Seshadripuram Educational Trust

# Seshadripuram College Tumakuru

**"Vikasa Bharathi" No.3, Melekote, Veerasagara Layout, Gangasandra Road**

**Tumakuru- 572106**

AFFILIATED TO TUMKUR UNIVERSITY

# House Prize Prediction

**Submitted By:**

**Manoj B - P11SZ23S126017**

**Name of the Trainer: Karthik**

**2023-24**

**DECLARATION BY THE STUDENT**

I'm Manoj.B, I'm currently studying MCA in Seshadripuram College Tumkur, hereby declare that all the information furnished in this project report is based on my own intensive work and is genuine.

**Place:** Tumkur

**Date:** **Signature:**

## GUIDE CERTIFICATE (Internal)

This is to certify that the Training Report **"Training on elementary data and analytics communication skill building and job readiness"** submitted by **Manoj B - [P11SZ23S126017]** to Seshadripuram College Tumkur for the award of degree of masters of computer Application is a record of work carried out by his/her under my guidance.

**Place:** Tumkur

**Date:** **Signature:**

## GUIDE CERTIFICATE (External)

This is to certify that **Manoj B** [**P11SZ23S126017]** has undergone training from me on **Machine learning** from **05/08/2024** to **08/08/2024** as a part of training program from Seshadripuram College Tumkur for the award of Degree of Master of computer Application.

**Place:** Tumkur

**Date:** **Signature:**

## Certificate

This is to certify that **Manoj B** bearing Registration no **P11SZ23S126017** Has successfully completed **Machine learning** project title, “House prize prediction” under my guidance and supervision to the best o my knowledge, the present work is the result off his original development effort, and study.

**Place:** Tumkur

**Date:** **Signature:**

**Table of content:**

## Introduction to python:

Python was created by **Guido van Rossum**. He started developing Python in the late 1980s and released its first version in 1991. Guido van Rossum is often referred to as the "Benevolent Dictator For Life" (BDFL) due to his long-term involvement in overseeing the language's development and guiding its evolution.

Van Rossum designed Python with an emphasis on code readability and simplicity, which has contributed to its popularity and widespread use in various fields today.

**What is Python :**

**Python** is a high-level, interpreted programming language known for its readability and simplicity. It's widely used in various fields, from web development and data analysis to artificial intelligence and scientific computing.

**Features of python:**

- Readability and Simplicity
- Interpreted Language
- High-Level Language
- Dynamic Typing
- Embeddable
- Interactive

## Simple program:

```
import calendar as ya

year = int(input ('enter the year: '))

month = int(input ("enter the month: "))

print (ya.month(year,month))

print (ya.calendar(year))
```

```
enter the year: 2024
enter the month: 10
    October 2024
Mo Tu We Th Fr Sa Su
    1  2  3  4  5  6
 7  8  9 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30 31

                                  2024

      January               February              March
Mo Tu We Th Fr Sa Su  Mo Tu We Th Fr Sa Su  Mo Tu We Th Fr Sa Su
 1  2  3  4  5  6  7                  1  2  3  4                  1  2  3
 8  9 10 11 12 13 14   5  6  7  8  9 10 11   4  5  6  7  8  9 10
15 16 17 18 19 20 21  12 13 14 15 16 17 18  11 12 13 14 15 16 17
22 23 24 25 26 27 28  19 20 21 22 23 24 25  18 19 20 21 22 23 24
29 30 31              26 27 28 29           25 26 27 28 29 30 31

      April                 May                   June
```

## Python Standard Data types:

## 1.

```
s= 'welcome_to_nmkrv'
print(s[3])
print(s[3:])
print(s[3:7])
print(s[:10])
print(s[-3])
print(s[-5:])
print(s[-5:-2])
```

```
c
come_to_nmkrv
come
welcome_to
k
nmkrv
nmk
```

**2.**

```
num =int(input("Enter the number: "))
if num>0:
    print(num," is +ve Number")
else:
    print(num,"is -ve Number")
```

```
Enter the number: 2
2  is +ve Number
```

## Python Pandas :

**1.**

```
import pandas as pd

calories = {"day1": 420, "day2": 380, "day3": 390}

myvar = pd.Series(calories)
print(myvar)

calories = {"day1": 420, "day2": 380, "day3": 390}

myvar = pd.Series(calories, index = ["day1", "day2","day3"])
print(myvar)
```

```
day1    420
day2    380
day3    390
dtype: int64
day1    420
day2    380
day3    390
dtype: int64
```

