In [18]:
```
#importing all the libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns

%matplotlib inline

#importing dataset using panda
dataset = pd.read_csv(r"C:\Users\admin\Downloads\24th- mlr\25th- mlr\MLR\House_data
#to see what my dataset is comprised of
dataset.head()
```

Out[18]:

	id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot	flo
0	7129300520	20141013T000000	221900.0	3	1.00	1180	5650	
1	6414100192	20141209T000000	538000.0	3	2.25	2570	7242	
2	5631500400	20150225T000000	180000.0	2	1.00	770	10000	
3	2487200875	20141209T000000	604000.0	4	3.00	1960	5000	
4	1954400510	20150218T000000	510000.0	3	2.00	1680	8080	

5 rows × 21 columns

In [19]:
```
#checking if any value is missing
print(dataset.isnull().any())
```

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

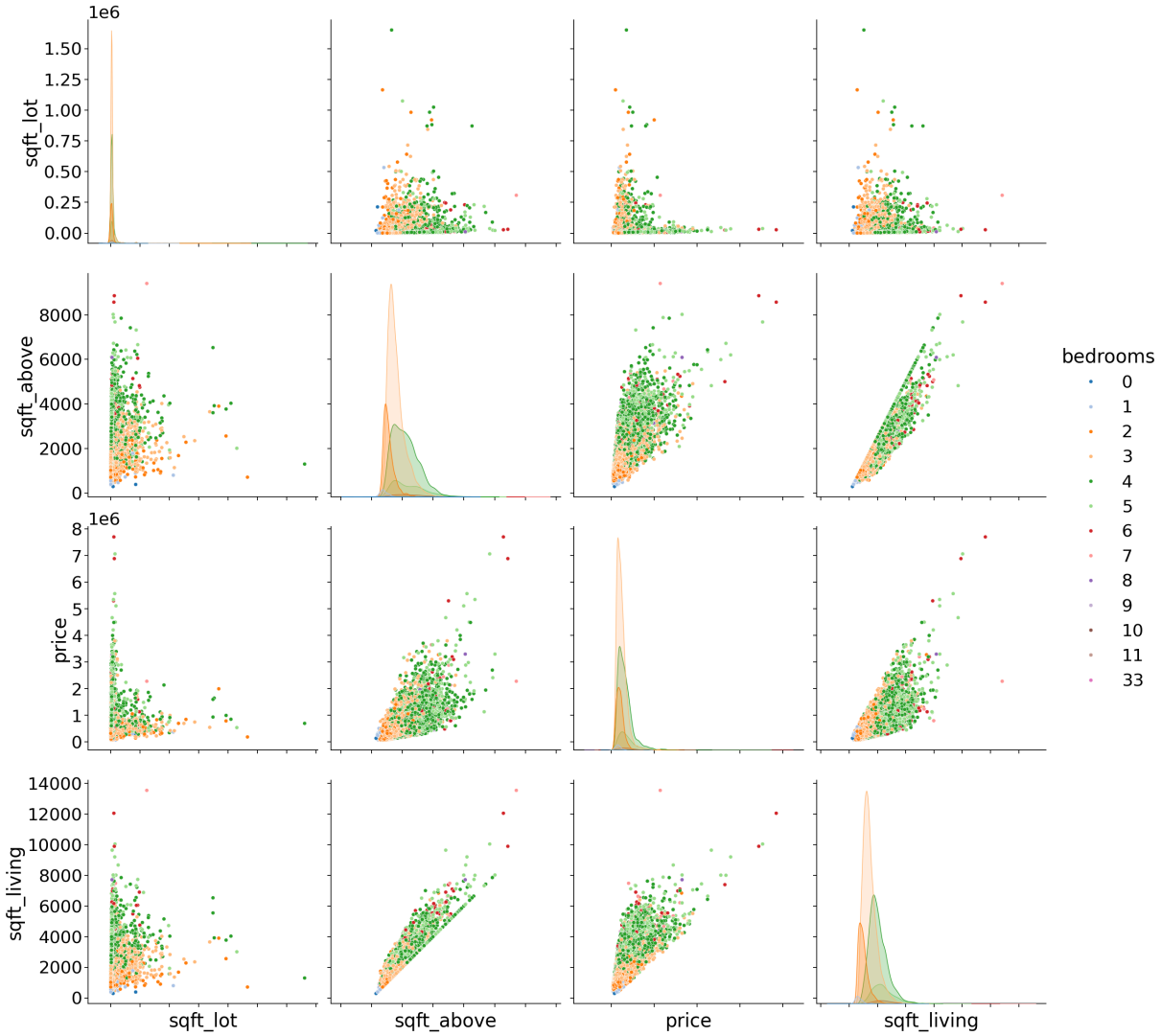
In [20]:
```
#checking for categorical data
print(dataset.dtypes)
```

```
id                int64
date             object
price           float64
bedrooms          int64
bathrooms       float64
sqft_living       int64
sqft_lot          int64
floors          float64
waterfront        int64
view              int64
condition         int64
grade             int64
sqft_above        int64
sqft_basement     int64
yr_built          int64
yr_renovated      int64
zipcode           int64
lat             float64
long            float64
sqft_living15     int64
sqft_lot15        int64
dtype: object
```

In [21]:
```
#dropping the id and date column
dataset = dataset.drop(['price','date'], axis = 1)
```

In [8]:
```
#understanding the distribution with seaborn
with sns.plotting_context("notebook",font_scale=2.5):
    g = sns.pairplot(dataset[['sqft_lot','sqft_above','price','sqft_living','bedroo
                 hue='bedrooms', palette='tab20',size=6)
g.set(xticklabels=[]);
```

```
c:\Users\admin\AppData\Local\Programs\Python\Python313\Lib\site-packages\seaborn\axi
sgrid.py:2100: UserWarning: The `size` parameter has been renamed to `height`; pleas
e update your code.
  warnings.warn(msg, UserWarning)
```



In [24]:
```
# Separating independent and dependent variables
X = dataset.iloc[:, 1:].values  # all columns except first
y = dataset.iloc[:, 0].values   # first column (target)

# Splitting dataset into training and testing datasets
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=1/3, random_state=0
)
```

In [25]:
```
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor.fit(X_train, y_train)

# Predicting the Test set results
y_pred = regressor.predict(X_test)
```

In [31]:
```
import statsmodels.api as sm
import numpy as np

def backwardElimination(X, y, SL=0.05):
    X = sm.add_constant(X)  # add intercept
```

```
    numVars = X.shape[1]

    for i in range(numVars):
        regressor_OLS = sm.OLS(y, X).fit()
        maxPval = max(regressor_OLS.pvalues)

        if maxPval > SL:
            maxPvalIndex = np.argmax(regressor_OLS.pvalues)
            X = np.delete(X, maxPvalIndex, 1)
        else:
            break

    print(regressor_OLS.summary())
    return X

# Usage
SL = 0.05
X_opt = X[:, :18]  # select first 18 columns as potential features
X_Modeled = backwardElimination(X_opt, y, SL)
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.027
Model:                            OLS   Adj. R-squared:                  0.027
Method:                 Least Squares   F-statistic:                     75.09
Date:                Thu, 25 Sep 2025   Prob (F-statistic):          9.22e-123
Time:                        13:45:39   Log-Likelihood:            -5.0110e+05
No. Observations:               21613   AIC:                         1.002e+06
Df Residuals:                   21604   BIC:                         1.002e+06
Df Model:                           8
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const       1.737e+11   1.86e+10      9.336      0.000    1.37e+11     2.1e+11
x1         -5140.4748    672.841     -7.640      0.000   -6459.294   -3821.656
x2          1.145e+08   2.66e+07      4.310      0.000    6.24e+07    1.67e+08
x3         -9.106e+07   3.03e+07     -3.005      0.003    -1.5e+08   -3.17e+07
x4          6.539e+07   2.57e+07      2.543      0.011     1.5e+07    1.16e+08
x5         -1.002e+05   3.76e+04     -2.662      0.008   -1.74e+05   -2.64e+04
x6         -1.077e+05   4.86e+04     -2.217      0.027   -2.03e+05   -1.25e+04
x7          1.382e+09   1.52e+08      9.080      0.000    1.08e+09    1.68e+09
x8         -1.081e+04   1027.972    -10.514      0.000   -1.28e+04   -8792.856
==============================================================================
Omnibus:                    22223.809   Durbin-Watson:                   2.003
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             1510.573
Skew:                           0.224   Prob(JB):                         0.00
Kurtosis:                       1.785   Cond. No.                     4.87e+07
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly spe
cified.
[2] The condition number is large, 4.87e+07. This might indicate that there are
strong multicollinearity or other numerical problems.
```