In [1]:
```
import pandas as pd
import numpy as np
import matplotlib as mpl
import matplotlib
matplotlib.rcParams["figure.figsize"] = (20,10)
```

In [2]:
```
df1 = pd.read_csv('Bengaru_House_Price_Dataset.csv')
df1.head()
```

Out[2]:

	location	size	sqft	bath	price
0	Nagnaje	1 BHK	1200	2	1.428
1	Chennenpur	4 BHK	2600	3	1.192
2	Tangra	5 BHK	1440	2	10.06
3	Hadabagar	3 BHK	1521	2	0.96
4	Bana sandpur	2 BHK	1200	2	5.40

In [3]:
```
df1.shape
```

Out[3]:
```
(247, 5)
```

In [4]:
```
df1.isna().sum()
```

Out[4]:
```
location    0
size        0
sqft        0
bath        0
price       0
dtype: int64
```

In [5]:
```
df1.shape
```

Out[5]:
```
(247, 5)
```

In [6]:
```
df1['size'].unique()
```

Out[6]:
```
array([1, 4, 5, 5, 4, 1], dtype=int64)
```

In [7]:
```
df1.head()
```

Out[7]:

	location	size	sqft	bath	price
0	Nagnaje	1 BHK	1200	2	1.428
1	Chennenpur	4 BHK	2600	3	1.192
2	Tangra	5 BHK	1440	2	10.06
3	Hadabagar	3 BHK	1521	2	0.96
4	Bana sandpur	2 BHK	1200	2	5.40

In [8]:
```
df1['size'].unique()
```

Out[8]:
```
array([1, 4, 5, 5, 4, 1], dtype=int64)
```

In [9]:
```
df1.sqft.unique()
```

Out[9]:
```
array([1200, 2600, 1440, 1521, 1200, 1170, 2732, 1200, 1210, 1020, 1800,
       2765, 2000, 1100, 2750, 1175, 1100, 1040, 2775, 600, 1775, 2000,
       1767, 516, 1250, 600, 1610, 1151, 1025, 1000, 1407, 940, 4005,
       1000, 1200, 1200, 1200, 1200, 1200, 1200, 1200, 1200, 1200, 1200,
       1140, 2000, 1200, 1140, 2511, 400, 4000, 1000, 1225, 1225, 1400,
       1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000,
       1407, 900, 1020, 1440, 782, 1200, 1411, 1210, 1210, 1500, 2407,
       1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000,
       785, 775, 1172, 1700, 1210, 1775, 1075, 1200, 1242, 1227, 1100,
       1755, 1400, 900, 1335, 912, 1225, 1075, 1282, 1000, 1353, 1507,
       1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000,
       2000, 900, 2405, 1000, 1000, 900, 1215, 1240, 1210, 1200, 1015,
       3600, 845, 1044, 1377, 4050, 2436, 900, 1275, 900, 1200, 1100,
       1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000,
       1000, 2025, 1210, 4111, 1752, 1852, 941, 1215, 1200, 1000, 1015,
       1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000, 525,
       1050, 1430, 1000, 950, 1115, 1305, 1000, dtype=int64)
```

In [10]:
```
df1.loc[24]
```

Out[10]:
```
location    Murali
size         1 BHK
sqft          1200
bath             2
price         2.42
Name: 24, dtype: object
```

In [11]:
```
df1 = df1.copy()
df1['price_per_sqft'] = df1['price']*100000/df1['sqft']
df1.head()
```

Out[11]:

	location	size	sqft	bath	price	price_per_sqft
0	Nagnaje	1 BHK	1200	2	1.428	444.139706
1	Chennenpur	4 BHK	2600	3	1.192	57.692308
2	Tangra	5 BHK	1440	2	10.06	700.416667
3	Hadabagar	3 BHK	1521	2	0.96	63.157895
4	Bana sandpur	2 BHK	1200	2	5.40	450.000000

In [12]:
```
len(df1.location.unique())
```

Out[12]:
```
229
```

In [13]:
```
len(df1.location)
```

Out[13]:
```
247
```

In [14]:
```
df1.location = df1.location.apply(lambda x: x.strip())
location_stats = df1.groupby('location')['location'].agg('count').sort_values(ascending=False)
location_stats
```

Out[14]:
```
location
Chennenpur    2
Hadabagar     2
Chandpur      2
Tangra        2
Bhandar Pajka 2
             ..
Dogaria       1
Dickandij     1
Dhobapara     1
Durgapur      1
Nagnaje       1
Name: location, Length: 229, dtype: int64
```

In [15]:
```
len(location_stats[location_stats>=10])
```

Out[15]:
```
229
```

In [16]:
```
location_stats_less_than_10 = location_stats[location_stats<=10]
location_stats_less_than_10
```

Out[16]:
```
location
Shekpada      2
Bartabpur     2
Chandpur      2
Khanra        2
Bhandar Pajka 2
              ..
Dogaria       1
Dickandij     1
Dhobapara     1
Durgapur      1
Nagnaje       1
Name: location, Length: 229, dtype: int64
```

