In [14]:

```
# This Python 3 environment comes with many helpful analytics libraries installed
# It is defined by the kaggle/python Docker image: https://github.com/kaggle/docker-pytho
n
# For example, here's several helpful packages to load

import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)

# Input data files are available in the read-only "../input/" directory
# For example, running this (by clicking run or pressing Shift+Enter) will list all files
under the input directory

import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))

# You can write up to 20GB to the current directory (/kaggle/working/) that gets preserve
d as output when you create a version using "Save & Run All"
# You can also write temporary files to /kaggle/temp/, but they won't be saved outside of
the current session
```

```
/kaggle/input/home-data-for-ml-course/sample_submission.csv
/kaggle/input/home-data-for-ml-course/sample_submission.csv.gz
/kaggle/input/home-data-for-ml-course/train.csv.gz
/kaggle/input/home-data-for-ml-course/data_description.txt
/kaggle/input/home-data-for-ml-course/test.csv.gz
/kaggle/input/home-data-for-ml-course/train.csv
/kaggle/input/home-data-for-ml-course/test.csv
```

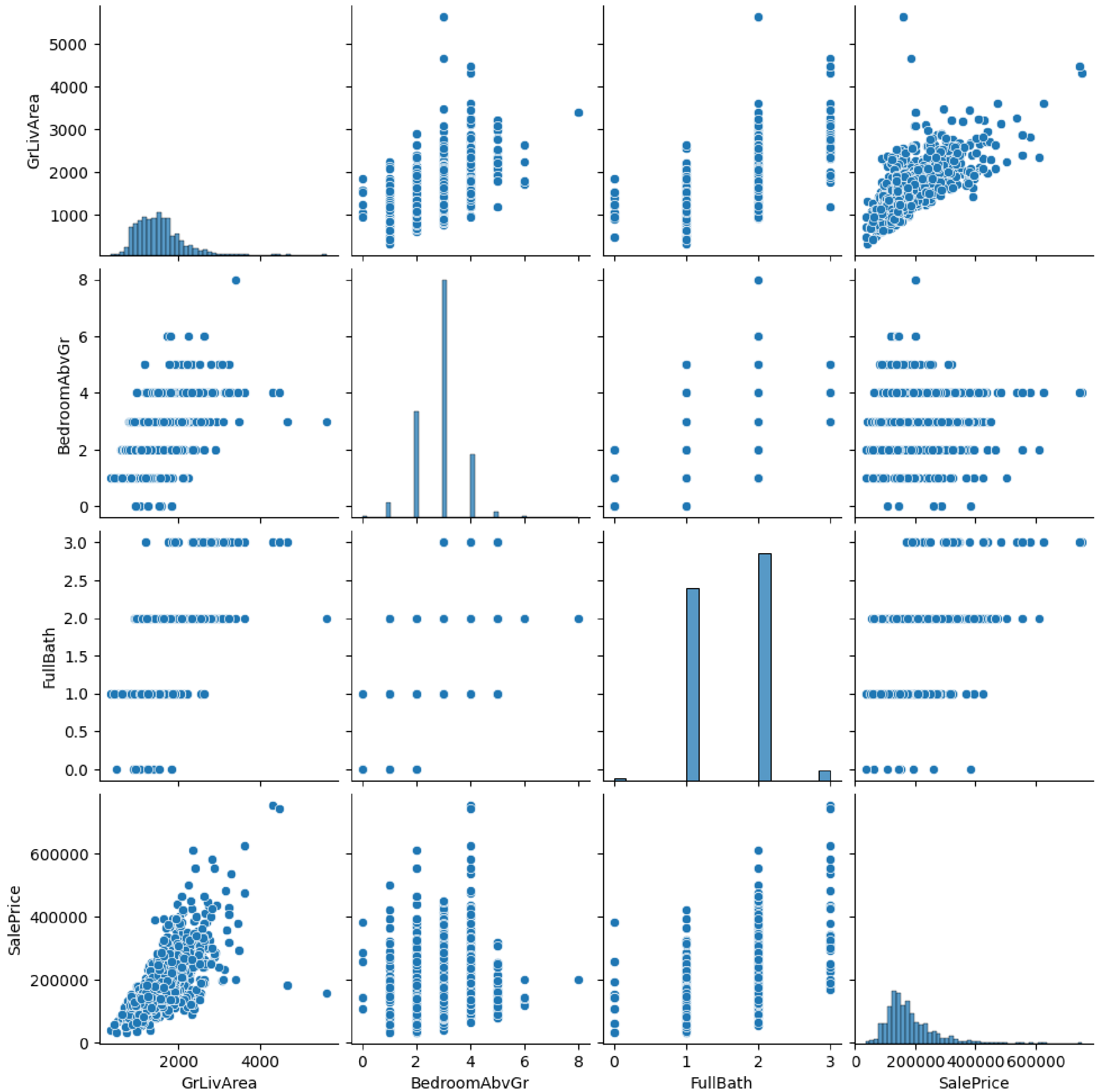
In [15]:

```
import pandas as pd
import seaborn as s
import matplotlib.pyplot as plt

from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error,r2_score
import numpy as np
ds=pd.read_csv("/kaggle/input/home-data-for-ml-course/train.csv")
print(ds.head())
style = ['GrLivArea', 'BedroomAbvGr', 'FullBath', 'SalePrice']
data = ds[style].dropna()

s.pairplot(data)
plt.show()
```

```
   Id  MSSubClass MSZoning  LotFrontage  LotArea Street Alley LotShape  \
0   1          60       RL         65.0     8450   Pave   NaN      Reg
1   2          20       RL         80.0     9600   Pave   NaN      Reg
2   3          60       RL         68.0    11250   Pave   NaN      IR1
3   4          70       RL         60.0     9550   Pave   NaN      IR1
4   5          60       RL         84.0    14260   Pave   NaN      IR1

  LandContour Utilities  ... PoolArea PoolQC Fence MiscFeature MiscVal MoSold  \
0         Lvl    AllPub  ...        0    NaN   NaN         NaN       0      2
1         Lvl    AllPub  ...        0    NaN   NaN         NaN       0      5
2         Lvl    AllPub  ...        0    NaN   NaN         NaN       0      9
3         Lvl    AllPub  ...        0    NaN   NaN         NaN       0      2
4         Lvl    AllPub  ...        0    NaN   NaN         NaN       0     12

  YrSold  SaleType  SaleCondition  SalePrice
0   2008        WD         Normal     208500
1   2007        WD         Normal     181500
2   2008        WD         Normal     223500
3   2006        WD        Abnorml     140000
```

```
4   2008        WD        Normal     250000

[5 rows x 81 columns]
```

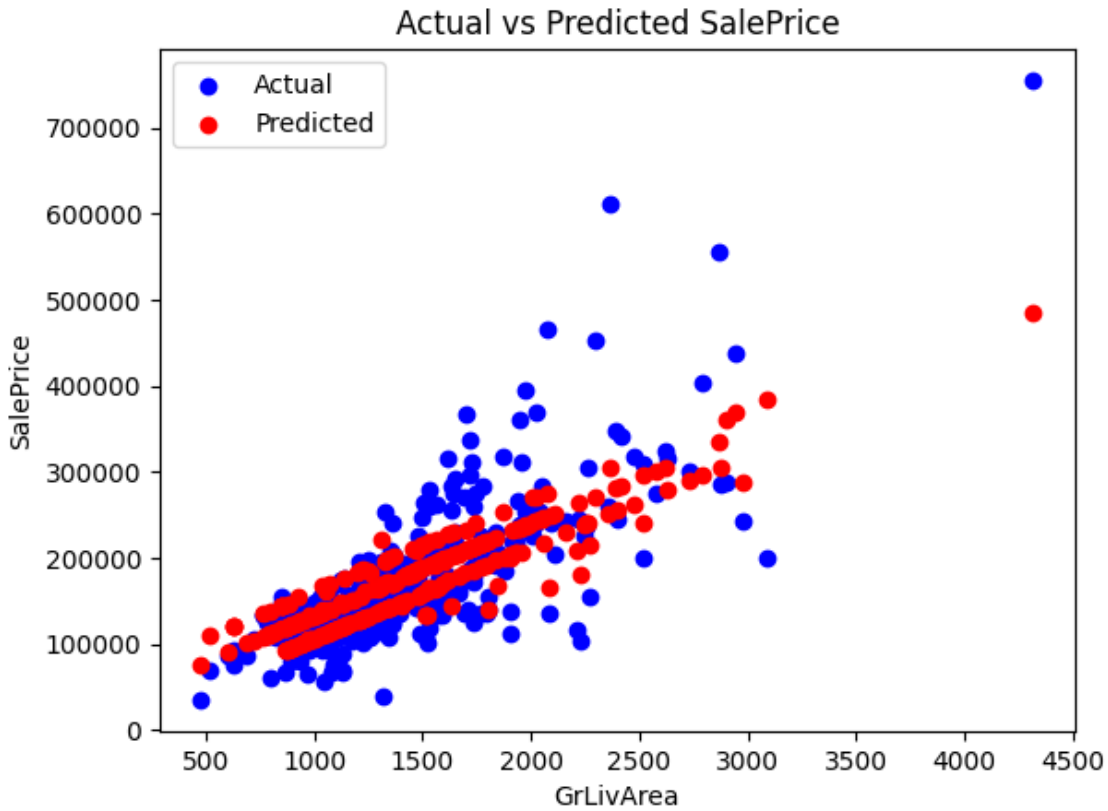
```
/usr/local/lib/python3.11/dist-packages/seaborn/_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
/usr/local/lib/python3.11/dist-packages/seaborn/_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
/usr/local/lib/python3.11/dist-packages/seaborn/_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
/usr/local/lib/python3.11/dist-packages/seaborn/_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
/usr/local/lib/python3.11/dist-packages/seaborn/axisgrid.py:118: UserWarning: The figure layout has changed to tight
  self._figure.tight_layout(*args, **kwargs)
```

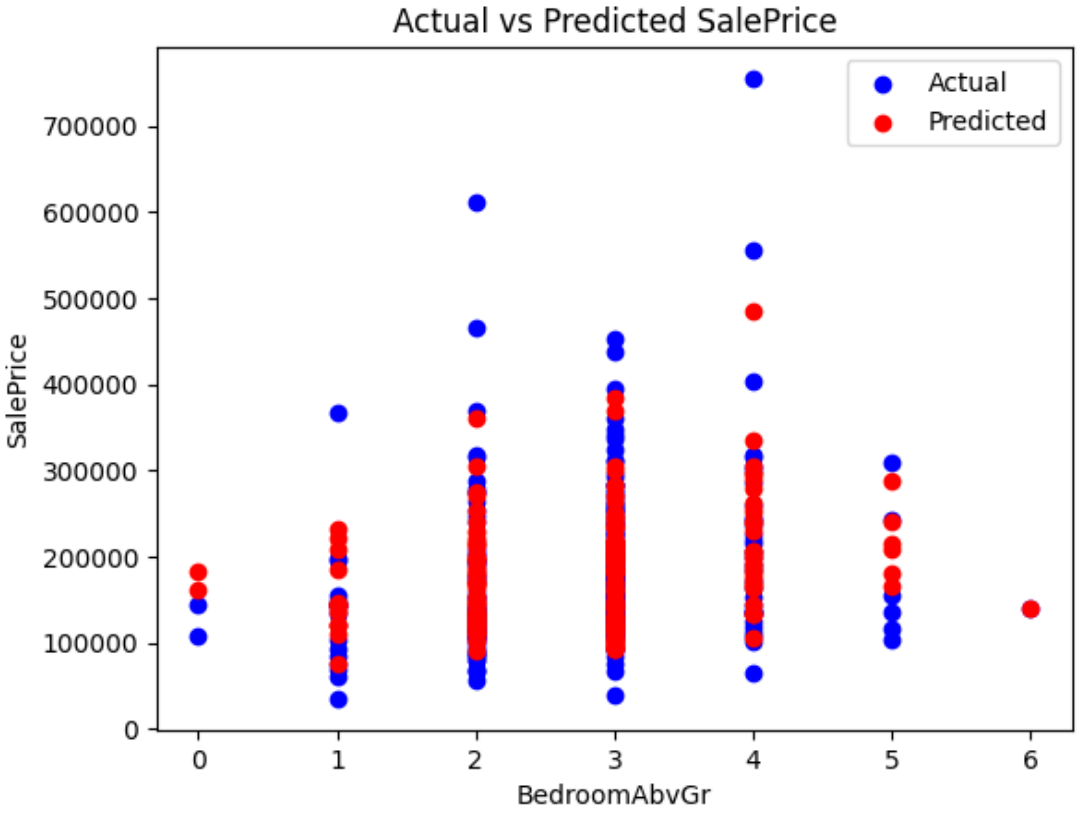


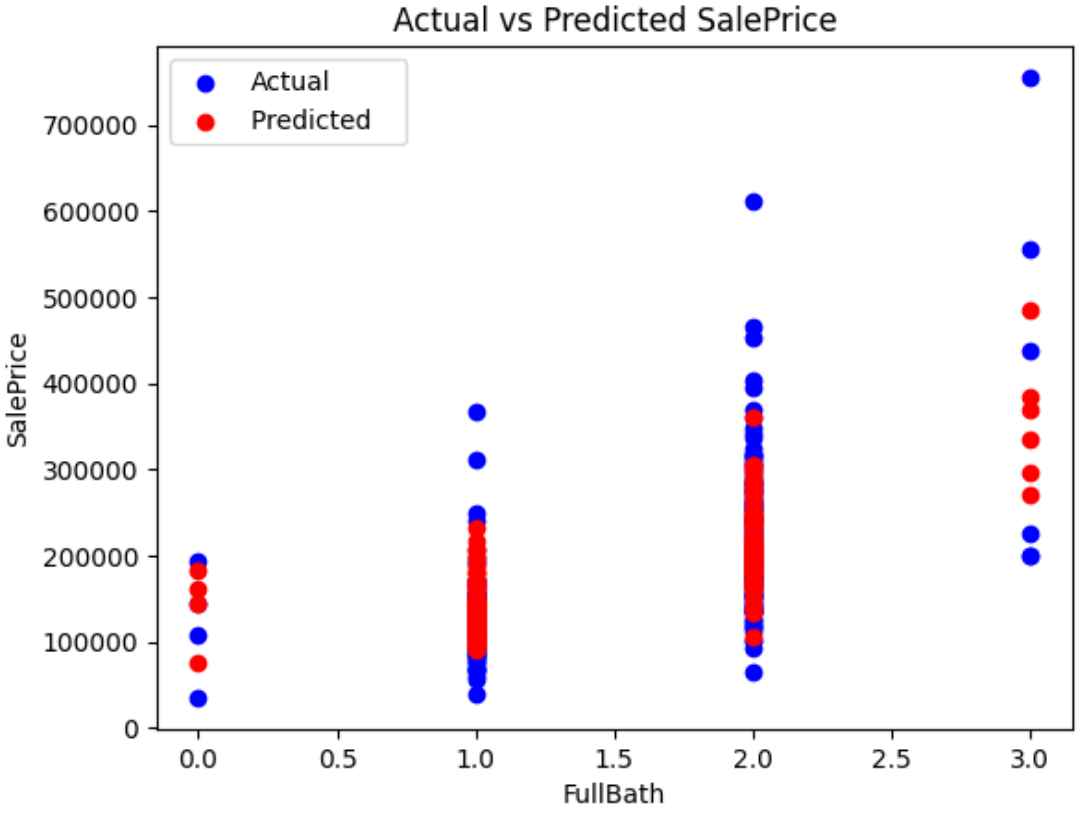
In [16]:

```
x_train, x_test, y_train, y_test = train_test_split(data.drop('SalePrice', axis=1), data
['SalePrice'], test_size=0.2, random_state=42)
model = LinearRegression()
model.fit(x_train, y_train)
y_pred = model.predict(x_test)
mse = mean_squared_error(y_test, y_pred)
print(f'Mean Squared Error: {mse}')
print("R2 Score:", r2_score(y_test, y_pred))
plt.scatter(x_test['GrLivArea'], y_test, color='blue', label='Actual')
plt.scatter(x_test['GrLivArea'], y_pred, color='red', label='Predicted')
plt.xlabel('GrLivArea')
plt.ylabel('SalePrice')
plt.title('Actual vs Predicted SalePrice')
plt.legend()
plt.show()
plt.scatter(x_test['BedroomAbvGr'], y_test, color='blue', label='Actual')
plt.scatter(x_test['BedroomAbvGr'], y_pred, color='red', label='Predicted')
plt.xlabel('BedroomAbvGr')
plt.ylabel('SalePrice')
plt.title('Actual vs Predicted SalePrice')
plt.legend()
plt.show()
plt.scatter(x_test['FullBath'], y_test, color='blue', label='Actual')
plt.scatter(x_test['FullBath'], y_pred, color='red', label='Predicted   ')
plt.xlabel('FullBath')
plt.ylabel('SalePrice')
plt.title('Actual vs Predicted SalePrice')
plt.legend()
plt.show()
gr_liv_area = float(input("Enter the living area in square feet: "))
bedrooms = int(input("Enter the number of bedrooms: "))
full_bath = int(input("Enter the number of full bathrooms: "))

newhouse = [[gr_liv_area, bedrooms, full_bath]]


predicted_price = model.predict(newhouse)

print(f"\n Estimated House Price: ${predicted_price[0]:,.2f}")
```

```
Mean Squared Error: 2806426667.247853
R2 Score: 0.6341189942328371
```








```
Estimated House Price: $5,233,640.25
```

```
/usr/local/lib/python3.11/dist-packages/sklearn/base.py:439: UserWarning: X does not have
valid feature names, but LinearRegression was fitted with feature names
  warnings.warn(
```