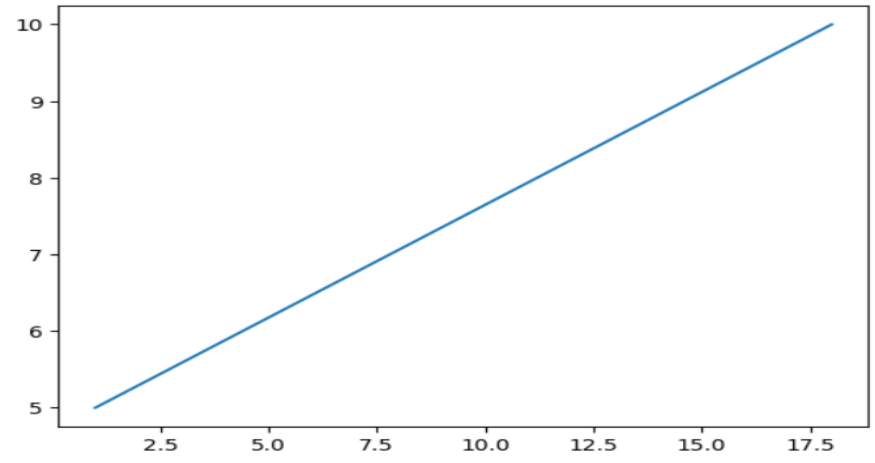
## 2.

```
import pandas as pd

df = pd.read_csv(r'C:\Users\Windows\Downloads\heart (1).csv',encoding='unicode_escape')

print(df)
```

```
     ï»¿age  sex  cp  trestbps  chol  fbs  restecg  thalach  exang  oldpeak  \
0         63    1   3       145   233    1        0      150      0      2.3
1         37    1   2       130   250    0        1      187      0      3.5
2         41    0   1       130   204    0        0      172      0      1.4
3         56    1   1       120   236    0        1      178      0      0.8
4         57    0   0       120   354    0        1      163      1      0.6
..       ...  ...  ..       ...   ...  ...      ...      ...    ...      ...
298       57    0   0       140   241    0        1      123      1      0.2
299       45    1   3       110   264    0        1      132      0      1.2
300       68    1   0       144   193    1        1      141      0      3.4
301       57    1   0       130   131    0        1      115      1      1.2
302       57    0   1       130   236    0        0      174      0      0.0

     slope  ca  thal  target
0        0   0     1       1
1        0   0     2       1
2        2   0     2       1
3        2   0     2       1
4        2   0     2       1
..     ...  ..   ...     ...
298      1   0     3       0
299      1   0     3       0
300      1   2     3       0
301      1   1     3       0
```

## Matplot Library:

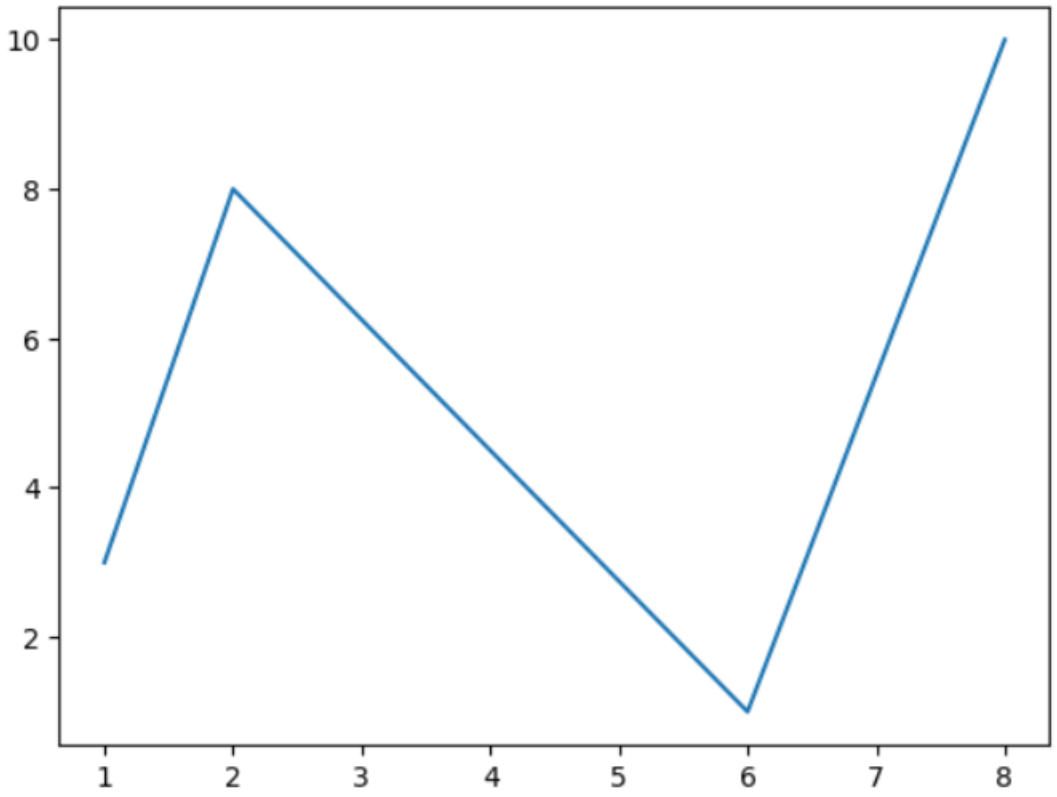
```
import matplotlib.pyplot as plt
import numpy as np
xpoint = np.array([1, 18])
ypoint = np.array([5, 10])
plt.plot(xpoint, ypoint)
plt.show()
```




✓ 0s completed at 2:22 PM

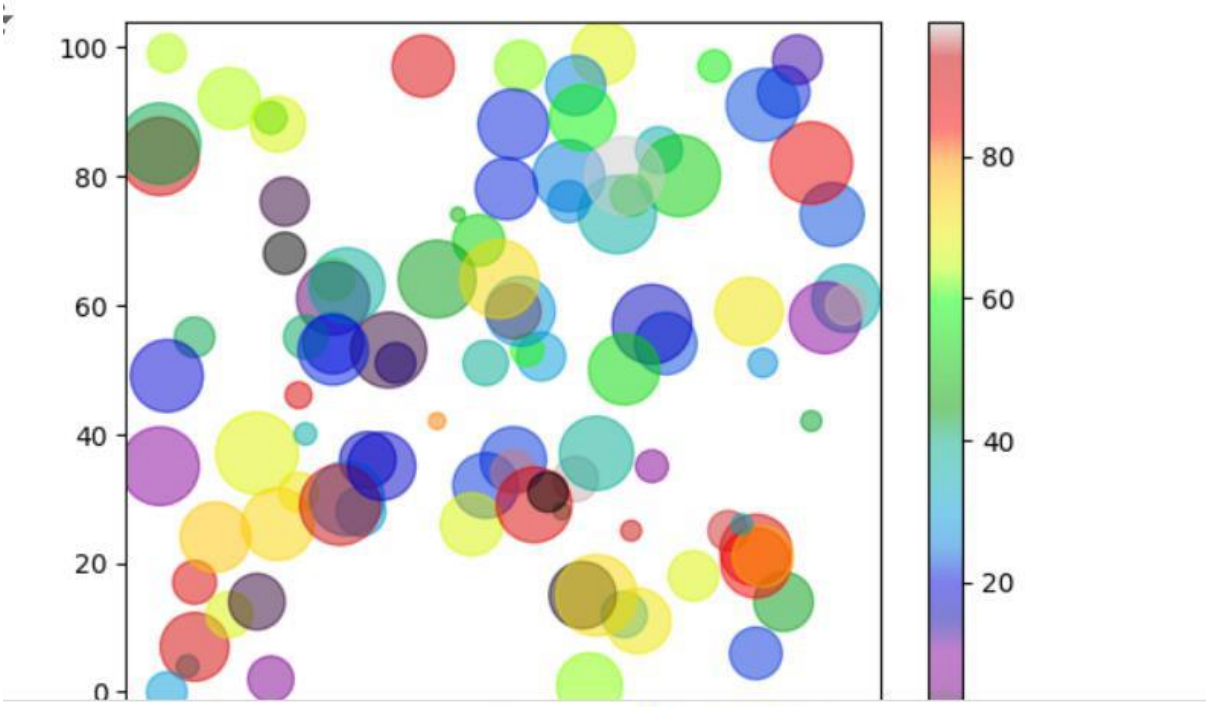
## Multiple Points:

```
import matplotlib.pyplot as plt
import numpy as np
xpoint = np.array([1, 2, 6, 8])
ypoint = np.array([3, 8, 1, 10])
plt.plot(xpoint, ypoint)
plt.show()
```



**Assignment:**

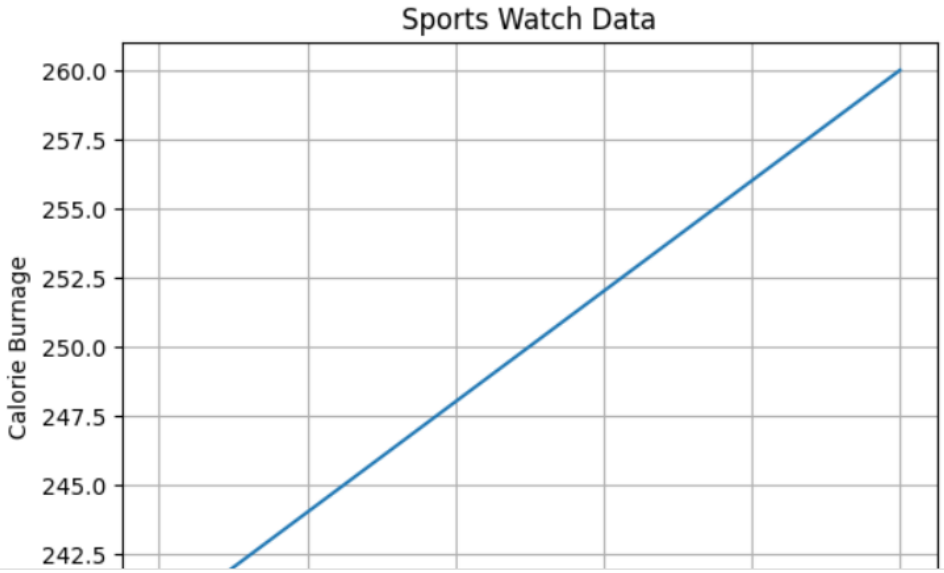
```
import matplotlib.pyplot as plt
import numpy as np

x = np.random.randint(100, size=(100))
y = np.random.randint(100, size=(100))
colors = np.random.randint(100, size=(100))
sizes = 10 * np.random.randint(100, size=(100))

plt.scatter(x, y, c=colors, s=sizes, alpha=0.5, cmap='nipy_spectral')

plt.colorbar()

plt.show()
```




✓ 1s completed at 2:26 PM

## Matplotlib Grid:

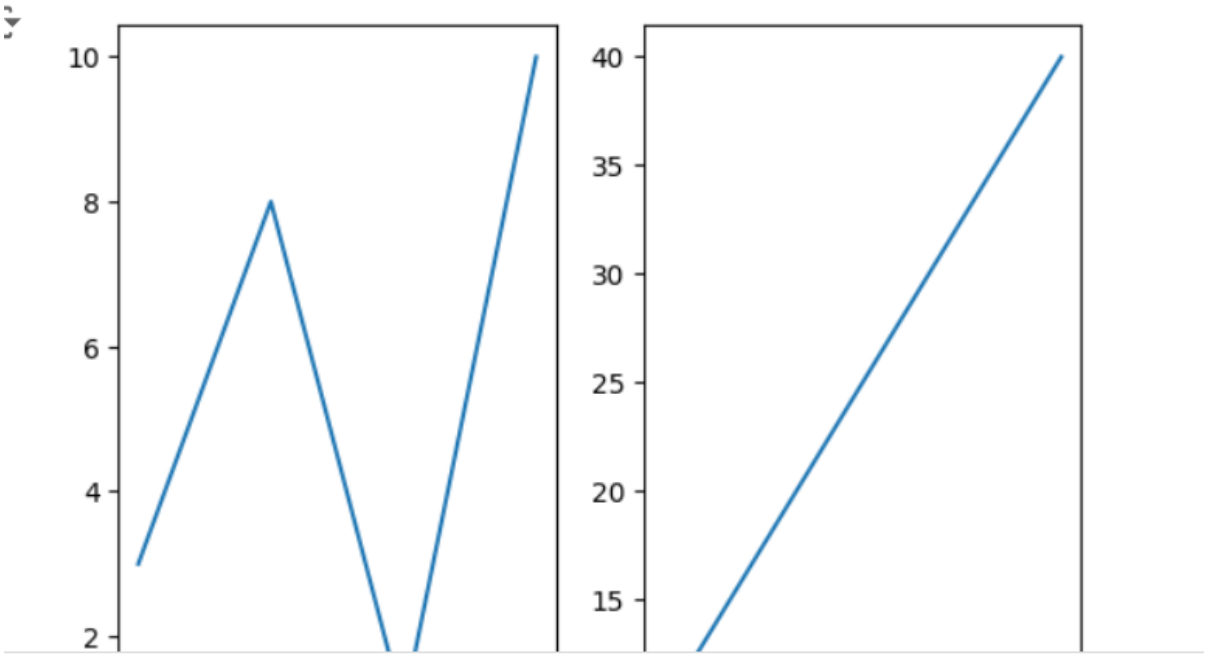
```
import numpy as np
import matplotlib.pyplot as plt

x = np.array([80, 85, 90])
y = np.array([240, 250, 260])

plt.title("Sports Watch Data")
plt.xlabel("Average Pulse")
plt.ylabel("Calorie Burnage")

plt.plot(x, y)
plt.grid()
plt.show()
```




✓ 0s completed at 2:28 PM

## Matplotlib Subplot:

```
import matplotlib.pyplot as plt
import numpy as np

#plot 1:
x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])
plt.subplot(1, 2, 1)
plt.plot(x,y)

#plot 2:
x = np.array([0, 1, 2, 3])
y = np.array([10, 20, 30, 40])
plt.subplot(1, 2, 2)
plt.plot(x,y)
plt.show()
```




✓ 0s completed at 2:29 PM

# House prize prediction

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


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

```
import numpy as np
import pandas as pd
import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))
```

```
import pandas as pd

df = pd.read_csv(r'C:\Users\lokes\Downloads\data.csv',encoding='unicode_escape')

print(df)
```

```
                  date         price  bedrooms  bathrooms  sqft_living  \
0     2014-05-02 00:00:00  3.130000e+05       3.0       1.50         1340
1     2014-05-02 00:00:00  2.384000e+06       5.0       2.50         3650
2     2014-05-02 00:00:00  3.420000e+05       3.0       2.00         1930
3     2014-05-02 00:00:00  4.200000e+05       3.0       2.25         2000
4     2014-05-02 00:00:00  5.500000e+05       4.0       2.50         1940
...                   ...           ...       ...        ...          ...
4595  2014-07-09 00:00:00  3.081667e+05       3.0       1.75         1510
4596  2014-07-09 00:00:00  5.343333e+05       3.0       2.50         1460
4597  2014-07-09 00:00:00  4.169042e+05       3.0       2.50         3010
4598  2014-07-10 00:00:00  2.034000e+05       4.0       2.00         2090
4599  2014-07-10 00:00:00  2.206000e+05       3.0       2.50         1490
```

```
      sqft_lot  floors  waterfront  view  condition  sqft_above  \
0         7912     1.5           0     0          3        1340
1         9050     2.0           0     4          5        3370
2        11947     1.0           0     0          4        1930
3         8030     1.0           0     0          4        1000
4        10500     1.0           0     0          4        1140
...        ...     ...         ...   ...        ...         ...
4595      6360     1.0           0     0          4        1510
4596      7573     2.0           0     0          3        1460
4597      7014     2.0           0     0          3        3010
4598      6630     1.0           0     0          3        1070
4599      8102     2.0           0     0          4        1490
...
4598    Seattle  WA 98178     USA
4599  Covington  WA 98042     USA
```

```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
from sklearn.preprocessing import StandardScaler
import matplotlib.pyplot as plt
import seaborn as sns
```

```
df = pd.read_csv(r'C:\Users\lokes\Downloads\data.csv')
df.head()
```

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

```
X = df[['sqft_living', 'bedrooms']]
y = df['price']
```

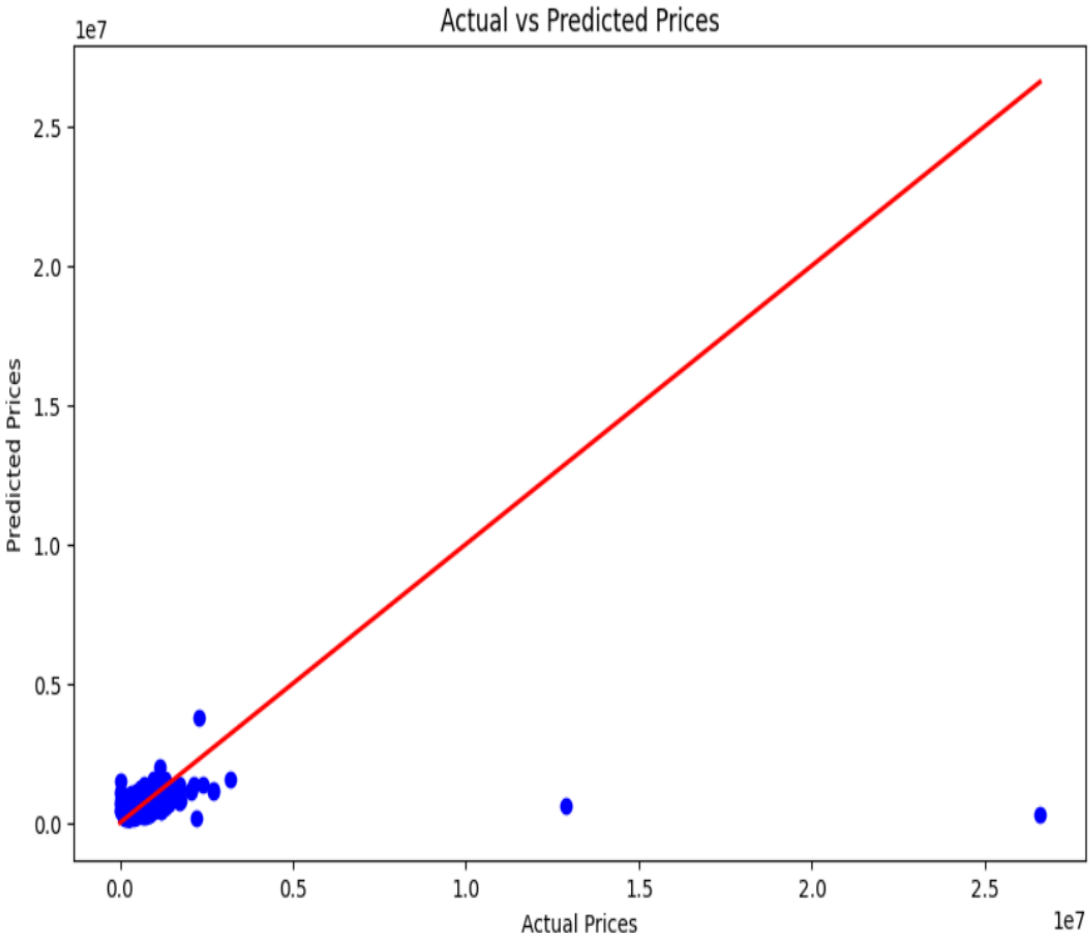
```
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

```
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, 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}')
```

```
plt.figure(figsize=(10, 6))
plt.scatter(y_test, y_pred, color='blue')
plt.plot([min(y_test), max(y_test)], [min(y_test), max(y_test)], color='red', linewidth=2)
plt.xlabel('Actual Prices')
plt.ylabel('Predicted Prices')
plt.title('Actual vs Predicted Prices')
plt.show()
```


Actual vs Predicted Prices
1e7
2.5
2.0
1.5
1.0
0.5
0.0
Predicted Prices
0.0
0.5
1.0
1.5
2.0
2.5
Actual Prices
1e7

## Conclusion:

The conclusion of Python, a widely used and versatile programming language, highlights its impact and benefits in various domains. Python is a highly effective and adaptable programming language that supports a wide array of applications and development practices. Its ease of use, strong community support, and versatility make it a valuable tool for both novice and seasoned programmers.

## References:


- https://www.kaggle.com/datasets/abisheksudarshan/health-care-analytics.
- https://www.kaggle.com/code/abisheksudarshan/health-care-analytics.