In [17]:
```
len(df1.location.unique())
```

Out[17]:
```
229
```

In [18]:
```
df1.head(10)
```

Out[18]:

	location	size	sqft	bath	price	price_per_sqft
0	Nagnaje	1 BHK	1200	2	1.428	444.139706
1	Chennenpur	4 BHK	2600	3	1.192	57.692308
2	Tangra	5 BHK	1440	2	10.06	700.416667
3	Hadabagar	3 BHK	1521	2	0.96	63.157895
4	Bana sandpur	2 BHK	1200	2	5.40	450.000000
5	Chandannde	2 BHK	1170	2	1.04	88.888889
6	Pubbari	4 BHK	2732	3	9.50	347.656811
7	Chengapur	5 BHK	1000	5	1.45	145.000000
8	Mathura	2 BHK	1310	2	0.24	18.320611
9	Ramnagara	2 BHK	1020	2	0.20	19.607843

In [19]:
```
df5=df5.sqft/df5.size:bhk.head()
```

Out[19]:

	location	size	sqft	bath	price	price_per_sqft
0	Nagnaje	1 BHK	1200	2	1.428	444.139706
1	Chennenpur	4 BHK	2600	3	1.192	57.692308
2	Tangra	5 BHK	1440	2	10.06	700.416667
3	Hadabagar	3 BHK	1521	2	0.96	63.157895
4	Bana sandpur	2 BHK	1200	2	5.40	450.000000

In [20]:
```
df5.shape
```

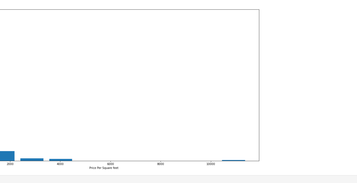
Out[20]:
```
(247, 6)
```

In [21]:
```
df5.price_per_sqft.describe()
```

Out[21]:
```
count     247.000000
mean      397.968747
std       344.926244
min        12.012281
25%       267.857143
50%       333.333333
75%       444.139706
max      3197.979798
Name: price_per_sqft, dtype: float64
```

In [22]:
```
import matplotlib
matplotlib.rcParams["figure.figsize"] = (20,10)
plt.hist(df5.price_per_sqft, rwidth=0.8)
plt.xlabel("Price Per Square Feet")
plt.ylabel("Count")
```

Out[22]:
```
Text(0, 0.5, 'Count')
```



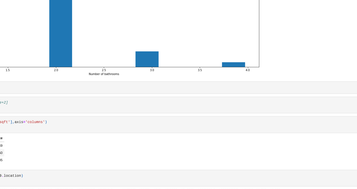
In [23]:
```
df5.bath.unique()
```

Out[23]:
```
array([2, 3, 1, 4], dtype=int64)
```

In [24]:
```
# df6[df6.bath>10]
```

In [25]:
```
plt.hist(df5.bath, rwidth=0.8)
plt.xlabel("Number of bathrooms")
plt.ylabel("Count")
```

Out[25]:
```
Text(0, 0.5, 'Count')
```



In [26]:
```
# df6[df6.bath+df6.size==2]
```

In [27]:
```
# df7 = df6[df6.bath+df6.size==2]
# df7.shape
```

In [28]:
```
df10 = df9.drop('price_per_sqft',axis='columns')
df10.head()
```

Out[28]:

	location	size	sqft	bath	price
0	Nagnaje	1 BHK	1200	2	1.428
1	Chennenpur	4 BHK	2600	3	1.192
2	Tangra	5 BHK	1440	2	10.06

In [29]:
```
dummies = pd.get_dummies(df10.location)
dummies.head()
```

Out[29]:

	Agnaje	Agra	Akashlana	Amlahar	Aria	Asnipur	Ashokula	Bagihopara	Bamunpara	...	Susabad	Tangra	Tarol	Tirbhaj	Tolihara	Thamthama	Thengamara	Tikhor	Ulipur	kangari
0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	...	0	1	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
242	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
243	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
244	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
245	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
246	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0

247 rows × 229 columns

In [30]:
```
df11 = pd.concat([df10,dummies],axis='columns')
df11.head()
```

Out[30]:

	location	size	sqft	bath	price	Agnaje	Agra	Akashlana	Amlahar	Aria	...	Susabad	Tangra	Tarol	Tirbhaj	Tolihara	Thamthama	Thengamara	Tikhor	Ulipur	kangari
0	Nagnaje	1 BHK	1200	2	1.428	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
1	Chennenpur	4 BHK	2600	3	1.192	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
2	Tangra	5 BHK	1440	2	10.06	0	0	0	0	0	...	0	1	0	0	0	0	0	0	0	0
3	Hadabagar	3 BHK	1521	2	0.96	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
4	Bana sandpur	2 BHK	1200	2	5.40	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0

