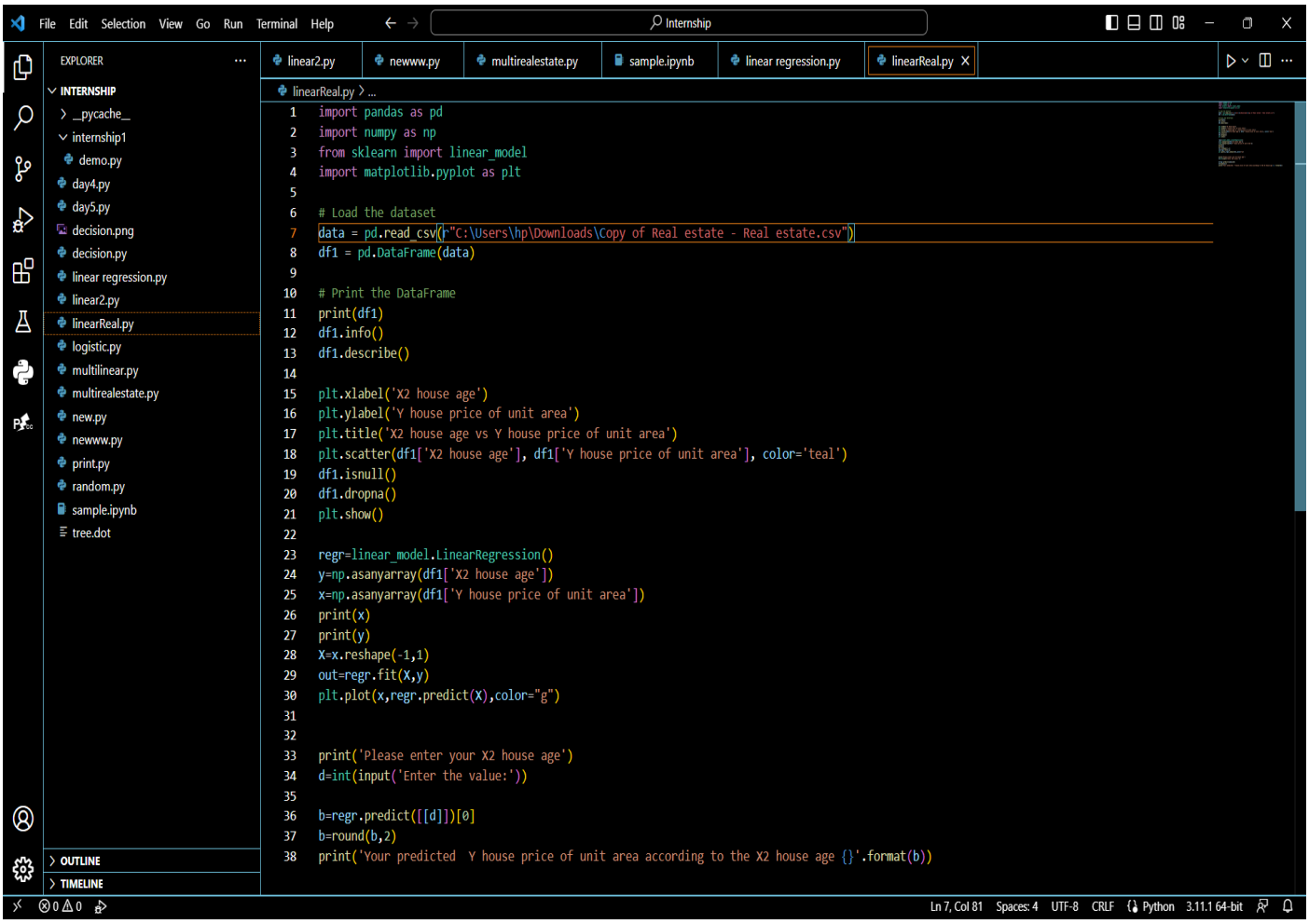
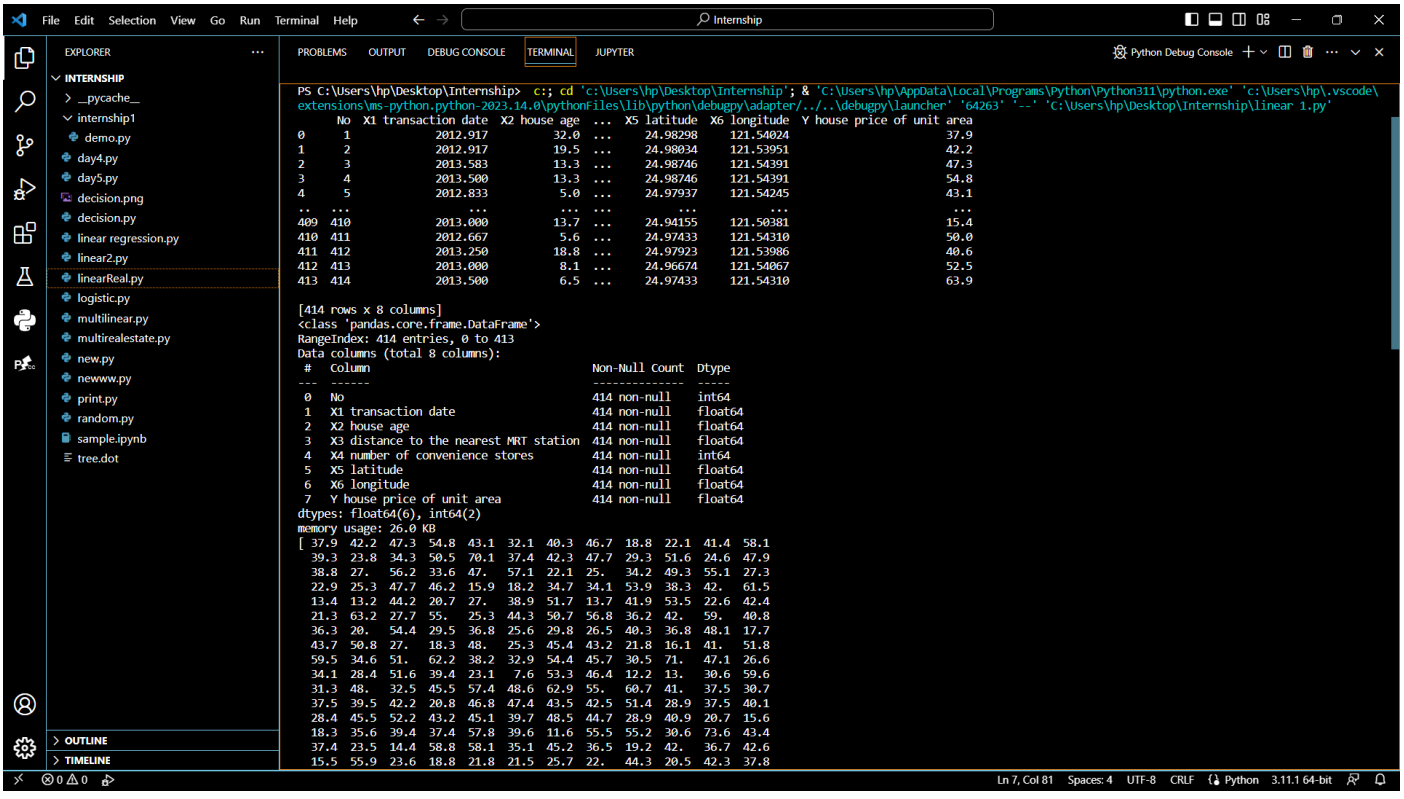
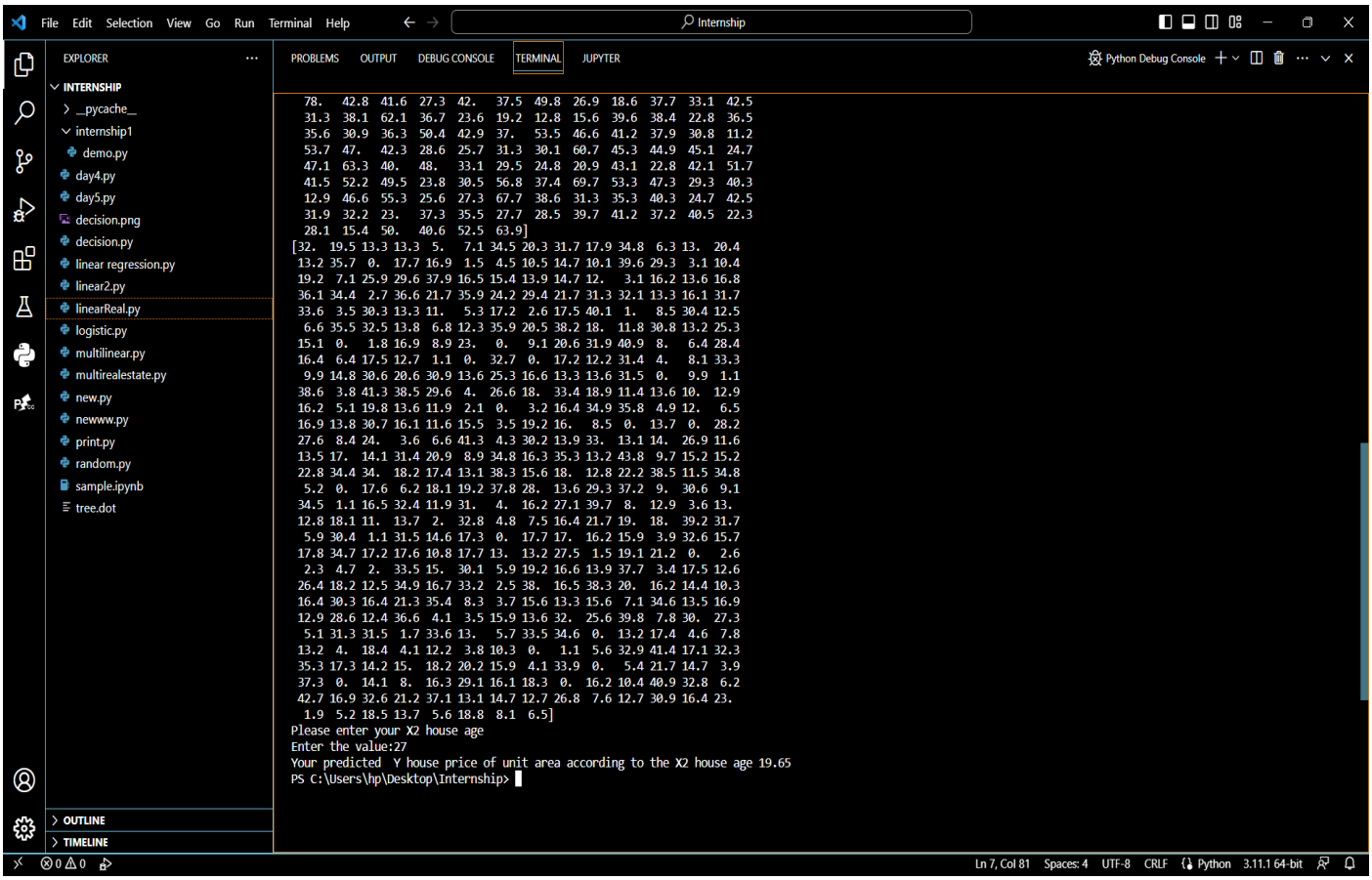
1.Linear regression



```
import pandas as pd
import numpy as np
from sklearn import linear_model
import matplotlib.pyplot as plt

# Load the dataset
data = pd.read_csv(r"C:\Users\hp\Downloads\Copy of Real estate - Real estate.csv")
df1 = pd.DataFrame(data)

# Print the DataFrame
print(df1)
df1.info()
df1.describe()

plt.xlabel('X2 house age')
plt.ylabel('Y house price of unit area')
plt.title('X2 house age vs Y house price of unit area')
plt.scatter(df1['X2 house age'], df1['Y house price of unit area'], color='teal')
df1.isnull()
df1.dropna()
plt.show()

regr=linear_model.LinearRegression()
y=np.asanyarray(df1['X2 house age'])
x=np.asanyarray(df1['Y house price of unit area'])
print(x)
print(y)
X=x.reshape(-1,1)
out=regr.fit(X,y)
plt.plot(x,regr.predict(X),color="g")


print('Please enter your X2 house age')
d=int(input('Enter the value:'))

b=regr.predict([[d]])[0]
b=round(b,2)
print('Your predicted  Y house price of unit area according to the X2 house age {}'.format(b))
```

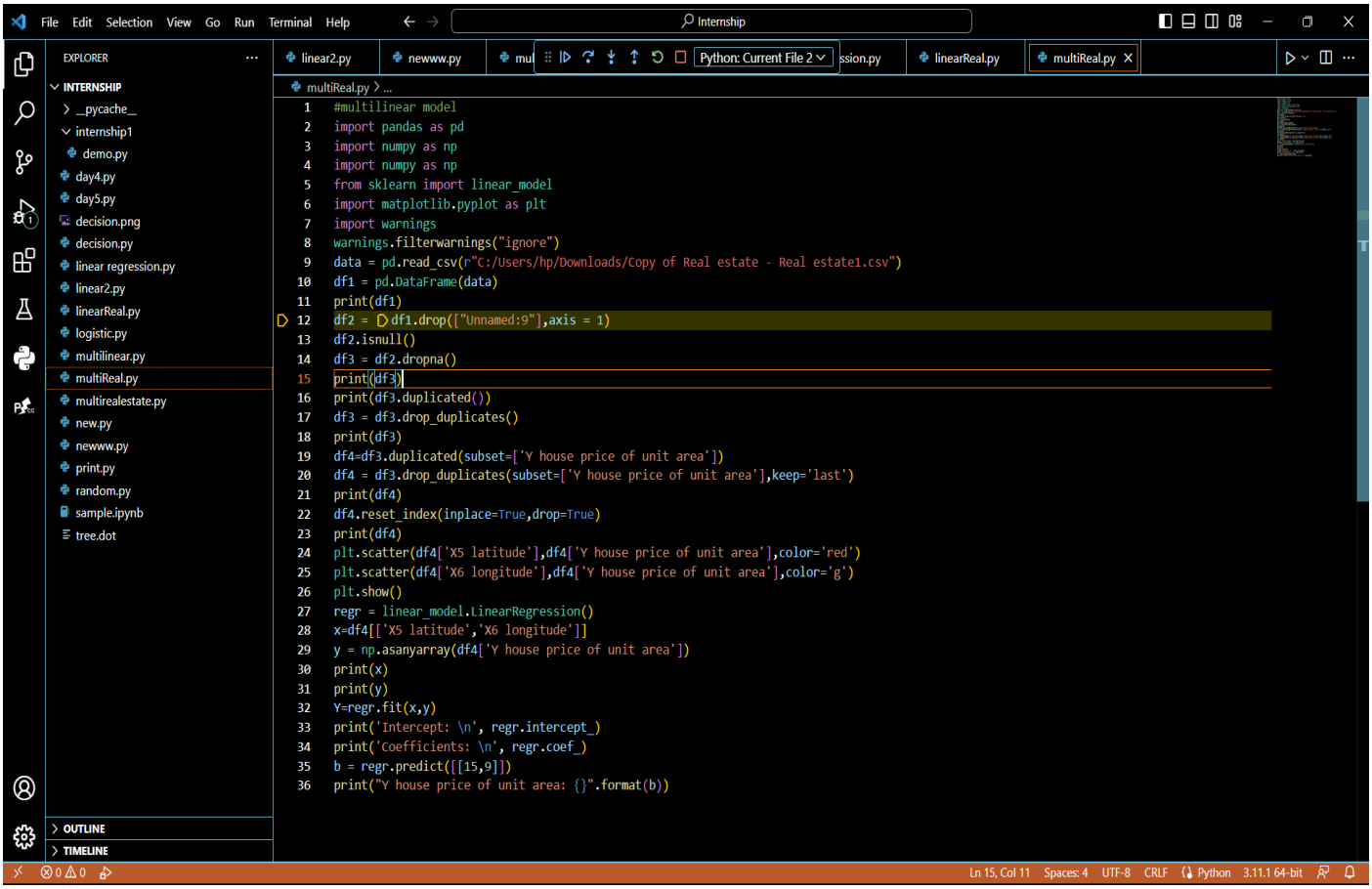


```
PS C:\Users\hp\Desktop\Internship>  c:; cd 'c:\Users\hp\Desktop\Internship'; & 'C:\Users\hp\AppData\Local\Programs\Python\Python311\python.exe' 'c:\Users\hp\.vscode\
extensions\ms-python.python-2023.14.0\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '64263' '--' 'C:\Users\hp\Desktop\Internship\linear 1.py'
      No  X1 transaction date  X2 house age  ...  X5 latitude  X6 longitude  Y house price of unit area
0      1            2012.917          32.0  ...     24.98298     121.54024                        37.9
1      2            2012.917          19.5  ...     24.98034     121.53951                        42.2
2      3            2013.583          13.3  ...     24.98746     121.54391                        47.3
3      4            2013.500          13.3  ...     24.98746     121.54391                        54.8
4      5            2012.833           5.0  ...     24.97937     121.54245                        43.1
..   ...                 ...           ...  ...          ...           ...                         ...
409  410            2013.000          13.7  ...     24.94155     121.50381                        15.4
410  411            2012.667           5.6  ...     24.97433     121.54310                        50.0
411  412            2013.250          18.8  ...     24.97923     121.53986                        40.6
412  413            2013.000           8.1  ...     24.96674     121.54067                        52.5
413  414            2013.500           6.5  ...     24.97433     121.54310                        63.9

[414 rows x 8 columns]
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 414 entries, 0 to 413
Data columns (total 8 columns):
 #   Column                                  Non-Null Count  Dtype
---  ------                                  --------------  -----
 0   No                                      414 non-null    int64
 1   X1 transaction date                     414 non-null    float64
 2   X2 house age                            414 non-null    float64
 3   X3 distance to the nearest MRT station  414 non-null    float64
 4   X4 number of convenience stores         414 non-null    int64
 5   X5 latitude                             414 non-null    float64
 6   X6 longitude                            414 non-null    float64
 7   Y house price of unit area              414 non-null    float64
dtypes: float64(6), int64(2)
memory usage: 26.0 KB
[ 37.9  42.2  47.3  54.8  43.1  32.1  40.3  46.7  18.8  22.1  41.4  58.1
  39.3  23.8  34.3  50.5  70.1  37.4  42.3  47.7  29.3  51.6  24.6  47.9
  38.8  27.   56.2  33.6  47.   57.1  22.1  25.   34.2  49.3  55.1  27.3
  22.9  25.3  47.7  46.2  15.9  18.2  34.7  34.1  53.9  38.3  42.   61.5
  13.4  13.2  44.2  20.7  27.   38.9  51.7  13.7  41.9  53.5  22.6  42.4
  21.3  63.2  27.7  55.   25.3  44.3  50.7  56.8  36.2  42.   59.   40.8
  36.3  20.   54.4  29.5  36.8  25.6  29.8  26.5  40.3  36.8  48.1  17.7
  43.7  50.8  27.   18.3  48.   25.3  45.4  43.2  21.8  16.1  41.   51.8
  59.5  34.6  51.   62.2  38.2  32.9  54.4  45.7  30.5  71.   47.1  26.6
  34.1  28.4  51.6  39.4  23.1   7.6  53.3  46.4  12.2  13.   30.6  59.6
  31.3  48.   32.5  45.5  57.4  48.6  62.9  55.   60.7  41.   37.5  30.7
  37.5  39.5  42.2  20.8  46.8  47.4  43.5  42.5  51.4  28.9  37.5  40.1
  28.4  45.5  52.2  43.2  45.1  39.7  48.5  44.7  28.9  40.9  20.7  15.6
  18.3  35.6  39.4  37.4  57.8  39.6  11.6  55.5  55.2  30.6  73.6  43.4
  37.4  23.5  14.4  58.8  58.1  35.1  45.2  36.5  19.2  42.   36.7  42.6
  15.5  55.9  23.6  18.8  21.8  21.5  25.7  22.   44.3  20.5  42.3  37.8
```


File Edit Selection View Go Run Terminal Help
Internship
EXPLORER
INTERNSHIP
__pycache__
internship1
demo.py
day4.py
day5.py
decision.png
decision.py
linear regression.py
linear2.py
linearReal.py
logistic.py
multilinear.py
multirealestate.py
new.py
newww.py
print.py
random.py
sample.ipynb
tree.dot
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
Python Debug Console
78. 42.8 41.6 27.3 42. 37.5 49.8 26.9 18.6 37.7 33.1 42.5
31.3 38.1 62.1 36.7 23.6 19.2 12.8 15.6 39.6 38.4 22.8 36.5
35.6 30.9 36.3 50.4 42.9 37. 53.5 46.6 41.2 37.9 30.8 11.2
53.7 47. 42.3 28.6 25.7 31.3 30.1 60.7 45.3 44.9 45.1 24.7
47.1 63.3 40. 48. 33.1 29.5 24.8 20.9 43.1 22.8 42.1 51.7
41.5 52.2 49.5 23.8 30.5 56.8 37.4 69.7 53.3 47.3 29.3 40.3
12.9 46.6 55.3 25.6 27.3 67.7 38.6 31.3 35.3 40.3 24.7 42.5
31.9 32.2 23. 37.3 35.5 27.7 28.5 39.7 41.2 37.2 40.5 22.3
28.1 15.4 50. 40.6 52.5 63.9]
[32. 19.5 13.3 13.3 5. 7.1 34.5 20.3 31.7 17.9 34.8 6.3 13. 20.4
13.2 35.7 0. 17.7 16.9 1.5 4.5 10.5 14.7 10.1 39.6 29.3 3.1 10.4
19.2 7.1 25.9 29.6 37.9 16.5 15.4 13.9 14.7 12. 3.1 16.2 13.6 16.8
36.1 34.4 2.7 36.6 21.7 35.9 24.2 29.4 21.7 31.3 32.1 13.3 16.1 31.7
33.6 3.5 30.3 13.3 11. 5.3 17.2 2.6 17.5 40.1 1. 8.5 30.4 12.5
6.6 35.5 32.5 13.8 6.8 12.3 35.9 20.5 38.2 18. 11.8 30.8 13.2 25.3
15.1 0. 1.8 16.9 8.9 23. 0. 9.1 20.6 31.9 40.9 8. 6.4 28.4
16.4 6.4 17.5 12.7 1.1 0. 32.7 0. 17.2 12.2 31.4 4. 8.1 33.3
9.9 14.8 30.6 20.6 30.9 13.6 25.3 16.6 13.3 13.6 31.5 0. 9.9 1.1
38.6 3.8 41.3 38.5 29.6 4. 26.6 18. 33.4 18.9 11.4 13.6 10. 12.9
16.2 5.1 19.8 13.6 11.9 2.1 0. 3.2 16.4 34.9 35.8 4.9 12. 6.5
16.9 13.8 30.7 16.1 11.6 15.5 3.5 19.2 16. 8.5 0. 13.7 0. 28.2
27.6 8.4 24. 3.6 6.6 41.3 4.3 30.2 13.9 33. 13.1 14. 26.9 11.6
13.5 17. 14.1 31.4 20.9 8.9 34.8 16.3 35.3 13.2 43.8 9.7 15.2 15.2
22.8 34.4 34. 18.2 17.4 13.1 38.3 15.6 18. 12.8 22.2 38.5 11.5 34.8
5.2 0. 17.6 6.2 18.1 19.2 37.8 28. 13.6 29.3 37.2 9. 30.6 9.1
34.5 1.1 16.5 32.4 11.9 31. 4. 16.2 27.1 39.7 8. 12.9 3.6 13.
12.8 18.1 11. 13.7 2. 32.8 4.8 7.5 16.4 21.7 19. 18. 39.2 31.7
5.9 30.4 1.1 31.5 14.6 17.3 0. 17.7 17. 16.2 15.9 3.9 32.6 15.7
17.8 34.7 17.2 17.6 10.8 17.7 13. 13.2 27.5 1.5 19.1 21.2 0. 2.6
2.3 4.7 2. 33.5 15. 30.1 5.9 19.2 16.6 13.9 37.7 3.4 17.5 12.6
26.4 18.2 12.5 34.9 16.7 33.2 2.5 38. 16.5 38.3 20. 16.2 14.4 10.3
16.4 30.3 16.4 21.3 35.4 8.3 3.7 15.6 13.3 15.6 7.1 34.6 13.5 16.9
12.9 28.6 12.4 36.6 4.1 3.5 15.9 13.6 32. 25.6 39.8 7.8 30. 27.3
5.1 31.3 31.5 1.7 33.6 13. 5.7 33.5 34.6 0. 13.2 17.4 4.6 7.8
13.2 4. 18.4 4.1 12.2 3.8 10.3 0. 1.1 5.6 32.9 41.4 17.1 32.3
35.3 17.3 14.2 15. 18.2 20.2 15.9 4.1 33.9 0. 5.4 21.7 14.7 3.9
37.3 0. 14.1 8. 16.3 29.1 16.1 18.3 0. 16.2 10.4 40.9 32.8 6.2
42.7 16.9 32.6 21.2 37.1 13.1 14.7 12.7 26.8 7.6 12.7 30.9 16.4 23.
1.9 5.2 18.5 13.7 5.6 18.8 8.1 6.5]
Please enter your X2 house age
Enter the value:27
Your predicted Y house price of unit area according to the X2 house age 19.65
PS C:\Users\hp\Desktop\Internship>
Ln 7, Col 81 Spaces: 4 UTF-8 CRLF Python 3.11.1 64-bit

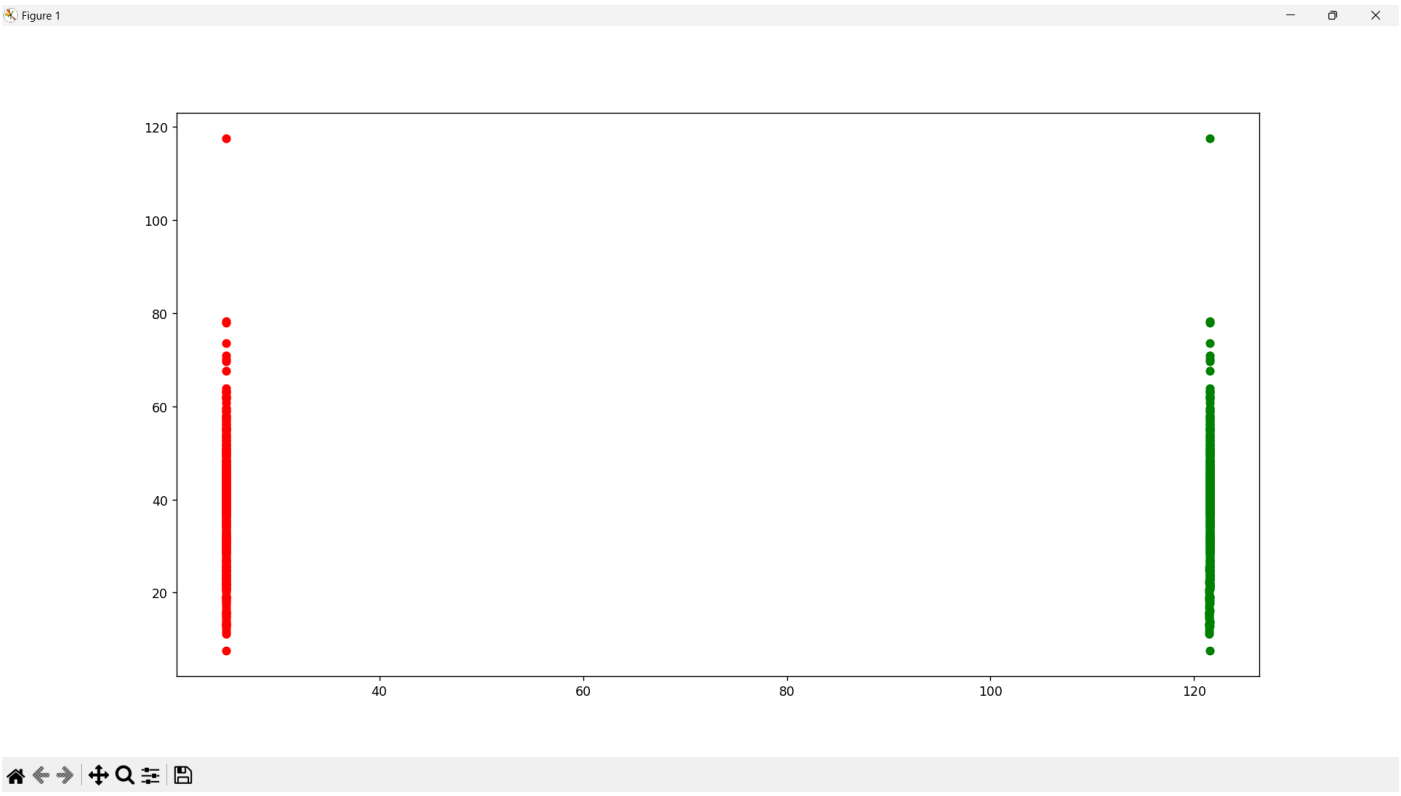

File Edit Selection View Go Run Terminal Help
Internship
EXPLORER
linear2.py
newww.py
multirealestate.py
linear regression.py
linearReal.py
Figure 1
X2 house age vs Y house price of unit area
Y house price of unit area
X2 house age
0 10 20 30 40
20 40 60 80 100 120
y of Real estate - Real estate.csv")
it area')
price of unit area'], color='teal')
a'])
Column Non-Null Count Dtype
0 No 414 non-null int64
1 X1 transaction date 414 non-null float64
2 X2 house age 414 non-null float64
3 X3 distance to the nearest MRT station 414 non-null float64
4 X4 number of convenience stores 414 non-null int64
5 X5 latitude 414 non-null float64
6 X6 longitude 414 non-null float64
7 Y house price of unit area 414 non-null float64
dtypes: float64(6), int64(2)
memory usage: 26.0 KB
Python Deb...
Python
OUTLINE
TIMELINE
Ln 7, Col 81 Spaces: 4 UTF-8 CRLF Python 3.11.1 64-bit

2.Multilinear regression

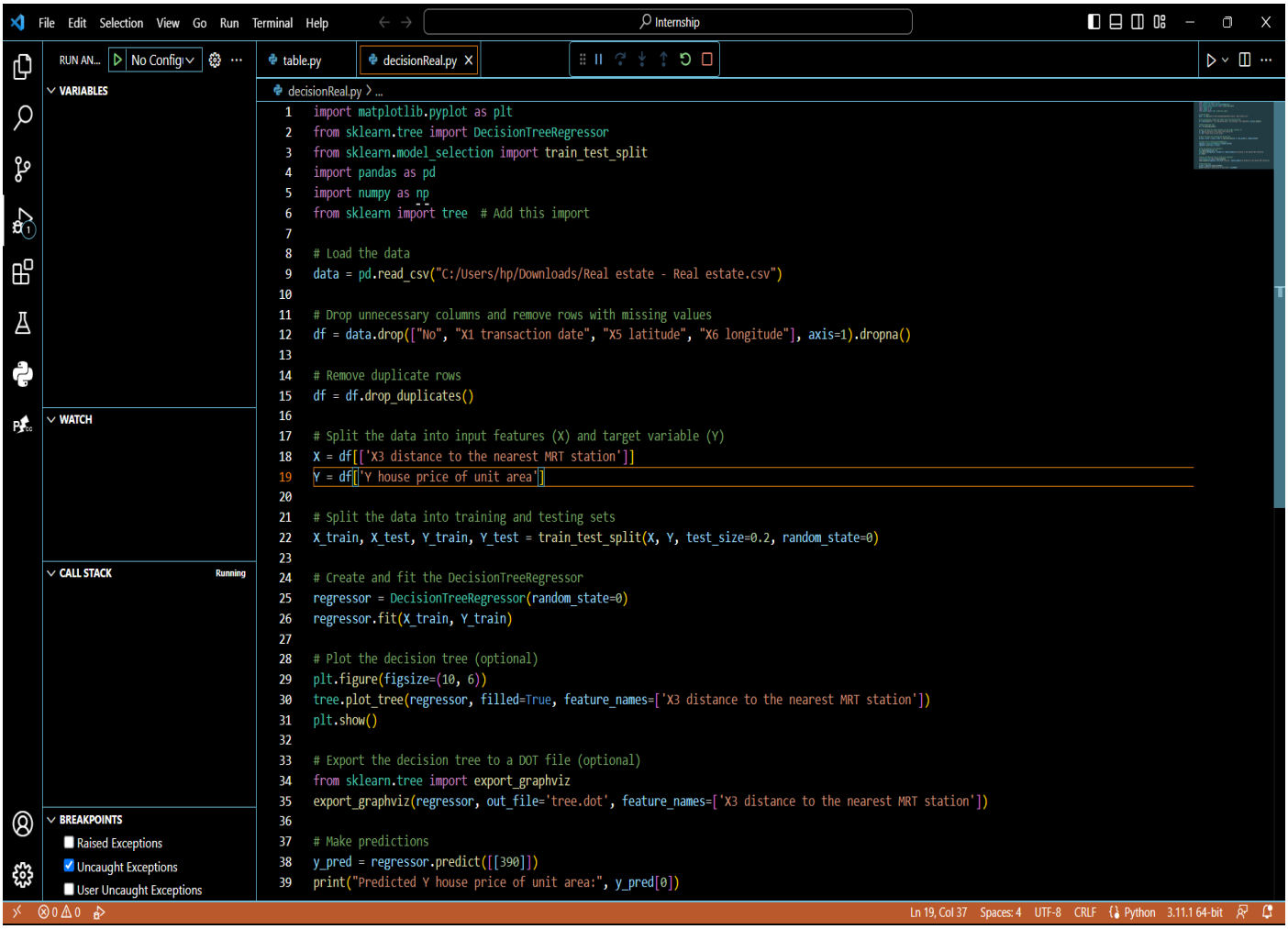


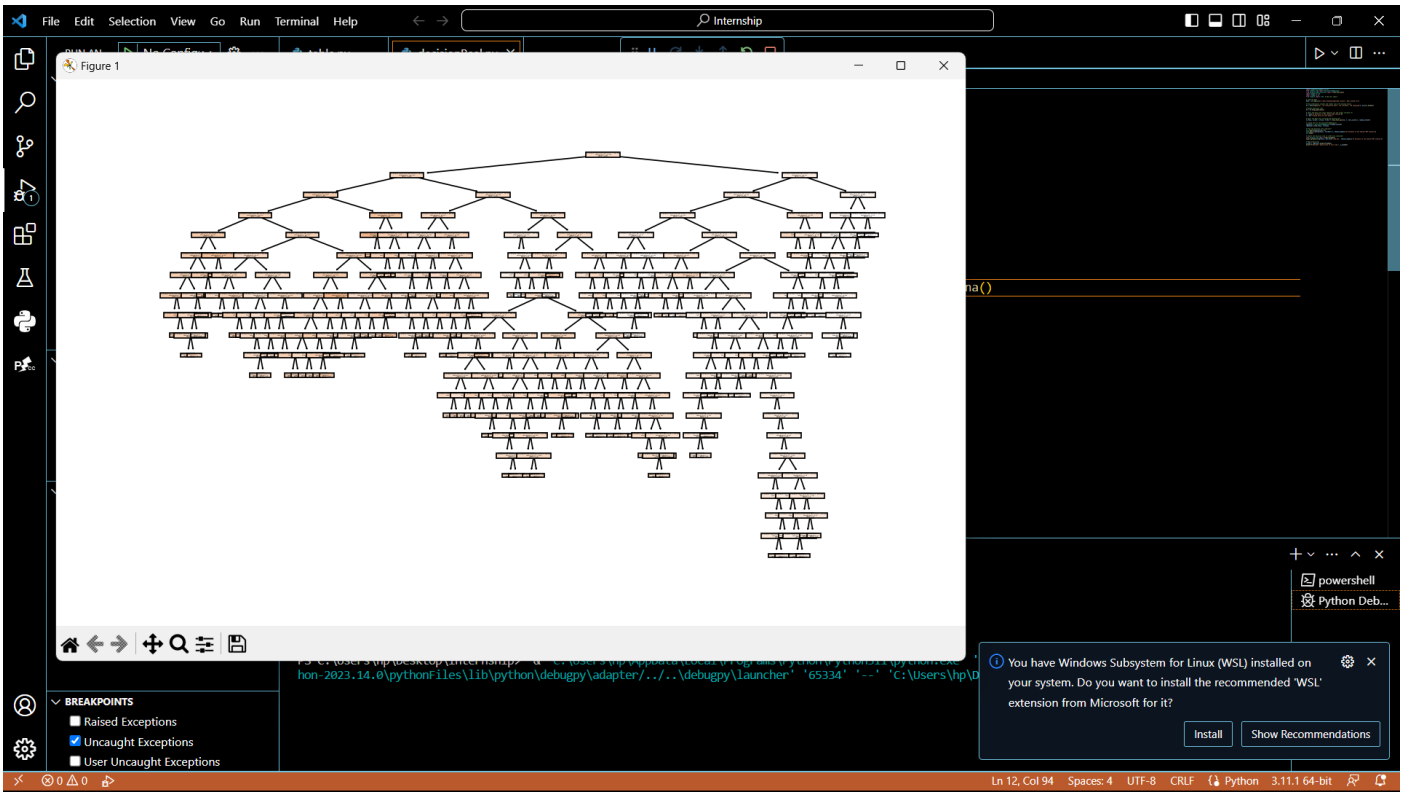
```
#multilinear model
import pandas as pd
import numpy as np
import numpy as np
from sklearn import linear_model
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings("ignore")
data = pd.read_csv(r"C:/Users/hp/Downloads/Copy of Real estate - Real estate1.csv")
df1 = pd.DataFrame(data)
print(df1)
df2 = df1.drop(["Unnamed:9"],axis = 1)
df2.isnull()
df3 = df2.dropna()
print(df3)
print(df3.duplicated())
df3 = df3.drop_duplicates()
print(df3)
df4=df3.duplicated(subset=['Y house price of unit area'])
df4 = df3.drop_duplicates(subset=['Y house price of unit area'],keep='last')
print(df4)
df4.reset_index(inplace=True,drop=True)
print(df4)
plt.scatter(df4['X5 latitude'],df4['Y house price of unit area'],color='red')
plt.scatter(df4['X6 longitude'],df4['Y house price of unit area'],color='g')
plt.show()
regr = linear_model.LinearRegression()
x=df4[['X5 latitude','X6 longitude']]
y = np.asanyarray(df4['Y house price of unit area'])
print(x)
print(y)
Y=regr.fit(x,y)
print('Intercept: \n', regr.intercept_)
print('Coefficients: \n', regr.coef_)
b = regr.predict([[15,9]])
print("Y house price of unit area: {}".format(b))
```

```
      No  X1 transaction date  X2 house age  ...  X5 latitude  X6 longitude  Y house price of unit area
0      1             2012.917          32.0  ...     24.98298     121.54024                        37.9
1      2             2012.917          19.5  ...     24.98034     121.53951                        42.2
2      3             2013.583          13.3  ...     24.98746     121.54391                        47.3
3      4             2013.500          13.3  ...     24.98746     121.54391                        54.8
4      5             2012.833           5.0  ...     24.97937     121.54245                        43.1
..   ...                  ...           ...  ...          ...           ...                         ...
409  410             2013.000          13.7  ...     24.94155     121.50381                        15.4
410  411             2012.667           5.6  ...     24.97433     121.54310                        50.0
411  412             2013.250          18.8  ...     24.97923     121.53986                        40.6
412  413             2013.000           8.1  ...     24.96674     121.54067                        52.5
413  414             2013.500           6.5  ...     24.97433     121.54310                        63.9

[414 rows x 8 columns]
      No  X1 transaction date  X2 house age  ...  X5 latitude  X6 longitude  Y house price of unit area
3      4             2013.500          13.3  ...     24.98746     121.54391                        54.8
5      6             2012.667           7.1  ...     24.96305     121.51254                        32.1
7      8             2013.417          20.3  ...     24.98042     121.54228                        46.7
14    15             2013.500          13.2  ...     24.99156     121.53406                        34.3
15    16             2013.583          35.7  ...     24.98240     121.54619                        50.5
..   ...                  ...           ...  ...          ...           ...                         ...
409  410             2013.000          13.7  ...     24.94155     121.50381                        15.4
410  411             2012.667           5.6  ...     24.97433     121.54310                        50.0
411  412             2013.250          18.8  ...     24.97923     121.53986                        40.6
412  413             2013.000           8.1  ...     24.96674     121.54067                        52.5
413  414             2013.500           6.5  ...     24.97433     121.54310                        63.9

[270 rows x 8 columns]
      No  X1 transaction date  X2 house age  ...  X5 latitude  X6 longitude  Y house price of unit area
0      4             2013.500          13.3  ...     24.98746     121.54391                        54.8
1      6             2012.667           7.1  ...     24.96305     121.51254                        32.1
2      8             2013.417          20.3  ...     24.98042     121.54228                        46.7
3     15             2013.500          13.2  ...     24.99156     121.53406                        34.3
4     16             2013.583          35.7  ...     24.98240     121.54619                        50.5
..   ...                  ...           ...  ...          ...           ...                         ...
265  410             2013.000          13.7  ...     24.94155     121.50381                        15.4
266  411             2012.667           5.6  ...     24.97433     121.54310                        50.0
267  412             2013.250          18.8  ...     24.97923     121.53986                        40.6
268  413             2013.000           8.1  ...     24.96674     121.54067                        52.5
269  414             2013.500           6.5  ...     24.97433     121.54310                        63.9

[270 rows x 8 columns]
    X5 latitude  X6 longitude
0      24.98746     121.54391
1      24.96305     121.51254
2      24.98042     121.54228
3      24.99156     121.53406
```

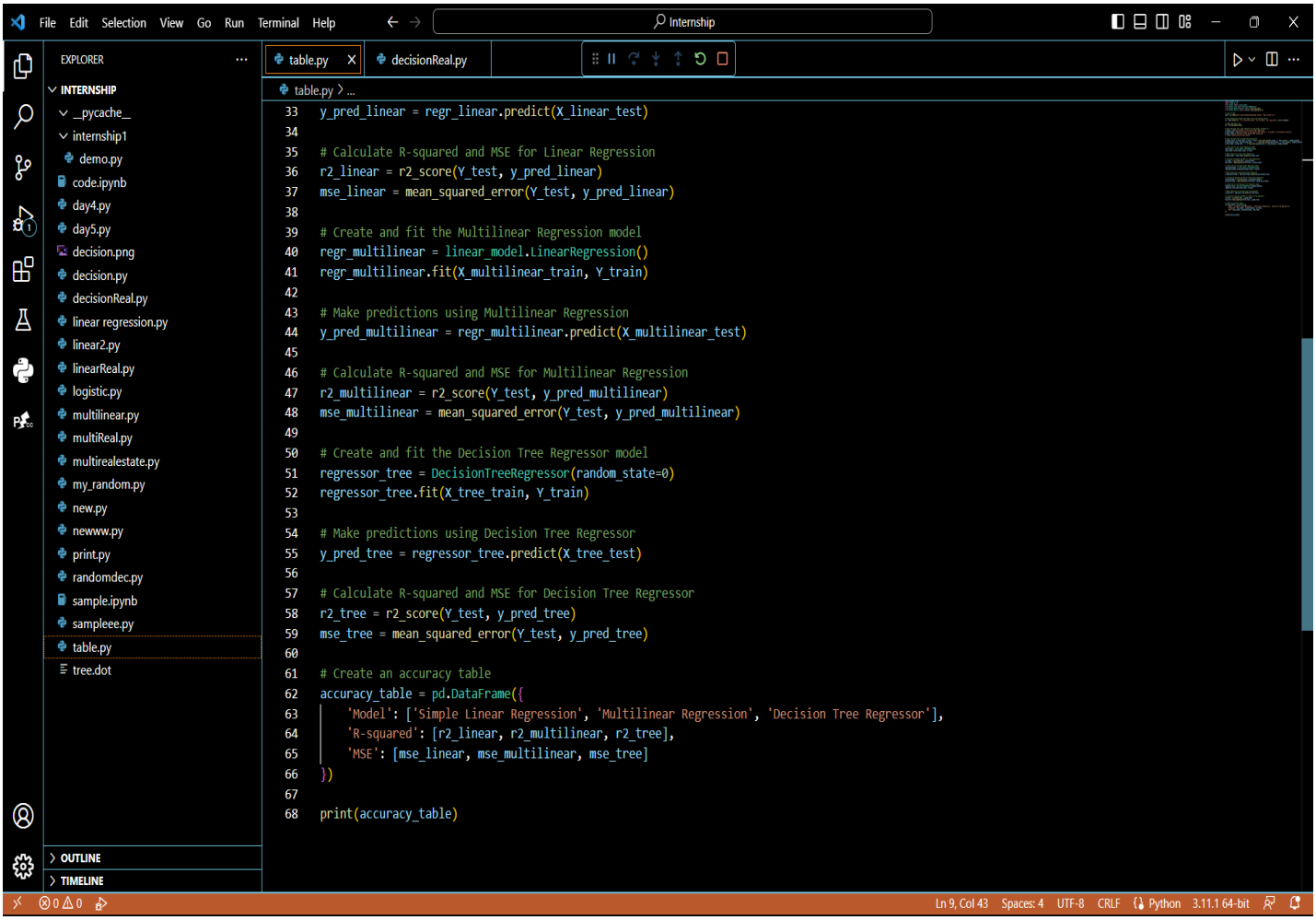


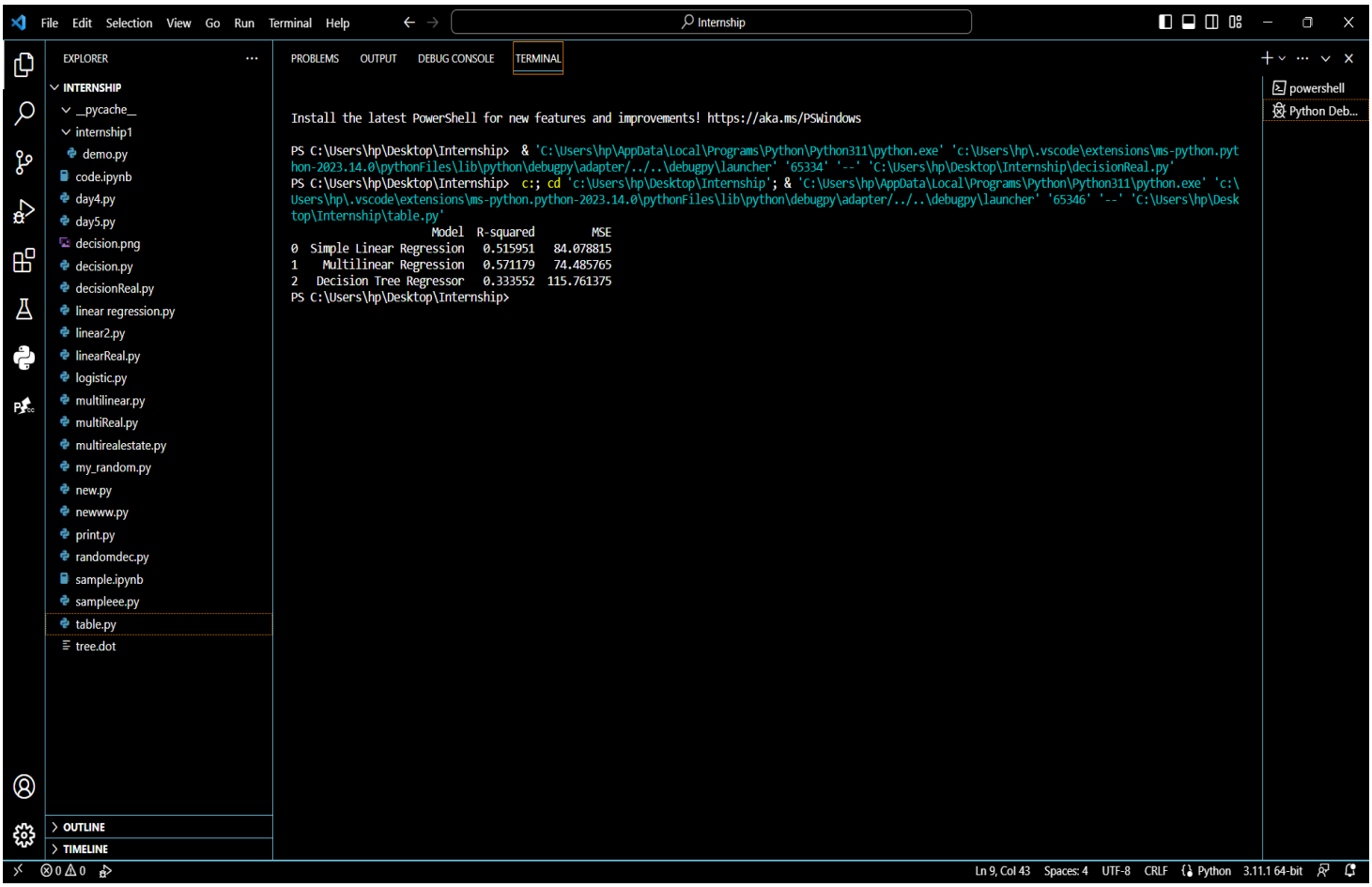
3.Decision tree




File Edit Selection View Go Run Terminal Help
Internship
Figure 1
BREAKPOINTS
Raised Exceptions
Uncaught Exceptions
User Uncaught Exceptions
powershell
Python Deb...
You have Windows Subsystem for Linux (WSL) installed on your system. Do you want to install the recommended 'WSL' extension from Microsoft for it?
Install
Show Recommendations
Ln 12, Col 94 Spaces: 4 UTF-8 CRLF Python 3.11.1 64-bit

4.Table


File Edit Selection View Go Run Terminal Help
Internship
EXPLORER
INTERNSHIP
pycache
internship1
demo.py
code.ipynb
day4.py
day5.py
decision.png
decision.py
decisionReal.py
linear regression.py
linear2.py
linearReal.py
logistic.py
multilinear.py
multiReal.py
multirealestate.py
my_random.py
new.py
newww.py
print.py
randomdec.py
sample.ipynb
sampleee.py
table.py
tree.dot
OUTLINE
TIMELINE
table.py
decisionReal.py
table.py > ...
33 y_pred_linear = regr_linear.predict(X_linear_test)
34
35 # Calculate R-squared and MSE for Linear Regression
36 r2_linear = r2_score(Y_test, y_pred_linear)
37 mse_linear = mean_squared_error(Y_test, y_pred_linear)
38
39 # Create and fit the Multilinear Regression model
40 regr_multilinear = linear_model.LinearRegression()
41 regr_multilinear.fit(X_multilinear_train, Y_train)
42
43 # Make predictions using Multilinear Regression
44 y_pred_multilinear = regr_multilinear.predict(X_multilinear_test)
45
46 # Calculate R-squared and MSE for Multilinear Regression
47 r2_multilinear = r2_score(Y_test, y_pred_multilinear)
48 mse_multilinear = mean_squared_error(Y_test, y_pred_multilinear)
49
50 # Create and fit the Decision Tree Regressor model
51 regressor_tree = DecisionTreeRegressor(random_state=0)
52 regressor_tree.fit(X_tree_train, Y_train)
53
54 # Make predictions using Decision Tree Regressor
55 y_pred_tree = regressor_tree.predict(X_tree_test)
56
57 # Calculate R-squared and MSE for Decision Tree Regressor
58 r2_tree = r2_score(Y_test, y_pred_tree)
59 mse_tree = mean_squared_error(Y_test, y_pred_tree)
60
61 # Create an accuracy table
62 accuracy_table = pd.DataFrame({
63 'Model': ['Simple Linear Regression', 'Multilinear Regression', 'Decision Tree Regressor'],
64 'R-squared': [r2_linear, r2_multilinear, r2_tree],
65 'MSE': [mse_linear, mse_multilinear, mse_tree]
66 })
67
68 print(accuracy_table)
Ln 9, Col 43 Spaces: 4 UTF-8 CRLF Python 3.11.1 64-bit


File Edit Selection View Go Run Terminal Help
Internship
EXPLORER
INTERNSHIP
__pycache__
internship1
demo.py
code.ipynb
day4.py
day5.py
decision.png
decision.py
decisionReal.py
linear regression.py
linear2.py
linearReal.py
logistic.py
multilinear.py
multiReal.py
multirealestate.py
my_random.py
new.py
newww.py
print.py
randomdec.py
sample.ipynb
sampleee.py
table.py
tree.dot
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
powershell
Python Deb...
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows
PS C:\Users\hp\Desktop\Internship> & 'C:\Users\hp\AppData\Local\Programs\Python\Python311\python.exe' 'c:\Users\hp\.vscode\extensions\ms-python.python-2023.14.0\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '65334' '--' 'C:\Users\hp\Desktop\Internship\decisionReal.py'
PS C:\Users\hp\Desktop\Internship> c:; cd 'c:\Users\hp\Desktop\Internship'; & 'C:\Users\hp\AppData\Local\Programs\Python\Python311\python.exe' 'c:\Users\hp\.vscode\extensions\ms-python.python-2023.14.0\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '65346' '--' 'C:\Users\hp\Desktop\Internship\table.py'
Model R-squared MSE
0 Simple Linear Regression 0.515951 84.078815
1 Multilinear Regression 0.571179 74.485765
2 Decision Tree Regressor 0.333552 115.761375
PS C:\Users\hp\Desktop\Internship>
Ln 9, Col 43 Spaces: 4 UTF-8 CRLF Python 3.11.1 64-bit