	1965	3.869926e+09	81296.098031	3.846154	2.038462	2057.6923
6	1970	4.685418e+09	-26341.081456	3.380952	1.904762	1993.8095
7	1975	5.433291e+09	-58104.081456	2.933333	1.683333	1822.6666
8	1980	4.851001e+09	147021.797332	2.909091	1.931818	1958.1818
9	1985	4.466804e+09	90343.251878	3.320755	2.037736	2063.5849
10	1990	4.876861e+09	158870.585211	3.291667	2.093750	2166.2500
11	1995	4.523660e+09	305449.735749	3.451613	2.282258	2379.0161
12	2000	4.254875e+09	261863.451878	3.400000	2.523333	2478.4933
13	2005	4.306589e+09	343149.008634	3.630631	2.495495	2466.2522
14	2010	4.364097e+09	308062.113580	3.595745	2.500000	2468.5531
15	2015	4.627709e+09	132147.432329	3.563910	2.313910	2121.8270

6.2 Visualizing the Comparison

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [86]:

```
# color coding the bars to be green if the average price of
# is higher than the average non-renovated home, and grey i
# visual intuition

values = yes_reno_averages['price']
colors = ['grey' if (val < 0) else 'green' for val in values

# making the graph
plt.figure(figsize = (20,7))
ax = sns.barplot(data = yes_reno_averages, x = 'yr_renovated
                 y ='price', palette = colors)

# making the labels for the bins
ax.set_xticklabels(['1935-1940', '1940-1945', '1945-1950',
                     '1955-1960', '1960-1965', '1965-1970'
                    '1975-1980', '1980-1985', '1985-1990',
                    '1995-2000', '2000-2005', '2005-2010',

# rotating my labels
plt.xticks(rotation=45);

# making the blue line where the average price for renovated
plt.axhline(reno_difference, color = 'blue', label = "Averag
plt.legend()

# making the labels and titles for the axes
ax.set(xlabel = 'Year Renovated',
                 ylabel = 'Price of Property Compared To Av
                 title = "Average Price of Renovated Home C
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

KEY

X axis = average price of non-renovated home

Green Bar = above average price of non-renovated home

Blue Line = average price of renovated home

This plot's 0 on the Y axis has been adjusted so that everything a where their price (grouped by every 5 years,) is above the averag renovated property. The blue line represents the average price of been renovated. Properties above the blue line represent renova valued above the average renovated property.

Observations:

1) all properties renovated after 1975 are more valuable than the property.

2) properties renovated in the years between 1990 to 2010, are a price compared even to other renovated properties.

Inferences:

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
  - 7.1 Getting Infrastucture Data
    - 7.1.1 Downtown
    - 7.1.2 Hospitals
    - 7.1.3 Police Stations
    - 7.1.4 Schools
  - 7.2 Finding Distance for Infrastructure
  - 7.3 Exploring Distance Data
    - 7.3.1 Downtown
    - 7.3.2 Hospitals
    - 7.3.3 Police Stations
    - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

Renovating properties increases their value, that's to be expecte
can now infer that if we renovate properties that have already be

6.3 Home History

6.3.1 Age of Home

In [87]:

```
def age_of_home(year_built):
    age = 2014 - year_built
    return age

data['age_of_home'] = data['yr_built'].map(age_of_home)

# showing the age of the home rather than the year it was b
# a different perspective on the data
```

In [88]:

```
data.head()
```

	id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot
0	7129300520	10/13/2014	221900.0	3	1.00	1180	5650
1	6414100192	12/9/2014	538000.0	3	2.25	2570	7242
2	5631500400	2/25/2015	180000.0	2	1.00	770	10000
3	2487200875	12/9/2014	604000.0	4	3.00	1960	5000
4	1954400510	2/18/2015	510000.0	3	2.00	1680	8080

5 rows × 22 columns

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
  - 7.1 Getting Infrastucture Data
    - 7.1.1 Downtown
    - 7.1.2 Hospitals
    - 7.1.3 Police Stations
    - 7.1.4 Schools
  - 7.2 Finding Distance for Infrastructure
  - 7.3 Exploring Distance Data
    - 7.3.1 Downtown
    - 7.3.2 Hospitals
    - 7.3.3 Police Stations
    - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [89]:

```
data['age_of_home'].hist()

# seeing the distributions of how old the home is
```

```
<AxesSubplot:>
```

6.3.2 Year Modified

I'm going to make a new column for year modified. This column w
amount of years since last renovation or the date of construction,
happened most recently. This column helps liken newer non-renc
recently renovated homes, and older non-renovated homes with
renovated a while ago.

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [90]:

```
data['yr_mod'] = data[['yr_built','yr_renovated']].max(axis=

# this will select the most recent value between either yea
```

In [91]:

```
data.head()
```

	id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot
0	7129300520	10/13/2014	221900.0	3	1.00	1180	5650
1	6414100192	12/9/2014	538000.0	3	2.25	2570	7242
2	5631500400	2/25/2015	180000.0	2	1.00	770	10000
3	2487200875	12/9/2014	604000.0	4	3.00	1960	5000
4	1954400510	2/18/2015	510000.0	3	2.00	1680	8080

5 rows × 23 columns

In [92]:

```
data['yr_mod'].value_counts().head()
```

```
2014.0    626
2005.0    465
2006.0    452
2003.0    440
2004.0    435
Name: yr_mod, dtype: int64
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
  - 7.1 Getting Infrastucture Data
    - 7.1.1 Downtown
    - 7.1.2 Hospitals
    - 7.1.3 Police Stations
    - 7.1.4 Schools
  - 7.2 Finding Distance for Infrastructure
  - 7.3 Exploring Distance Data
    - 7.3.1 Downtown
    - 7.3.2 Hospitals
    - 7.3.3 Police Stations
    - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [93]:

```
sns.scatterplot(data = data, x = 'yr_mod', y = 'price')
plt.show()
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [94]:

```
sns.distplot( a=data["yr_mod"], hist=True, kde=False, rug=Fa
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
y:2551: FutureWarning: `distplot` is a deprecated function and will be rem
n. Please adapt your code to use either `displot` (a figure-level function
y) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

```
<AxesSubplot:xlabel='yr_mod'>
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

7 Distance Data

Now that I have limited the amount of properties I'll be looking up for, it's time to gather some more data.

Using Google's Geocoding api and scraping with Selenium to me from downtown, police stations, hospitals and schools.

ASSUMPTIONS:

I have not had the resources yet to cross reference if the infrastru have changed since 2015. I'm assuming in this data that the loca Police Stations and Schools in 2022 is the same as it was in 201

FURTHER WORK:

I'd also like to include data about firehouses, which I have a sour "Data" but didn't have the resources to currently explore. As the s increases, I'd like to include distances away from public parks, lib universities, and public transportation.

I'd also like to do more research about the history of these infrast

7.1 Getting Infrastucture Data

In [95]:

```
# for the api

with open('.secrets/google.json') as f:
    creds = json.load(f)
```

Making a function that intakes a list of addresses and outputs a li containing the coordinates for that address using google's API.

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [96]:

```
# making an empty dictionary to contain the address as the
# of latitude and logitude as the values.

def get_coordinates(addresses):
    for address in addresses:

        google_query = f"https://maps.googleapis.com/maps/a

        results = requests.get(google_query)
        results_json = results.json()
        lat = results_json['results'][0]['geometry']['locat
        long = results_json['results'][0]['geometry']['loca

        coordinates[f"{address}"]=(lat, long)
```

7.1.1 Downtown

In [97]:

```
downtown = (47.6050, -122.3344)
```

Making a function that intakes the dataframe, and a specific locat
coordinates and dds a new row to the DF, column name as the p
that location, this column lists the distance each property is away

7.1.2 Hospitals

CITATION

https://data-seattlecitygis.opendata.arcgis.com/datasets/hospitals/e
(https://data-seattlecitygis.opendata.arcgis.com/datasets/hospitals/
showTable=true)

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [98]:

```
# loading the data

hospital_data = pd.read_csv('Data/Hospitals.csv')
```

In [99]:

```
hospital_data.head()

# taking a look at the data in this csv
```

	X	Y	OBJECTID	FACILITY	ADDRESS	SE_ANNO_CAD_DATA
0	-122.336888	47.714248	1	UW Medical Center - Northwest	1550 N 115th ST	NaN
1	-122.379555	47.667365	2	Swedish Medical Center - Ballard Campus	5300 Tallman Ave NW	NaN
2	-122.341419	47.457800	3	Highline Medical Center	16251 Sylvester Rd SW	NaN
3	-122.214121	47.442273	4	Valley Medical Center	400 S 43rd St	NaN
4	-122.327279	47.292612	5	St. Francis Hospital	34515 9th Ave S	NaN

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [100]:

```
hospital_addresses = hospital_data['ADDRESS'] + hospital_da

# getting proper addresses for the Google API
```

In [101]:

```
hospital_addresses.head()

# seeing the addresses
```

```
0         1550 N 115th STSeattle, WA, 98133-8401
1           5300 Tallman Ave NWSeattle, WA 98107
2    16251 Sylvester Rd SWBurien, WA, 98166-3017
3            400 S 43rd StRenton, WA, 98055-5714
4     34515 9th Ave SFederal Way, WA, 98003-6761
dtype: object
```

In [102]:

```
hospital_names = hospital_data['FACILITY']

# making a list of hospital names for later
```

7.1.3 Police Stations

CITATION

http://www.seattle.gov/police/about-us/police-locations (http://www.seattle.gov/police/about-us/police-locations)

Contents

6.1 Comparing Renovated Homes with Unrenovated
6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [103]:

```
# I didn't bother to scrape for police data, there are only
# scraping would be more work than hardcoding at this point

police_stations = {'north_ps' : "10049 College Way N. Seatt
'west_ps' : "810 Virginia Street, Seattle, WA 98101",
'east_ps' : "1519 12th Avenue Seattle, WA 98122",
'south_ps' : "3001 S. Myrtle Seattle, WA 98108",
'southwest_ps' : "2300 S.W. Webster Seattle, WA 98106"}

ps_addresses = police_stations.values()
ps_names = police_stations.keys()
```

7.1.4 Schools

For the purposes of this project, I'm only including public school c

Scraping for School Data

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
  - 7.1 Getting Infrastucture Data
    - 7.1.1 Downtown
    - 7.1.2 Hospitals
    - 7.1.3 Police Stations
    - 7.1.4 Schools
  - 7.2 Finding Distance for Infrastructure
  - 7.3 Exploring Distance Data
    - 7.3.1 Downtown
    - 7.3.2 Hospitals
    - 7.3.3 Police Stations
    - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [104]:

```
# first telling selenium where the webdriver is on my local
driver = webdriver.Chrome('/Users/b0ihazard/Desktop/chromed

# telling selenium where the url is that I want to scrape f
# that up
school_url = "https://www.seattleschools.org/schools/"
go_article = driver.get(school_url)

# making a list of school names and addresses as selenium o
grab_school_names = driver.find_elements(By.CSS_SELECTOR, "
grab_school_addresses = driver.find_elements(By.CSS_SELECTO

# making corrasponding lists of addresses and names for tex
school_addresses = []
school_names = []

# reformatting each raw address to a string and putting it
for raw_address in grab_school_addresses:
    address = raw_address.text.split("Main Office")[0]
    school_addresses.append(address)

# reformatting each raw name to a string and putting it on
for raw_name in grab_school_names:
    school_name = raw_name.text
    school_names.append(school_name)

# closing the webdriver after use
driver.close()
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [105]:

```
# taking a look at each school just to makesure formatting

number = 0

for address in school_addresses:
    number = number + 1
    print(number, address)
```

```
1 6110 28th Ave. NW
Seattle, WA 98107

2 3928 S Graham St.
Seattle, WA 98118

3 8601 Rainier Ave. S
Seattle, WA 98118

4 3010 59th Ave. SW
Seattle, WA 98116

5 3701 SW 104th St.
Seattle, WA 98146

6 1301 E Yesler Way
Seattle, WA 98122

7 1418 NW 65th St.
Seattle, WA 98117
```

In [106]:

```
school_addresses.remove(school_addresses[65])
```

This isn't an address so we must remove it from the data, but I ha
take the accompanying name, Middle College from the list as we

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [107]:
```
school_names[65]
```

```
'Middle College'
```

In [108]:
```
school_names.remove('Middle College')
```

7.2 Finding Distance for Infrastructure

Getting a dictionary of coordinates based on the collected addres
Police Stations, Schools

In [109]:
```
coordinates ={}
```

In [110]:
```
get_coordinates(hospital_addresses)
```

In [111]:
```
get_coordinates(ps_addresses)
```

In [112]:
```
get_coordinates(school_addresses)
```

In [113]:
```
coordinates['downtown'] = downtown
```

Getting a list of all the names of these points of interests for the c

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [114]:

```
all_names = []

for hospital in hospital_names:
    all_names.append(hospital)
for ps_name in ps_names:
    all_names.append(ps_name)
for school_name in school_names:
    all_names.append(school_name)

```

In [115]:

```
all_names.append('downtown')
```

In [116]:

```
len(all_names) == len(coordinates)

# to make sure that each name has a set of coordinates and
```

```
True
```

In [117]:

```
counter = 0
columns = {}
for name in all_names:
    columns[name] = counter
    counter +=1

# making a dictionary of column names with an index for ref
```

Making a function that will calculate how far away each point of in
each point of interest a column, and making the rows how far awa
is from each property. Each column should be labeled properly by

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [118]:

```
# intake: dataframe, location as a touple of coordinates, l
# output: column on dataframe that measures distance from l

def distance_from (data, coordinates, all_names):
    counter = 0
    for coordinate_value in coordinates.values():
        distance_from_coordinates = []
        col_name = all_names[counter]
        for x in data.index:
            property_coordinates = (data.loc[x]['lat'], data
            dist = distance.distance(property_coordinates, 
            distance_from_coordinates.append(dist)
        data[col_name] = distance_from_coordinates
        counter += 1

#     return f"added distance from {coordinate}, {col_name}
```

In [119]:

```
distance_from(data, coordinates, all_names)

# implimenting this function on all of our data does take s
```

Finding the closest possible Hospital, Police Station and Sch property.

FURTHER WORK:

This is now a much larger amount of information, I've added over original dataset. In further work, I would like to isolate each point to compare how individual points of interest affect property value other individual points of interest. However, given the parameters

Contents

- 6.1 Comparing Renovated Homes with Unrenovated
- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

must simplify this data significantly. Therefore, I'll be reducing the
only three columns, to show the closest possible hospital, police
This will avoid the multicoliniarity problem with having too much c

In [120]:

```
hospitals = hospital_names
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [121]:

```
# making a dataframe just for the hospital information for
# the last columns in this dataframe will default to the di

data_h = data[hospitals]
data_h.head()
data_h['closest_hospital_distance'] = data[['UW Medical Cen
 'Swedish Medical Center - Ballard Campus',
 'Highline Medical Center',
 'Valley Medical Center',
 'St. Francis Hospital',
 'Snoqualmie Valley Hospital',
 'Swedish Medical Center - First Hill',
 'MultiCare  Auburn Medical Center',
 'Overlake Hospital Medical Center',
 "Seattle Children's Hospital",
 'Virginia Mason Medical Center',
 'EvergreenHealth - Kirkland',
 'Swedish Medical Center - Cherry Hill',
 'Seattle VA Medical Center',
 'Swedish Medical Center - Issaquah Campus',
 'Kaiser Permanente Capitol Hill Campus',
 'St. Elizabeth Hospital',
 'Harborview Medical Center',
 'University of Washington Medical Center']].min(axis=1)
```

```
<ipython-input-121-b04b706e5bbc>:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/use
view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-
  data_h['closest_hospital_distance'] = data[['UW Medical Center - Northwe
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [122]:

```
making a dataframe just for school information for future
the last columns in this dataframe will default to the di

ata_s = data[school_names]
ata_s['closest_school_distance'] = data[['Adams Elementary
Aki Kurose Middle School',
Alan T. Sugiyama High School',
Alki Elementary',
Arbor Heights Elementary',
Bailey Gatzert Elementary',
Ballard High School',
Beacon Hill International Elementary',
Benjamin Franklin Day Elementary',
Broadview-Thomson K-8',
Bryant Elementary',
Cascade Parent Partnership Program',
Cascadia Elementary',
Catharine Blaine K-8',
Cedar Park Elementary',
Chief Sealth International High School',
Cleveland High School',
Coe Elementary',
Concord International Elementary',
Daniel Bagley Elementary',
Dearborn Park International Elementary',
Decatur Elementary',
Denny International Middle School',
Dunlap Elementary',
Eckstein Middle School',
Emerson Elementary',
Fairmount Park Elementary',
Franklin High School',
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools


```
Garfield High School',
Gatewood Elementary',
Genesee Hill Elementary',
Graham Hill Elementary',
Green Lake Elementary',
Greenwood Elementary',
Hamilton International Middle School',
Hawthorne Elementary',
Hazel Wolf K-8',
Highland Park Elementary',
Ingraham High School',
Interagency Academy',
Jane Addams Middle School',
John Hay Elementary',
John Muir Elementary',
John Rogers Elementary',
John Stanford International Elementary',
Kimball Elementary',
Lafayette Elementary',
Laurelhurst Elementary',
Lawton Elementary',
Leschi Elementary',
Licton Springs K-8',
Lincoln High School',
Louisa Boren K-8',
Lowell Elementary',
Loyal Heights Elementary',
Madison Middle School',
Madrona Elementary',
Magnolia Elementary',
Maple Elementary',
Martin Luther King, Jr. Elementary',
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools


```
McClure Middle School',
McDonald International Elementary',
McGilvra Elementary',
Meany Middle School',
Mercer International Middle School',
Montlake Elementary',
Nathan Hale High School',
North Beach Elementary',
Northgate Elementary',
Nova',
Olympic Hills Elementary',
Olympic View Elementary',
Orca K-8',
Pathfinder K-8',
Queen Anne Elementary',
Rainier Beach High School',
Rainier View Elementary',
Rising Star Elementary',
Robert Eagle Staff Middle School',
Roosevelt High School',
Roxhill Elementary',
Sacajawea Elementary',
Salmon Bay K-8',
Sand Point Elementary',
Sanislo Elementary',
Seattle World School',
Skills Center',
South Shore PK-8',
Stevens Elementary',
The Center School',
Thornton Creek Elementary',
Thurgood Marshall Elementary',
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools


```
TOPS K-8',
View Ridge Elementary',
Viewlands Elementary',
Washington Middle School',
Wedgwood Elementary',
West Seattle Elementary',
West Seattle High School',
West Woodland Elementary',
Whitman Middle School',
Whittier Elementary',
Wing Luke Elementary']].min(axis=1)
```

```
<ipython-input-122-d282df538c6c>:5: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/use
view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-
  data_s['closest_school_distance'] = data[['Adams Elementary',
```

In [123]:

```
# data.to_csv('Housing_Project_Full')
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [124]:

```
# making a dataframe just for police stations for future wo
# the last columns in this dataframe will default to the di


data_ps = data[ps_names]
data_ps['closest_police_station'] = data[['north_ps', 'west_
                                          'south_ps', 'soutl
```

```
<ipython-input-124-643df7c19b3e>:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/use
view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-
  data_ps['closest_police_station'] = data[['north_ps', 'west_ps', 'east_p
```

Simplifying the Data To Only Show the Closest Distance

In [125]:

```
to_drop = hospitals, school_names, ps_names
for columns in to_drop:
    data = data.drop(columns = columns)

# dropping all distance data from main dataframe
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [126]:

```
data.head()
```

	id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot
0	7129300520	10/13/2014	221900.0	3	1.00	1180	5650
1	6414100192	12/9/2014	538000.0	3	2.25	2570	7242
2	5631500400	2/25/2015	180000.0	2	1.00	770	10000
3	2487200875	12/9/2014	604000.0	4	3.00	1960	5000
4	1954400510	2/18/2015	510000.0	3	2.00	1680	8080

5 rows × 24 columns

In [127]:

```
_h['closest_hospital_distance'], data_s['closest_school_distance'
 to_add:
ta.join(column)

 lified distance data back into main dataframe
```

Contents

6.1 Comparing Renovated Homes with Unrenovated
6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [128]:

```
data.head()
```

	id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot
0	7129300520	10/13/2014	221900.0	3	1.00	1180	5650
1	6414100192	12/9/2014	538000.0	3	2.25	2570	7242
2	5631500400	2/25/2015	180000.0	2	1.00	770	10000
3	2487200875	12/9/2014	604000.0	4	3.00	1960	5000
4	1954400510	2/18/2015	510000.0	3	2.00	1680	8080

5 rows × 27 columns

7.3 Exploring Distance Data

7.3.1 Downtown

I'm going to make a cut-off for houses more than 25 miles away f
goal of this project is to increase affordable housing in Seattle pro

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [129]:

```
sns.regplot(data = data, x = 'price', y = 'downtown',
            line_kws={"color":"r","alpha":0.3,"lw":5})
plt.show()
```

In [130]:

```
data = data.loc[data['downtown'] < 25]
```

In [131]:

```
data['downtown'].max()
```

```
24.934920914948457
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [132]:

```
sns.regplot(data = data, x = 'price', y = 'downtown',
            line_kws={"color":"r","alpha":0.3,"lw":5})
plt.show()
```

We can see some clustering in this graph, showing us that prices from downtown increases in the maximum prices for homes.

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [133]:

```
sns.distplot( a=data["downtown"], hist=True, kde=False, rug
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
y:2551: FutureWarning: `distplot` is a deprecated function and will be rem
n. Please adapt your code to use either `displot` (a figure-level function
y) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

```
<AxesSubplot:xlabel='downtown'>
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fc
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Loc
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [134]:

```
data['downtown'].describe()
```

```
count    20109.000000
mean        10.923908
std          5.919272
min          0.719565
25%          5.979404
50%         10.079701
75%         15.018175
max         24.934921
Name: downtown, dtype: float64
```

7.3.2 Hospitals

In [135]:

```
sns.regplot(data = data, x = 'price', y = 'closest_hospital_
            line_kws={"color":"r","alpha":0.3,"lw":5})
plt.show()
```

Observations: while there are plenty of low priced homes close to
limit of how expensive a home can be does look to go down the f

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

hospital it is.

In [136]:

```
sns.distplot( a=data["closest_hospital_distance"], hist=Tru
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
y:2551: FutureWarning: `distplot` is a deprecated function and will be rem
n. Please adapt your code to use either `displot` (a figure-level function
y) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

```
<AxesSubplot:xlabel='closest_hospital_distance'>
```

7.3.3 Police Stations

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [137]:

```
sns.regplot(data = data, x = 'price', y = 'closest_police_s
            line_kws={"color":"r","alpha":0.3,"lw":5})
plt.show()
```

Observations: while there are plenty of low priced homes close to upper limit of how expensive a home can be does look to go dow from a police station it is, similar to hospitals.

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [138]:

```
sns.distplot( a=data["closest_police_station"], hist=True, k
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
y:2551: FutureWarning: `distplot` is a deprecated function and will be rem
n. Please adapt your code to use either `displot` (a figure-level function
y) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

```
<AxesSubplot:xlabel='closest_police_station'>
```

7.3.4 Schools

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
  - 7.1 Getting Infrastucture Data
    - 7.1.1 Downtown
    - 7.1.2 Hospitals
    - 7.1.3 Police Stations
    - 7.1.4 Schools
  - 7.2 Finding Distance for Infrastructure
  - 7.3 Exploring Distance Data
    - 7.3.1 Downtown
    - 7.3.2 Hospitals
    - 7.3.3 Police Stations
    - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [139]:

```
sns.regplot(data = data, x = 'price', y = 'closest_school_d
            line_kws={"color":"r","alpha":0.3,"lw":5})
plt.show()
```

Observations: while there are plenty of low priced homes close to
limit of how expensive a home can be does look to go down the f
school it is, similar to hospitals and police stations.

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fc
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Loc
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [140]:

```
sns.distplot( a=data["closest_school_distance"], hist=True,
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
y:2551: FutureWarning: `distplot` is a deprecated function and will be rem
n. Please adapt your code to use either `displot` (a figure-level function
y) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

```
<AxesSubplot:xlabel='closest_school_distance'>
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

8 Preparing for Modeling

8.0.1 Dropping View and Waterfront Properties Modeling

Since we are using a linear regression model, I must drop values features.

In [141]:
```
data['view'].value_counts()
```

```
0.0    19419
1.0      690
Name: view, dtype: int64
```

In [142]:
```
data['waterfront'].value_counts()
```

```
0    20005
1      104
Name: waterfront, dtype: int64
```

As a majority of data does not have a view or a waterfront, in ord enough data to manipulate and have findings I must drop propert values are at 1.

In [143]:
```
data = data.loc[data['view'] == 0]
data = data.loc[data['waterfront'] == 0]
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [144]:

```
data.hist(figsize=(20,20));
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
  - 7.1 Getting Infrastucture Data
    - 7.1.1 Downtown
    - 7.1.2 Hospitals
    - 7.1.3 Police Stations
    - 7.1.4 Schools
  - 7.2 Finding Distance for Infrastructure
  - 7.3 Exploring Distance Data
    - 7.3.1 Downtown
    - 7.3.2 Hospitals
    - 7.3.3 Police Stations
    - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

9 Zipcode

Using this kind of model, I'll need to pick one zipcode as the defa
compare the price of a property going up or down depending on

Seeing as this project aims to specifically research class disparity
my default zipcode, it's the wealthiest zipcode in Seattle.

CITATION: https://www.zipdatamaps.com/economics/income/agi/r
zipcodes-in-metro-seattle-tacoma
(https://www.zipdatamaps.com/economics/income/agi/metro/wealt
metro-seattle-tacoma) (Source: US Internal Revenue Service - 20

By comparing all other zipcodes to the wealthiest zipcode, we ca
how much a zipcode can affect property values. I'm assuming 1)
infrastructure is working well in this zipcode, and is reflected parti
also assuming 2) that this zipcode was also the wealthiest zipcoc

FURTHER WORK: I'd like to, in the future, expand this model to
functionality, so the user can select which zipcode they'd like to n
zipcode. That would allow for civic workers to compare each and
and ever zipcode in relative scale to each other.

In [145]:

```
zip_dummies = pd.get_dummies(data.zipcode, drop_first=False

# Concatenate the dummies to original dataframe
zipcode_data = pd.concat([data, zip_dummies], axis='columns

## drop zipcode 98039 instead of "first"
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
  - 7.1 Getting Infrastucture Data
    - 7.1.1 Downtown
    - 7.1.2 Hospitals
    - 7.1.3 Police Stations
    - 7.1.4 Schools
  - 7.2 Finding Distance for Infrastructure
  - 7.3 Exploring Distance Data
    - 7.3.1 Downtown
    - 7.3.2 Hospitals
    - 7.3.3 Police Stations
    - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [146]:

```
zipcode_data= zipcode_data.drop(columns = 'zipcode_98039')
```

In [147]:

```
zipcode_data.shape
```

```
(19414, 94)
```

In [148]:

```
data.shape
```

```
(19414, 27)
```

I'll be using a combination of data and zipcode_data when appro
identical except one has each zipcode as a separate column and
both of them is useful in making the correlation matrix.

Use csvs when only working on models for speed, and api usage

In [149]:

```
# data = pd.read_csv('Data/Housing_Project_No_Dummies',
#                    index_col="Unnamed: 0").reset_index(dr
# zipcode_data = pd.read_csv('Data/Housing_Project_With_Zip
#                    index_col="Unnamed: 0").reset_index(dr
```

In [150]:

```
# data.to_csv('Housing_Project_No_Dummies')
# zipcode_data.to_csv('Housing_Project_With_Zipcodes')
```

10 Seeing the Correlation

The next step is to avoid multicollinearity by looking at the correla
columns have a correlation of over 70 percent, one will be chose
reasons described below.

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [151]:

```
corr = data.corr()
# building the matrix
trimask = np.triu(np.ones_like(corr, dtype = 'bool'))
# defining the filter to only show half and avoid repeating
plt.figure(figsize=(12,10))
sns.heatmap(corr, mask = trimask | (np.abs(corr) <= 0.70), a
# visualizing the correlation in seaborn and setting it to
# correlation of over 70 percent
```

```
<AxesSubplot:>
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

Choosing which features to drop: sqft_living has an over 70% co
living15. This makes sense, the closest 15 neighbors to a home
size of living space. sqft_living also has a correlation with sqft_ab
makes sense, as these features are very similar. sqft_lot is close
sqft_lot15 for similar reasons. sqft_living also is interestingly corr
Because this feature is so correlated to others, I'll be dropping it.

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

For only keeping one of those variables, Keeping sqft_15 makes about the closest 15 neighbors, not about the property itself, whic better since this metric is more closely related to the theory of mi However, sqft_above is also too closely correlated with grade. Ar or grade, grade is the feature that we have most control of with re property. Therefore I'll be getting rid of sqft_above to keep grade

yr_built, yr_renovated and yr_mod are all correlating with each ot rate which makes sense. Of the three features, I'm keeping yr_re closely related to our interests.

Closest school is almost entirely predicted by closest police static I'll be dropping closest police station as closest school has a mor data associated with it. Downtown is shows multicollinearity with stations so I'll be dropping downtown also.

Additionally, I'll be dropping zipcode since I've made those into du

In [152]:

```
zipcode_data = zipcode_data.drop(columns = ['sqft_living',
                                            'sqft_lot', 'zi
                                            'yr_mod','downt
                                            'closest_police_
data = data.drop(columns = ['sqft_living', 'sqft_living15',
                            'zipcode', 'yr_built', 'yr_mod'
                            'sqft_above', 'closest_police_s

# dropping these features from both data sets
```

Contents

6.1 Comparing Renovated Homes with Unrenovated
6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [153]:

```
corr = data.corr()
trimask = np.triu(np.ones_like(corr, dtype = 'bool'))
plt.figure(figsize=(25,20))
sns.heatmap(corr, mask = trimask | (np.abs(corr) <= 0.5), a

# taking a look to see the results and to make sure there's
# close to 70 percent
```

```
<AxesSubplot:>
```

Contents

- 6.1 Comparing Renovated Homes with Unrenovated
- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [154]:

```
sns.set_style('white')

plt.figure(figsize=(5, 25))
sns.heatmap(abs(zipcode_data.corr())[['price']], annot=True

# now we can see the correlation with price and get a sense
# zipcode data will be
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

	price
zipcode_98177	0.019
zipcode_98178	0.091
zipcode_98188	0.065
zipcode_98198	0.097
zipcode_98199	0.088

In [155]:

```
order_and_values = np.abs(data.corr()['price']).sort_values(
# saving this series of features and correlations as an obj
```

In [156]:

```
order = list(order_and_values.index)
# saving only the index of that series
```

In [157]:

```
order.remove('price')
order.remove('id')
# removing price and id from order
```

In [158]:

```
features_to_remove = []
# making an empty list of features to remove
order = [feature for feature in order if feature not in fea
# making sure the order doesn't include any of those featur
```

11 Building a model

I'm going to build to helper functions here, one to tell me which fe above 0.05 and are therefore unable to reject the null hypothesis doing a good job of predicting.)

I'm also going to build a simple function to tell me the next three r features in the order of features.

Further work:

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

Currently these two helper functions are sketched out but with mo resources, I'd like to make the model building more automated. T for the civic workers tasked with using the model, and also highe these preliminary models however, I wanted to keep a track of wh was including, which is why some of the code is repetitive.

Bugs:

Currently, the list features_to_remove never gets cleared of infor be emptied before each iteration of the model. As things are right needing to remove that many features, so this is working as well But with more time and resources, this functionality would be imp

In [159]:

```
def high_p_vals(model):
    p_vals = model.pvalues
    labels = list(model.pvalues.index)
    p_dict = dict(zip(labels, p_vals))
    for key, value in p_dict.items():
        if value > 0.05:
            features_to_remove.append(key)
        else:
            print(f'{key} has an acceptable p_value')
    return f'drop {features_to_remove} for the next model d
```

In [160]:

```
def next_features(order, features_to_remove):
    print (order[0: 3])
    del order[0: 3]
```

11.1 Baseline Model: Grade, Bathroom

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [161]:

```
next_features(order, features_to_remove)

# finding out what features to include
```

```
['grade', 'bathrooms', 'lat']
```

In [162]:

```
df_baseline = zipcode_data.loc[:, ['price', 'grade', 'bathr
df_baseline.info()

# making a new DF with those features
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 19414 entries, 0 to 21596
Data columns (total 4 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   price      19414 non-null  float64
 1   grade      19414 non-null  int64
 2   bathrooms  19414 non-null  float64
 3   lat        19414 non-null  float64
dtypes: float64(3), int64(1)
memory usage: 758.4 KB
```

In [163]:

```
y = zipcode_data['price']
X = df_baseline.drop(['price'], axis = 1)

# assigning my X and y variables
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [164]:

```
baseline_model = sm.OLS(y, sm.add_constant(X)).fit()
baseline_model.summary()

# building the model and getting a summary
```

OLS Regression Results

Dep. Variable:	price	R-squared:	0.544
Model:	OLS	Adj. R-squared:	0.544
Method:	Least Squares	F-statistic:	7709.
Date:	Tue, 12 Apr 2022	Prob (F-statistic):	0.00
Time:	17:54:48	Log-Likelihood:	-2.6330e+05
No. Observations:	19414	AIC:	5.266e+05
Df Residuals:	19410	BIC:	5.266e+05
Df Model:	3		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-3.169e+07	4.77e+05	-66.366	0.000	-3.26e+07	-3.08e+07
grade	1.414e+05	1660.306	85.164	0.000	1.38e+05	1.45e+05
bathrooms	5.583e+04	2385.026	23.407	0.000	5.12e+04	6.05e+04
lat	6.607e+05	1e+04	65.753	0.000	6.41e+05	6.8e+05

Omnibus:	9369.097	Durbin-Watson:	1.958
Prob(Omnibus):	0.000	Jarque-Bera (JB):	88342.127
Skew:	2.108	Prob(JB):	0.00
Kurtosis:	12.562	Cond. No.	1.69e+04

Notes:

[1] Standard Errors assume that the covariance matrix of the errors is corr

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools


```
[2] The condition number is large, 1.69e+04. This might indicate that there
strong multicollinearity or other numerical problems.
```

Observations: the model is currently predicting at above 50% acc
already a turning point, it's getting it right more than it's getting it
however is way too high for only having three features, this must
model to not be broken.

In [165]:

```
high_p_vals(baseline_model)
```

```
const has an acceptable p_value
grade has an acceptable p_value
bathrooms has an acceptable p_value
lat has an acceptable p_value


'drop [] for the next model due to high p_value(s)'
```

All the P values look good to continue.

In [166]:

```
scaler = StandardScaler()
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [167]:

```
scaled_resid_1 = scaler.fit_transform(baseline_model.resid.
sns.histplot(scaled_resid_1)
```

```
<AxesSubplot:ylabel='Count'>
```

Observations: very long tail on the right side, showing where the
than 2 standard deviations away from the mean.

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [168]:

```
sm.qqplot(baseline_model.resid, dist = stats.norm, line = "
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

Observations: the blue line curves up sharply from the red line at
deviations from the mean, representing the long tail we saw in th

In [169]:

```
sns.scatterplot(baseline_model.predict(sm.add_constant(X)),
plt.axhline(0, color = 'red')
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
6: FutureWarning: Pass the following variables as keyword args: x, y. From
valid positional argument will be `data`, and passing other arguments with
will result in an error or misinterpretation.
  warnings.warn(
```

```
<matplotlib.lines.Line2D at 0x7ff364ab0ac0>
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

Observations: There is a cone shape on the left side of this plot, homoscedasticity. This is to say that there isn't homogeneity of va fix for the next model.

In [170]:

```
sns.scatterplot(y, baseline_model.resid);
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
6: FutureWarning: Pass the following variables as keyword args: x, y. From
valid positional argument will be `data`, and passing other arguments with
will result in an error or misinterpretation.
  warnings.warn(
```

Observations: This is a different way to see the homogeneity of tl this visual doesn't tell us much.

11.2 Second Model: Transforming the P Log

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [171]:

```
sns.histplot(zipcode_data['price'])
```

```
<AxesSubplot:xlabel='price', ylabel='Count'>
```

Price of home does not have a normal distribution in this data set more lower priced homes and a much longer tail on the right han the lack of upward limit on home price.

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [172]:

```
sns.histplot(np.log(zipcode_data['price']))
```

```
<AxesSubplot:xlabel='price', ylabel='Count'>
```

The log of the price however, does have a much more normal dis
the log of the price for all future models.

NOTE: I'll now be interpreting the price by percentage instea
amount.

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [173]:

```
zipcode_data['price_log'] = np.log(zipcode_data['price'])

# making a new column in the data for the log of the price
```

In [174]:

```
price_log = zipcode_data['price_log']
```

In [175]:

```
features_2 = zipcode_data.loc[:,['price_log', 'grade', 'batl
features_2.info()

# making the dataframe for the second model
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 19414 entries, 0 to 21596
Data columns (total 4 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   price_log  19414 non-null  float64
 1   grade      19414 non-null  int64
 2   bathrooms  19414 non-null  float64
 3   lat        19414 non-null  float64
dtypes: float64(3), int64(1)
memory usage: 758.4 KB
```

In [176]:

```
y = features_2['price_log']
X = features_2.drop(['price_log'], axis = 1)

# labeling the axes
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [177]:

```
model_2 = sm.OLS(y, sm.add_constant(X)).fit()
model_2.summary()

# making the model
```

OLS Regression Results

Dep. Variable:	price_log	R-squared:	0.631
Model:	OLS	Adj. R-squared:	0.630
Method:	Least Squares	F-statistic:	1.104e+04
Date:	Tue, 12 Apr 2022	Prob (F-statistic):	0.00
Time:	17:54:49	Log-Likelihood:	-3753.2
No. Observations:	19414	AIC:	7514.
Df Residuals:	19410	BIC:	7546.
Df Model:	3		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-61.9788	0.746	-83.051	0.000	-63.442	-60.516
grade	0.2313	0.003	89.137	0.000	0.226	0.236
bathrooms	0.1188	0.004	31.870	0.000	0.111	0.126
lat	1.5491	0.016	98.632	0.000	1.518	1.580

Omnibus:	441.832	Durbin-Watson:	1.973
Prob(Omnibus):	0.000	Jarque-Bera (JB):	590.661
Skew:	0.279	Prob(JB):	5.49e-129
Kurtosis:	3.647	Cond. No.	1.69e+04

Notes:

[1] Standard Errors assume that the covariance matrix of the errors is corr

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools


```
[2] The condition number is large, 1.69e+04. This might indicate that there
strong multicollinearity or other numerical problems.
```

Observations: the kurtosis is much lower now, and we've improve
model by about 10%.

In [178]:

```
scaled_resid_2 = scaler.fit_transform(model_2.resid.values.
sns.histplot(scaled_resid_2)
```

```
<AxesSubplot:ylabel='Count'>
```

Observations: these residuals fit a normal distribution much bette

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fc
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Loc
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [179]:

```
sm.qqplot(model_2.resid, dist = stats.norm, line = "45", fi
```

Contents

6.1 Comparing Renovated Homes with Unrenovated
6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

The blue line is much closer to the red line, representing the mor
variance.

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [180]:

```
sns.scatterplot(model_2.predict(sm.add_constant(X)), model_
plt.axhline(0, color = 'red')
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
6: FutureWarning: Pass the following variables as keyword args: x, y. From
valid positional argument will be `data`, and passing other arguments with
will result in an error or misinterpretation.
  warnings.warn(
```

```
<matplotlib.lines.Line2D at 0x7ff364b2ab20>
```

Observations: the cone shape is gone now on the left side, repre
a pattern that the model isn't picking up.

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [181]:

```
sns.scatterplot(y, model_2.resid);
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
6: FutureWarning: Pass the following variables as keyword args: x, y. From
valid positional argument will be `data`, and passing other arguments with
will result in an error or misinterpretation.
  warnings.warn(
```

Observations: the points are now a bit more scattered, which is w

12 Improving the Model with Ad Features

12.1 Third Model: Bedrooms, Floors, Cl Distance

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [182]:

```
next_features(order, features_to_remove)
```

```
['bedrooms', 'floors', 'closest_school_distance']
```

In [183]:

```
features_3 = ['bedrooms', 'floors', 'closest_school_distanc
```

In [184]:

```
X = pd.concat([X, zipcode_data[features_3]], axis = 1)
```

In [185]:

```
X.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 19414 entries, 0 to 21596
Data columns (total 6 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   grade                    19414 non-null  int64
 1   bathrooms                19414 non-null  float64
 2   lat                      19414 non-null  float64
 3   bedrooms                 19414 non-null  int64
 4   floors                   19414 non-null  float64
 5   closest_school_distance  19414 non-null  float64
dtypes: float64(4), int64(2)
memory usage: 1.0 MB
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [186]:

```
model_3 = sm.OLS(y, sm.add_constant(X)).fit()
model_3.summary()
```

OLS Regression Results

Dep. Variable:	price_log	R-squared:	0.653
Model:	OLS	Adj. R-squared:	0.653
Method:	Least Squares	F-statistic:	6079.
Date:	Tue, 12 Apr 2022	Prob (F-statistic):	0.00
Time:	17:54:50	Log-Likelihood:	-3152.5
No. Observations:	19414	AIC:	6319.
Df Residuals:	19407	BIC:	6374.
Df Model:	6		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-50.0830	0.872	-57.451	0.000	-51.792	-48.374
grade	0.2437	0.003	93.661	0.000	0.239	0.249
bathrooms	0.1078	0.004	25.545	0.000	0.100	0.116
lat	1.2971	0.018	70.756	0.000	1.261	1.333
bedrooms	0.0554	0.003	20.328	0.000	0.050	0.061
floors	-0.0427	0.005	-9.442	0.000	-0.052	-0.034
closest_school_distance	-0.0127	0.000	-26.354	0.000	-0.014	-0.012

Omnibus:	292.743	Durbin-Watson:	1.979
Prob(Omnibus):	0.000	Jarque-Bera (JB):	433.905
Skew:	0.172	Prob(JB):	6.01e-95
Kurtosis:	3.647	Cond. No.	2.06e+04

Contents

- 6.1 Comparing Renovated Homes with Unrenovated
- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools


```
Notes:
[1] Standard Errors assume that the covariance matrix of the errors is corr
[2] The condition number is large, 2.06e+04. This might indicate that there
strong multicollinearity or other numerical problems.
```

Observations: adding these three features didn't make the accura
3%.

FURTHER WORK:

I have not looked too deeply into floors as I have with some other
model, it's predicting that as floors decrease, the price rises. This
intuitive, and bears further investigation.

In [187]:

```
high_p_vals(model_3)
```

```
const has an acceptable p_value
grade has an acceptable p_value
bathrooms has an acceptable p_value
lat has an acceptable p_value
bedrooms has an acceptable p_value
floors has an acceptable p_value
closest_school_distance has an acceptable p_value


'drop [] for the next model due to high p_value(s)'
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [188]:

```
scaled_resid_3 = scaler.fit_transform(model_3.resid.values.
sns.histplot(scaled_resid_3)
```

```
<AxesSubplot:ylabel='Count'>
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [189]:

```
sm.qqplot(model_3.resid, dist = stats.norm, line = "45", fi
```

Contents

6.1 Comparing Renovated Homes with Unrenovated
6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [190]:

```
sns.scatterplot(model_3.predict(sm.add_constant(X)), model_
plt.axhline(0, color = 'red')
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
6: FutureWarning: Pass the following variables as keyword args: x, y. From
valid positional argument will be `data`, and passing other arguments with
will result in an error or misinterpretation.
  warnings.warn(
```

```
<matplotlib.lines.Line2D at 0x7ff36599f490>
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [191]:

```
sns.scatterplot(y, model_3.resid);
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
6: FutureWarning: Pass the following variables as keyword args: x, y. From
valid positional argument will be `data`, and passing other arguments with
will result in an error or misinterpretation.
  warnings.warn(
```

Observations: these plots show that the model is functioning prop
the lack of cone shapes in the scatterplots and the blue line being

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

on the QQ plot.

12.2 Fourth Model: Square Foot Basem Hospital Distance and Year Renovated

In [192]:

```
next_features(order, features_to_remove)
```

```
['sqft_basement', 'closest_hospital_distance', 'yr_renovated']
```

In [193]:

```
features_4 = ['sqft_basement', 'closest_hospital_distance',
```

In [194]:

```
X = pd.concat([X, zipcode_data[features_4]], axis = 1)
X.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 19414 entries, 0 to 21596
Data columns (total 9 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   grade                      19414 non-null  int64
 1   bathrooms                  19414 non-null  float64
 2   lat                        19414 non-null  float64
 3   bedrooms                   19414 non-null  int64
 4   floors                     19414 non-null  float64
 5   closest_school_distance    19414 non-null  float64
 6   sqft_basement              19414 non-null  float64
 7   closest_hospital_distance  19414 non-null  float64
 8   yr_renovated               19414 non-null  float64
dtypes: float64(7), int64(2)
memory usage: 1.5 MB
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [195]:

```
model_4 = sm.OLS(y, sm.add_constant(X)).fit()
model_4.summary()
```

OLS Regression Results

Dep. Variable:	price_log	R-squared:	0.661
Model:	OLS	Adj. R-squared:	0.661
Method:	Least Squares	F-statistic:	4213.
Date:	Tue, 12 Apr 2022	Prob (F-statistic):	0.00
Time:	17:54:50	Log-Likelihood:	-2903.6
No. Observations:	19414	AIC:	5827.
Df Residuals:	19404	BIC:	5906.
Df Model:	9		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-51.2004	0.863	-59.347	0.000	-52.891	-49.509
grade	0.3416	0.003	95.185	0.000	0.340	0.350

In [196]:

```
high_p_vals(model_4)
```

```
const has an acceptable p_value
grade has an acceptable p_value
bathrooms has an acceptable p_value
lat has an acceptable p_value
bedrooms has an acceptable p_value
floors has an acceptable p_value
closest_school_distance has an acceptable p_value
sqft_basement has an acceptable p_value
closest_hospital_distance has an acceptable p_value
yr_renovated has an acceptable p_value


'drop [] for the next model due to high p_value(s)'
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [197]:

```
scaled_resid_4 = scaler.fit_transform(model_4.resid.values.
sns.histplot(scaled_resid_4)
```

```
<AxesSubplot:ylabel='Count'>
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [198]:

```
sm.qqplot(model_4.resid, dist = stats.norm, line = "45", fi
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [199]:

```
sns.scatterplot(model_4.predict(sm.add_constant(X)), model_
plt.axhline(0, color = 'red')
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
6: FutureWarning: Pass the following variables as keyword args: x, y. From
valid positional argument will be `data`, and passing other arguments with
will result in an error or misinterpretation.
  warnings.warn(
```

```
<matplotlib.lines.Line2D at 0x7ff365951f10>
```

Observations: now there is a slight cone shape on the right side. will not do anything to fix it for the next model.

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [200]:

```
sns.scatterplot(y, model_4.resid);
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
6: FutureWarning: Pass the following variables as keyword args: x, y. From
valid positional argument will be `data`, and passing other arguments with
will result in an error or misinterpretation.
  warnings.warn(
```

Observations:

12.3 Fifth Model: Square Foot Lot of Cl Neighbors, Longitude and Age of Home

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [201]:

```

next_features(order, features_to_remove)
```

```
['sqft_lot15', 'long', 'age_of_home']
```

In [202]:

```
features_5 = ['sqft_lot15', 'long', 'age_of_home']
```

In [203]:

```
X = pd.concat([X, zipcode_data[features_5]], axis = 1)
X.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 19414 entries, 0 to 21596
Data columns (total 12 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   grade                      19414 non-null  int64
 1   bathrooms                  19414 non-null  float64
 2   lat                        19414 non-null  float64
 3   bedrooms                   19414 non-null  int64
 4   floors                     19414 non-null  float64
 5   closest_school_distance    19414 non-null  float64
 6   sqft_basement              19414 non-null  float64
 7   closest_hospital_distance  19414 non-null  float64
 8   yr_renovated               19414 non-null  float64
 9   sqft_lot15                 19414 non-null  int64
 10  long                       19414 non-null  float64
 11  age_of_home                19414 non-null  int64
dtypes: float64(8), int64(4)
memory usage: 1.9 MB
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [204]:

```
model_5 = sm.OLS(y, sm.add_constant(X)).fit()
model_5.summary()
```

OLS Regression Results

Dep. Variable:	price_log	R-squared:	0.706
Model:	OLS	Adj. R-squared:	0.706
Method:	Least Squares	F-statistic:	3878.
Date:	Tue, 12 Apr 2022	Prob (F-statistic):	0.00
Time:	17:54:51	Log-Likelihood:	-1542.8
No. Observations:	19414	AIC:	3112.
Df Residuals:	19401	BIC:	3214.
Df Model:	12		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	46.0883	3.775	12.208	0.000	38.689	53.488
grade	0.2543	0.002	103.909	0.000	0.250	0.259
bathrooms	0.1398	0.004	32.812	0.000	0.131	0.148
lat	1.0569	0.019	54.354	0.000	1.019	1.095
bedrooms	0.0291	0.003	11.393	0.000	0.024	0.034
floors	0.0551	0.005	11.450	0.000	0.046	0.065
closest_school_distance	-0.0140	0.001	-18.848	0.000	-0.015	-0.013
sqft_basement	0.0625	0.005	13.702	0.000	0.054	0.071
closest_hospital_distance	-0.0150	0.001	-13.945	0.000	-0.017	-0.013
yr_renovated	2.387e-05	5.69e-06	4.197	0.000	1.27e-05	3.5e-05
sqft_lot15	1.068e-06	1.66e-07	6.441	0.000	7.43e-07	1.39e-06
long	0.6962	0.026	26.315	0.000	0.644	0.748
age_of_home	0.0045	9.51e-05	47.781	0.000	0.004	0.005

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

Omnibus:	363.810	Durbin-Watson:	1.991
Prob(Omnibus):	0.000	Jarque-Bera (JB):	726.252
Skew:	0.083	Prob(JB):	1.98e-158
Kurtosis:	3.933	Cond. No.	3.11e+07

```
Notes:
[1] Standard Errors assume that the covariance matrix of the errors is corr
[2] The condition number is large, 3.11e+07. This might indicate that there
strong multicollinearity or other numerical problems.
```

In [205]:

```
high_p_vals(model_5)
```

```
const has an acceptable p_value
grade has an acceptable p_value
bathrooms has an acceptable p_value
lat has an acceptable p_value
bedrooms has an acceptable p_value
floors has an acceptable p_value
closest_school_distance has an acceptable p_value
sqft_basement has an acceptable p_value
closest_hospital_distance has an acceptable p_value
yr_renovated has an acceptable p_value
sqft_lot15 has an acceptable p_value
long has an acceptable p_value
age_of_home has an acceptable p_value


'drop [] for the next model due to high p_value(s)'
```

Contents

6.1 Comparing Renovated Homes with Unrenovated
6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [206]:

```
scaled_resid_5 = scaler.fit_transform(model_5.resid.values.
sns.histplot(scaled_resid_5)
```

```
<AxesSubplot:ylabel='Count'>
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fc
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Loc
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [207]:

```
sm.qqplot(model_5.resid, dist = stats.norm, line = "45", fi
```

Contents

- 6.1 Comparing Renovated Homes with Unrenovated
- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [208]:

```
sns.scatterplot(model_5.predict(sm.add_constant(X)), model_
plt.axhline(0, color = 'red')
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
6: FutureWarning: Pass the following variables as keyword args: x, y. From
valid positional argument will be `data`, and passing other arguments with
will result in an error or misinterpretation.
  warnings.warn(
```

```
<matplotlib.lines.Line2D at 0x7ff364b3c6a0>
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fc
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Loc
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [209]:

```
sns.scatterplot(y, model_5.resid)
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
6: FutureWarning: Pass the following variables as keyword args: x, y. From
valid positional argument will be `data`, and passing other arguments with
will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='price_log'>
```

12.4 Sixth Model: Condition, Waterfron View

Contents

- 6.1 Comparing Renovated Homes with Unrenovated
- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [210]:

```
next_features(order, features_to_remove)
```

```
['condition', 'waterfront', 'view']
```

In [211]:

```
features_6 = ['condition', 'waterfront', 'view']
```

In [212]:

```
X = pd.concat([X, zipcode_data[features_6]], axis = 1)
X.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 19414 entries, 0 to 21596
Data columns (total 15 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   grade                      19414 non-null  int64
 1   bathrooms                  19414 non-null  float64
 2   lat                        19414 non-null  float64
 3   bedrooms                   19414 non-null  int64
 4   floors                     19414 non-null  float64
 5   closest_school_distance    19414 non-null  float64
 6   sqft_basement              19414 non-null  float64
 7   closest_hospital_distance  19414 non-null  float64
 8   yr_renovated               19414 non-null  float64
 9   sqft_lot15                 19414 non-null  int64
 10  long                       19414 non-null  float64
 11  age_of_home                19414 non-null  int64
 12  condition                  19414 non-null  int64
 13  waterfront                 19414 non-null  int64
 14  view                       19414 non-null  float64
dtypes: float64(9), int64(6)
memory usage: 2.4 MB
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [213]:

```
model_6 = sm.OLS(y, sm.add_constant(X)).fit()
model_6.summary()
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
inear_model.py:1860: RuntimeWarning: divide by zero encountered in double_
  return np.sqrt(eigvals[0]/eigvals[-1])
```

OLS Regression Results

Dep. Variable:	price_log	R-squared:	0.711
Model:	OLS	Adj. R-squared:	0.711
Method:	Least Squares	F-statistic:	3675.
Date:	Tue, 12 Apr 2022	Prob (F-statistic):	0.00
Time:	17:54:51	Log-Likelihood:	-1362.2
No. Observations:	19414	AIC:	2752.
Df Residuals:	19400	BIC:	2863.
Df Model:	13		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	43.4383	3.743	11.606	0.000	36.102	50.775
grade	0.2550	0.002	105.124	0.000	0.250	0.260
bathrooms	0.1329	0.004	31.358	0.000	0.125	0.141
lat	1.0735	0.019	55.665	0.000	1.036	1.111
bedrooms	0.0275	0.003	10.853	0.000	0.023	0.032
floors	0.0643	0.005	13.422	0.000	0.055	0.074
closest_school_distance	-0.0139	0.001	-18.936	0.000	-0.015	-0.012
sqft_basement	0.0605	0.005	13.389	0.000	0.052	0.069
closest_hospital_distance	-0.0151	0.001	-14.177	0.000	-0.017	-0.013
yr_renovated	4.025e-05	5.7e-06	7.062	0.000	2.91e-05	5.14e-05

Contents

- 6.1 Comparing Renovated Homes with Unrenovated
- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

sqft_lot15	9.948e-07	1.64e-07	6.057	0.000	6.73e-07	1.32e-06
long	0.6819	0.026	26.007	0.000	0.631	0.733
age_of_home	0.0040	9.89e-05	40.225	0.000	0.004	0.004
condition	0.0613	0.003	19.090	0.000	0.055	0.068
waterfront	0	0	nan	nan	0	0
view	0	0	nan	nan	0	0

Omnibus:	365.540	Durbin-Watson:	1.992
Prob(Omnibus):	0.000	Jarque-Bera (JB):	681.974
Skew:	0.124	Prob(JB):	8.15e-149
Kurtosis:	3.884	Cond. No.	inf

```
Notes:
[1] Standard Errors assume that the covariance matrix of the errors is corr
[2] The smallest eigenvalue is 0. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [214]:

```
high_p_vals(model_6)
```

```
const has an acceptable p_value
grade has an acceptable p_value
bathrooms has an acceptable p_value
lat has an acceptable p_value
bedrooms has an acceptable p_value
floors has an acceptable p_value
closest_school_distance has an acceptable p_value
sqft_basement has an acceptable p_value
closest_hospital_distance has an acceptable p_value
yr_renovated has an acceptable p_value
sqft_lot15 has an acceptable p_value
long has an acceptable p_value
age_of_home has an acceptable p_value
condition has an acceptable p_value
waterfront has an acceptable p_value
view has an acceptable p_value


'drop [] for the next model due to high p_value(s)'
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [215]:

```
scaled_resid_6 = scaler.fit_transform(model_6.resid.values.
sns.histplot(scaled_resid_6)
```

```
<AxesSubplot:ylabel='Count'>
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [216]:

```
sm.qqplot(model_6.resid, dist = stats.norm, line = "45", fi
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [217]:

```
sns.scatterplot(model_6.predict(sm.add_constant(X)), model_
plt.axhline(0, color = 'red')
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
6: FutureWarning: Pass the following variables as keyword args: x, y. From
valid positional argument will be `data`, and passing other arguments with
will result in an error or misinterpretation.
  warnings.warn(
```

```
<matplotlib.lines.Line2D at 0x7ff34422c0a0>
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [218]:

```
sns.scatterplot(y, model_6.resid)
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
6: FutureWarning: Pass the following variables as keyword args: x, y. From
valid positional argument will be `data`, and passing other arguments with
will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='price_log'>
```

12.5 Seventh Model: Adding Zipcodes

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [219]:

```
zipcode_data.columns
```

```
Index(['id', 'date', 'price', 'bedrooms', 'bathrooms', 'floors', 'waterfro
       'view', 'condition', 'grade', 'sqft_basement', 'yr_renovated', 'lat
       'long', 'sqft_lot15', 'age_of_home', 'closest_hospital_distance',
       'closest_school_distance', 'zipcode_98001', 'zipcode_98002',
       'zipcode_98003', 'zipcode_98004', 'zipcode_98005', 'zipcode_98006',
       'zipcode_98007', 'zipcode_98008', 'zipcode_98010', 'zipcode_98011',
       'zipcode_98014', 'zipcode_98019', 'zipcode_98023', 'zipcode_98024',
       'zipcode_98027', 'zipcode_98028', 'zipcode_98029', 'zipcode_98030',
       'zipcode_98031', 'zipcode_98032', 'zipcode_98033', 'zipcode_98034',
       'zipcode_98038', 'zipcode_98040', 'zipcode_98042', 'zipcode_98052',
       'zipcode_98053', 'zipcode_98055', 'zipcode_98056', 'zipcode_98058',
       'zipcode_98059', 'zipcode_98065', 'zipcode_98070', 'zipcode_98072',
       'zipcode_98074', 'zipcode_98075', 'zipcode_98077', 'zipcode_98092',
       'zipcode_98102', 'zipcode_98103', 'zipcode_98105', 'zipcode_98106',
       'zipcode_98107', 'zipcode_98108', 'zipcode_98109', 'zipcode_98112',
       'zipcode_98115', 'zipcode_98116', 'zipcode_98117', 'zipcode_98118',
       'zipcode_98119', 'zipcode_98122', 'zipcode_98125', 'zipcode_98126',
       'zipcode_98133', 'zipcode_98136', 'zipcode_98144', 'zipcode_98146',
       'zipcode_98148', 'zipcode_98155', 'zipcode_98166', 'zipcode_98168',
       'zipcode_98177', 'zipcode_98178', 'zipcode_98188', 'zipcode_98198',
       'zipcode_98199', 'price_log'],
      dtype='object')
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
  - 7.1 Getting Infrastucture Data
    - 7.1.1 Downtown
    - 7.1.2 Hospitals
    - 7.1.3 Police Stations
    - 7.1.4 Schools
  - 7.2 Finding Distance for Infrastructure
  - 7.3 Exploring Distance Data
    - 7.3.1 Downtown
    - 7.3.2 Hospitals
    - 7.3.3 Police Stations
    - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [220]:

```
features_7 = ['zipcode_98001',
        'zipcode_98002', 'zipcode_98003', 'zipcode_98004', '
        'zipcode_98006', 'zipcode_98007', 'zipcode_98008', '
        'zipcode_98011', 'zipcode_98014', 'zipcode_98019', '
        'zipcode_98024', 'zipcode_98027', 'zipcode_98028', '
        'zipcode_98030', 'zipcode_98031', 'zipcode_98032', '
        'zipcode_98034', 'zipcode_98038', 'zipcode_98040', '
        'zipcode_98052', 'zipcode_98053', 'zipcode_98055', '
        'zipcode_98058', 'zipcode_98059', 'zipcode_98065', '
        'zipcode_98072', 'zipcode_98074', 'zipcode_98075', '
        'zipcode_98092', 'zipcode_98102', 'zipcode_98103', '
        'zipcode_98106', 'zipcode_98107', 'zipcode_98108', '
        'zipcode_98112', 'zipcode_98115', 'zipcode_98116', '
        'zipcode_98118', 'zipcode_98119', 'zipcode_98122', '
        'zipcode_98126', 'zipcode_98133', 'zipcode_98136', '
        'zipcode_98146', 'zipcode_98148', 'zipcode_98155', '
        'zipcode_98168', 'zipcode_98177', 'zipcode_98178', '
        'zipcode_98198', 'zipcode_98199']
```

Contents

6.1 Comparing Renovated Homes with Unrenovated
6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [221]:

```
X = pd.concat([X, zipcode_data[features_7]], axis = 1)
X.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 19414 entries, 0 to 21596
Data columns (total 82 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   grade                      19414 non-null  int64
 1   bathrooms                  19414 non-null  float64
 2   lat                        19414 non-null  float64
 3   bedrooms                   19414 non-null  int64
 4   floors                     19414 non-null  float64
 5   closest_school_distance    19414 non-null  float64
 6   sqft_basement              19414 non-null  float64
 7   closest_hospital_distance  19414 non-null  float64
 8   yr_renovated               19414 non-null  float64
 9   sqft_lot15                 19414 non-null  int64
 10  long                       19414 non-null  float64
 11  age_of_home                19414 non-null  int64
 12  condition                  19414 non-null  int64
 13  waterfront                 19414 non-null  int64
 14  view                       19414 non-null  float64
 15  zipcode_98001              19414 non-null  uint8
 16  zipcode_98002              19414 non-null  uint8
 17  zipcode_98003              19414 non-null  uint8
 18  zipcode_98004              19414 non-null  uint8
 19  zipcode_98005              19414 non-null  uint8
 20  zipcode_98006              19414 non-null  uint8
 21  zipcode_98007              19414 non-null  uint8
 22  zipcode_98008              19414 non-null  uint8
 23  zipcode_98010              19414 non-null  uint8
 24  zipcode_98011              19414 non-null  uint8
 25  zipcode_98014              19414 non-null  uint8
 26  zipcode_98019              19414 non-null  uint8
 27  zipcode_98023              19414 non-null  uint8
 28  zipcode_98024              19414 non-null  uint8
 29  zipcode_98027              19414 non-null  uint8
 30  zipcode_98028              19414 non-null  uint8
 31  zipcode_98029              19414 non-null  uint8
 32  zipcode_98030              19414 non-null  uint8
```

Contents

6.1 Comparing Renovated Homes with Unrenovated
6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools


```
 33  zipcode_98031              19414 non-null  uint8
 34  zipcode_98032              19414 non-null  uint8
 35  zipcode_98033              19414 non-null  uint8
 36  zipcode_98034              19414 non-null  uint8
 37  zipcode_98038              19414 non-null  uint8
 38  zipcode_98040              19414 non-null  uint8
 39  zipcode_98042              19414 non-null  uint8
 40  zipcode_98052              19414 non-null  uint8
 41  zipcode_98053              19414 non-null  uint8
 42  zipcode_98055              19414 non-null  uint8
 43  zipcode_98056              19414 non-null  uint8
 44  zipcode_98058              19414 non-null  uint8
 45  zipcode_98059              19414 non-null  uint8
 46  zipcode_98065              19414 non-null  uint8
 47  zipcode_98070              19414 non-null  uint8
 48  zipcode_98072              19414 non-null  uint8
 49  zipcode_98074              19414 non-null  uint8
 50  zipcode_98075              19414 non-null  uint8
 51  zipcode_98077              19414 non-null  uint8
 52  zipcode_98092              19414 non-null  uint8
 53  zipcode_98102              19414 non-null  uint8
 54  zipcode_98103              19414 non-null  uint8
 55  zipcode_98105              19414 non-null  uint8
 56  zipcode_98106              19414 non-null  uint8
 57  zipcode_98107              19414 non-null  uint8
 58  zipcode_98108              19414 non-null  uint8
 59  zipcode_98109              19414 non-null  uint8
 60  zipcode_98112              19414 non-null  uint8
 61  zipcode_98115              19414 non-null  uint8
 62  zipcode_98116              19414 non-null  uint8
 63  zipcode_98117              19414 non-null  uint8
 64  zipcode_98118              19414 non-null  uint8
 65  zipcode_98119              19414 non-null  uint8
 66  zipcode_98122              19414 non-null  uint8
 67  zipcode_98125              19414 non-null  uint8
 68  zipcode_98126              19414 non-null  uint8
 69  zipcode_98133              19414 non-null  uint8
 70  zipcode_98136              19414 non-null  uint8
 71  zipcode_98144              19414 non-null  uint8
 72  zipcode_98146              19414 non-null  uint8
 73  zipcode_98148              19414 non-null  uint8
 74  zipcode_98155              19414 non-null  uint8
 75  zipcode_98166              19414 non-null  uint8
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools


```
 76  zipcode_98168          19414 non-null  uint8
 77  zipcode_98177          19414 non-null  uint8
 78  zipcode_98178          19414 non-null  uint8
 79  zipcode_98188          19414 non-null  uint8
 80  zipcode_98198          19414 non-null  uint8
 81  zipcode_98199          19414 non-null  uint8
dtypes: float64(9), int64(6), uint8(67)
memory usage: 3.6 MB
```

In [222]:

```
model_7 = sm.OLS(y, sm.add_constant(X)).fit()
model_7.summary()
```

OLS Regression Results

Dep. Variable:	price_log	R-squared:	0.824
Model:	OLS	Adj. R-squared:	0.824
Method:	Least Squares	F-statistic:	1135.
Date:	Tue, 12 Apr 2022	Prob (F-statistic):	0.00
Time:	17:54:52	Log-Likelihood:	3469.4
No. Observations:	19414	AIC:	-6777.
Df Residuals:	19333	BIC:	-6139.
Df Model:	80		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	42.7426	14.032	3.046	0.002	15.238	70.247
grade	0.1961	0.002	95.269	0.000	0.192	0.200

Contents

6.1 Comparing Renovated Homes with Unrenovated
6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [223]:

```
high_p_vals(model_7)
```

```
const has an acceptable p_value
grade has an acceptable p_value
bathrooms has an acceptable p_value
lat has an acceptable p_value
bedrooms has an acceptable p_value
floors has an acceptable p_value
closest_school_distance has an acceptable p_value
sqft_basement has an acceptable p_value
yr_renovated has an acceptable p_value
sqft_lot15 has an acceptable p_value
long has an acceptable p_value
age_of_home has an acceptable p_value
condition has an acceptable p_value
waterfront has an acceptable p_value
view has an acceptable p_value
zipcode_98001 has an acceptable p_value
zipcode_98002 has an acceptable p_value
zipcode_98003 has an acceptable p_value
zipcode_98004 has an acceptable p_value
zipcode_98005 has an acceptable p_value
```

Adding in zipcodes has made the latitude and the closest hospita
predictors and I'll be removing those features for the next iteratio

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [224]:

```
scaled_resid_7 = scaler.fit_transform(model_7.resid.values.
sns.histplot(scaled_resid_7)
```

```
<AxesSubplot:ylabel='Count'>
```

Contents

6.1 Comparing Renovated Homes with Unrenovated
6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [225]:

```
sm.qqplot(model_7.resid, dist = stats.norm, line = "45", fi
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

Observations: zipcode data has moved the model into performing
are now more residuals in the -4 through -2 and 2 through 4 dista
mean. Basically, the model is predicting less correctly where the
on the red line.

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [226]:

```
sns.scatterplot(model_7.predict(sm.add_constant(X)), model_
plt.axhline(0, color = 'red')
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
6: FutureWarning: Pass the following variables as keyword args: x, y. From
valid positional argument will be `data`, and passing other arguments with
will result in an error or misinterpretation.
  warnings.warn(
```

```
<matplotlib.lines.Line2D at 0x7ff343f8cfa0>
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [227]:

```
sns.scatterplot(y, model_7.resid)
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
6: FutureWarning: Pass the following variables as keyword args: x, y. From
valid positional argument will be `data`, and passing other arguments with
will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='price_log'>
```

12.6 Eighth Model: Latitude and Close: Distance

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [228]:
```
X = X.drop(columns = ['lat', 'closest_hospital_distance'])
```

In [229]:
```
features_8 = [col for col in X.columns]
```

In [230]:
```
X = zipcode_data.loc[:, features_8]
```

In [231]:
```
X.info()
```

```
 63  zipcode_98119            19414 non-null  uint8
 64  zipcode_98122            19414 non-null  uint8
 65  zipcode_98125            19414 non-null  uint8
 66  zipcode_98126            19414 non-null  uint8
 67  zipcode_98133            19414 non-null  uint8
 68  zipcode_98136            19414 non-null  uint8
 69  zipcode_98144            19414 non-null  uint8
 70  zipcode_98146            19414 non-null  uint8
 71  zipcode_98148            19414 non-null  uint8
 72  zipcode_98155            19414 non-null  uint8
 73  zipcode_98166            19414 non-null  uint8
 74  zipcode_98168            19414 non-null  uint8
 75  zipcode_98177            19414 non-null  uint8
 76  zipcode_98178            19414 non-null  uint8
 77  zipcode_98188            19414 non-null  uint8
 78  zipcode_98198            19414 non-null  uint8
 79  zipcode_98199            19414 non-null  uint8
dtypes: float64(7), int64(6), uint8(67)
memory usage: 3.3 MB
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [232]:

```
model_8 = sm.OLS(y, sm.add_constant(X)).fit()
model_8.summary()
```

OLS Regression Results

Dep. Variable:	price_log	R-squared:	0.824
Model:	OLS	Adj. R-squared:	0.824
Method:	Least Squares	F-statistic:	1163.
Date:	Tue, 12 Apr 2022	Prob (F-statistic):	0.00
Time:	17:54:53	Log-Likelihood:	3465.5
No. Observations:	19414	AIC:	-6773.
Df Residuals:	19335	BIC:	-6151.
Df Model:	78		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	57.6242	12.627	4.564	0.000	32.875	82.373

Contents

6.1 Comparing Renovated Homes with Unrenovated
6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [233]:

```
high_p_vals(model_8)
```

```
const has an acceptable p_value
grade has an acceptable p_value
bathrooms has an acceptable p_value
bedrooms has an acceptable p_value
floors has an acceptable p_value
closest_school_distance has an acceptable p_value
sqft_basement has an acceptable p_value
yr_renovated has an acceptable p_value
sqft_lot15 has an acceptable p_value
long has an acceptable p_value
age_of_home has an acceptable p_value
condition has an acceptable p_value
waterfront has an acceptable p_value
view has an acceptable p_value
zipcode_98001 has an acceptable p_value
zipcode_98002 has an acceptable p_value
zipcode_98003 has an acceptable p_value
zipcode_98004 has an acceptable p_value
zipcode_98005 has an acceptable p_value
zipcode_98006 has an acceptable p_value
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [234]:

```
scaled_resid_8 = scaler.fit_transform(model_8.resid.values.
sns.histplot(scaled_resid_8)
```

```
<AxesSubplot:ylabel='Count'>
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [235]:

```
sm.qqplot(model_8.resid, dist = stats.norm, line = "45", fi
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [236]:

```
sns.scatterplot(model_8.predict(sm.add_constant(X)), model_8
plt.axhline(0, color = 'red')
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
6: FutureWarning: Pass the following variables as keyword args: x, y. From
valid positional argument will be `data`, and passing other arguments with
will result in an error or misinterpretation.
  warnings.warn(
```

```
<matplotlib.lines.Line2D at 0x7ff344b0aaf0>
```

Contents

6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fo
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Log
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [237]:

```
sns.scatterplot(y, model_8.resid)
```

```
/Users/b0ihazard/opt/anaconda3/envs/learn-env/lib/python3.8/site-packages/
6: FutureWarning: Pass the following variables as keyword args: x, y. From
valid positional argument will be `data`, and passing other arguments with
will result in an error or misinterpretation.
  warnings.warn(
```

```
<AxesSubplot:xlabel='price_log'>
```

13 Conclusion

I'm going to generate a report of the findings of the model's coeff

Contents

6.1 Comparing Renovated Homes with Unrenovated
6.2 Visualizing the Comparison
6.3 Home History
6.3.1 Age of Home
6.3.2 Year Modified
7 Distance Data
7.1 Getting Infrastucture Data
7.1.1 Downtown
7.1.2 Hospitals
7.1.3 Police Stations
7.1.4 Schools
7.2 Finding Distance for Infrastructure
7.3 Exploring Distance Data
7.3.1 Downtown
7.3.2 Hospitals
7.3.3 Police Stations
7.3.4 Schools
8 Preparing for Modeling
8.0.1 Dropping View and Waterfront Properties fc
9 Zipcode
10 Seeing the Correlation
11 Building a model
11.1 Baseline Model: Grade, Bathrooms, Lat
11.2 Second Model: Transforming the Price with Loc
12 Improving the Model with Adding Features
12.1 Third Model: Bedrooms, Floors, Closest School
12.2 Fourth Model: Square Foot Basement, Closest
12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
12.4 Sixth Model: Condition, Waterfront and View
12.5 Seventh Model: Adding Zipcodes
12.6 Eighth Model: Latitude and Closest Hospital D
13 Conclusion
14 Recommendations:
14.0.1 Visualizing Schools

In [238]:

```
pd.set_option('display.max_rows', None)

coeff_s = model_8.params
coeff_s = coeff_s.drop('const')

labels = coeff_s.index
coefficiants = dict(zip(labels, coeff_s))

# grabbing the coefficiants from the stats models reports
```

By generating a report of the findings, should this model be used
future, understanding the model will become much more approac
friendly. This is good news for the civic workers that will be using

Contents

- 6.1 Comparing Renovated Homes with Unrenovated
- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Loc
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [239]:

```
for x, y in coefficiants.items():
    y = y * 100
    if x.startswith('view') or x.startswith('waterfront'):
        print (f' If {x} was a feature of a property, the p
    elif not x.startswith('zipcode') and not x.startswith('
        print (f' As {x} increases by one unit, the price cl
    else:
        x.startswith('zipcode') and not x.startswith('const
        print (f' If a property was in {x} zipcode instead 
```

```
 If a property was in zipcode_98146 zipcode instead of 98039, the price wo
percent.
 If a property was in zipcode_98148 zipcode instead of 98039, the price wo
percent.
 If a property was in zipcode_98155 zipcode instead of 98039, the price wo
ercent.
 If a property was in zipcode_98166 zipcode instead of 98039, the price wo
ercent.
 If a property was in zipcode_98168 zipcode instead of 98039, the price wo
percent.
 If a property was in zipcode_98177 zipcode instead of 98039, the price wo
ercent.
 If a property was in zipcode_98178 zipcode instead of 98039, the price wo
percent.
 If a property was in zipcode_98188 zipcode instead of 98039, the price wo
percent.
 If a property was in zipcode_98198 zipcode instead of 98039, the price wo
percent.
 If a property was in zipcode_98199 zipcode instead of 98039, the price wo
ercent.
```

14 Recommendations:

1) As we can read in the printed report above, improving the grac
(as is our aim with renovations) would make the value of the prop
roughly 1/4th. *Therefore* I'm recommending that the city of Seattl

Contents

- 6.1 Comparing Renovated Homes with Unrenovated
- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fc
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

both grade and condition of properties during the renovation. In tl year_renovated by itself had little affect on the price of the home. is that it's not enough to simply renovate a home, but we must im and/or grade with that renovation.

Choosing which properties to renovate: the properties where the changes will increase the grade, and/or condition of the property high return on investment.

2) There are certain zipcodes where the class disparity is quite la the city of Seattle spend 1/3 of this current projects budget just fc properties in these zipcodes with the lowest grade/conditions, an public housing. These zipcodes are as follows: 98168, 98188, 98 98146, 98108, 98092, 98058, 98055, 98042, 98032, 98031, 9803 98002, 98001. The data shows that properties in these zipcodes value by over 100% if they were in the 98039 zipcode instead.

3) While one and a half percent may not seem like a lot--there is evidence that putting new schools specifically where the closest s will improve the property values of all the homes closest to that s increase distance away from homes, the property value of the ho 1.5%. While building schools is outside the range of the scope of project, it is a finding to take back to the city counsel and state se recommend that when selecting properties for this project, flaggir within 3 miles of a public school for selection across zipcodes. Pe deserve functional infrastructure.

14.0.1 Visualizing Schools

In [240]:

```
school_coords = []
for x, y in coordinates.items():
    if x in school_addresses:
        school_coords.append(y)
```

Contents

- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
  - 11.1 Baseline Model: Grade, Bathrooms, Lat
  - 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
  - 12.1 Third Model: Bedrooms, Floors, Closest School
  - 12.2 Fourth Model: Square Foot Basement, Closest
  - 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
  - 12.4 Sixth Model: Condition, Waterfront and View
  - 12.5 Seventh Model: Adding Zipcodes
  - 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

In [241]:

```
school_labels = dict(zip(school_names, school_coords))
```

In [242]:

```
base_map = folium.Map(downtown, zoom_start=11)

for school, location in school_labels.items():
    lat = location[0]
    long = location[1]
    popup_text = school
    popup = folium.Popup(popup_text, parse_html=True)
    marker = folium.CircleMarker(location=[lat, long], popu
                                 fill = True, radius = 20)
    marker.add_to(base_map)

base_map
```

Contents

- 6.1 Comparing Renovated Homes with Unrenovated
- 6.2 Visualizing the Comparison
- 6.3 Home History
  - 6.3.1 Age of Home
  - 6.3.2 Year Modified
- 7 Distance Data
- 7.1 Getting Infrastucture Data
  - 7.1.1 Downtown
  - 7.1.2 Hospitals
  - 7.1.3 Police Stations
  - 7.1.4 Schools
- 7.2 Finding Distance for Infrastructure
- 7.3 Exploring Distance Data
  - 7.3.1 Downtown
  - 7.3.2 Hospitals
  - 7.3.3 Police Stations
  - 7.3.4 Schools
- 8 Preparing for Modeling
  - 8.0.1 Dropping View and Waterfront Properties fo
- 9 Zipcode
- 10 Seeing the Correlation
- 11 Building a model
- 11.1 Baseline Model: Grade, Bathrooms, Lat
- 11.2 Second Model: Transforming the Price with Log
- 12 Improving the Model with Adding Features
- 12.1 Third Model: Bedrooms, Floors, Closest School
- 12.2 Fourth Model: Square Foot Basement, Closest
- 12.3 Fifth Model: Square Foot Lot of Closest 15 Nei
- 12.4 Sixth Model: Condition, Waterfront and View
- 12.5 Seventh Model: Adding Zipcodes
- 12.6 Eighth Model: Latitude and Closest Hospital D
- 13 Conclusion
- 14 Recommendations:
  - 14.0.1 Visualizing Schools

While we can visually inspect the map above and look for places could go, we can also combine our knowledge of the zipcode dat building a new school in the zipcodes that have the highest class the wealthiest zipcode such as zipcode 98032 (the zipcode with t properties in this dataset.)

FURTHER WORK:

With more time and resources, I'd like to exactly calculate the are isolated from public schools, but the above map shows a rough id areas of the map not shaded blue by a circle. In the future, I'd like the distance of these circles in human measurements, not in pixe