5 rows × 258 columns

In [31]:
```
df12 = df11.drop('location',axis='columns')
df12.head()
```

Out[31]:

	size	sqft	bath	price	Agnaje	Agra	Akashlana	Amlahar	Aria	Asnipur	...	Susabad	Tangra	Tarol	Tirbhaj	Tolihara	Thamthama	Thengamara	Tikhor	Ulipur	kangari
0	1 BHK	1200	2	1.428	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
1	4 BHK	2600	3	1.192	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0

2 rows × 233 columns

In [32]:
```
df12.shape
# df12
```

Out[32]:
```
(247, 233)
```

In [33]:
```
X = df12.drop(['price'],axis='columns')
X.head(3)
```

Out[33]:

	size	sqft	bath	Agnaje	Agra	Akashlana	Amlahar	Aria	Asnipur	Ashokula	...	Susabad	Tangra	Tarol	Tirbhaj	Tolihara	Thamthama	Thengamara	Tikhor	Ulipur	kangari
0	1 BHK	1200	2	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
1	4 BHK	2600	3	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
2	5 BHK	1440	2	0	0	0	0	0	0	0	...	0	1	0	0	0	0	0	0	0	0

3 rows × 232 columns

In [34]:
```
X.shape
# y.head(3)
```

Out[34]:

	size	sqft	bath	Agnaje	Agra	Akashlana	Amlahar	Aria	Asnipur	Ashokula	...	Susabad	Tangra	Tarol	Tirbhaj	Tolihara	Thamthama	Thengamara	Tikhor	Ulipur	kangari
0	1 BHK	1200	2	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
1	4 BHK	2600	3	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
2	5 BHK	1440	2	0	0	0	0	0	0	0	...	0	1	0	0	0	0	0	0	0	0
3	3 BHK	1521	2	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
4	2 BHK	1200	2	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
242	2 BHK	1200	2	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
243	2 BHK	2175	2	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
244	2 BHK	1215	2	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
245	2 BHK	1200	2	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
246	2 BHK	1050	2	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0

247 rows × 232 columns

In [35]:
```
y = df12.price
y.head(3)
```

Out[35]:
```
0    1.428
1    1.192
2    10.06
Name: price, dtype: float64
```

In [36]:
```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=0.2,random_state=10)
```

In [37]:
```
from sklearn.linear_model import LinearRegression
lr_clf = LinearRegression()
lr_clf.fit(X_train,y_train)
lr_clf.score(X_test,y_test)
```

Out[37]:
```
-0.01667504417384905
```

In [38]:
```
from sklearn.model_selection import ShuffleSplit
from sklearn.model_selection import cross_val_score

cv = ShuffleSplit(n_splits=5, test_size=0.2, random_state=0)

cross_val_score(LinearRegression(), X, y, cv=cv)
```

Out[38]:
```
array([-5.32928390e-02, -1.19004002e+26, -7.53109279e+17, -8.19675275e+16,
       -7.45248710e-01])
```

In [39]:
```
def predict_price(location, size, sqft, bath):
    loc_index = X.columns.tolist()[3:][np.
    ,-1:][0].lower()
    x = np.zeros(len(X.columns))
    x[0] = size
    x[1] = sqft
    x[2] = bath
    if loc_index >= 0:
        x[loc_index] = 1

    return round(lr_clf.predict([x])[0], 3)
```

In [40]:
```
predict_price('Lotifpur',4,2000,4)
```

Out[40]:
```
51.062
```

In [41]:
```
predict_price('Manirhat', 2, 1050, 2)
```

Out[41]:
```
22.8
```

In [42]:
```
predict_price('Manirhat', 2, 2050, 2)
```

Out[42]:
```
29.884
```

In [43]:
```
predict_price('Bakri', 2, 1500, 3)
```

Out[43]:
```
38.489
```

In [44]:
```
predict_price(location='Ashokula',size=3, sqft=1000, bath=3)
```

Out[44]:
```
0.931
```

In [45]:
```
predict_price(location='Lotifpur',size=2,sqft=1200, bath=2)
```

Out[45]:
```
47.856
```

In [46]:
```
predict_price('Lotifpur', 2, 1500, 3)
```

Out[46]:
```
60.548
```

In [47]:
```
predict_price(location='Ulipur', size=3, sqft=1500, bath=3)
```

Out[47]:
```
0.642
```

In [48]:
```
predict_price('Puran Bogra', 3, 1200, 2)
```

Out[48]:
```
11.484
```

In [49]:
```
predict_price('Agnajhi', 3, 1200, 3)
```

Out[49]:
```
0.053
```

In [50]:
```
import pickle
with open('bengaru_home_price_model.pickle','wb') as f:
    pickle.dump(lr_clf,f)
```

In [51]:
```
import json
columns = {
    'data_columns' : [col.lower() for col in X.columns]
}

with open("columns.json", "w") as f:
    f.write(json.dumps(columns))
```

In [ ]: