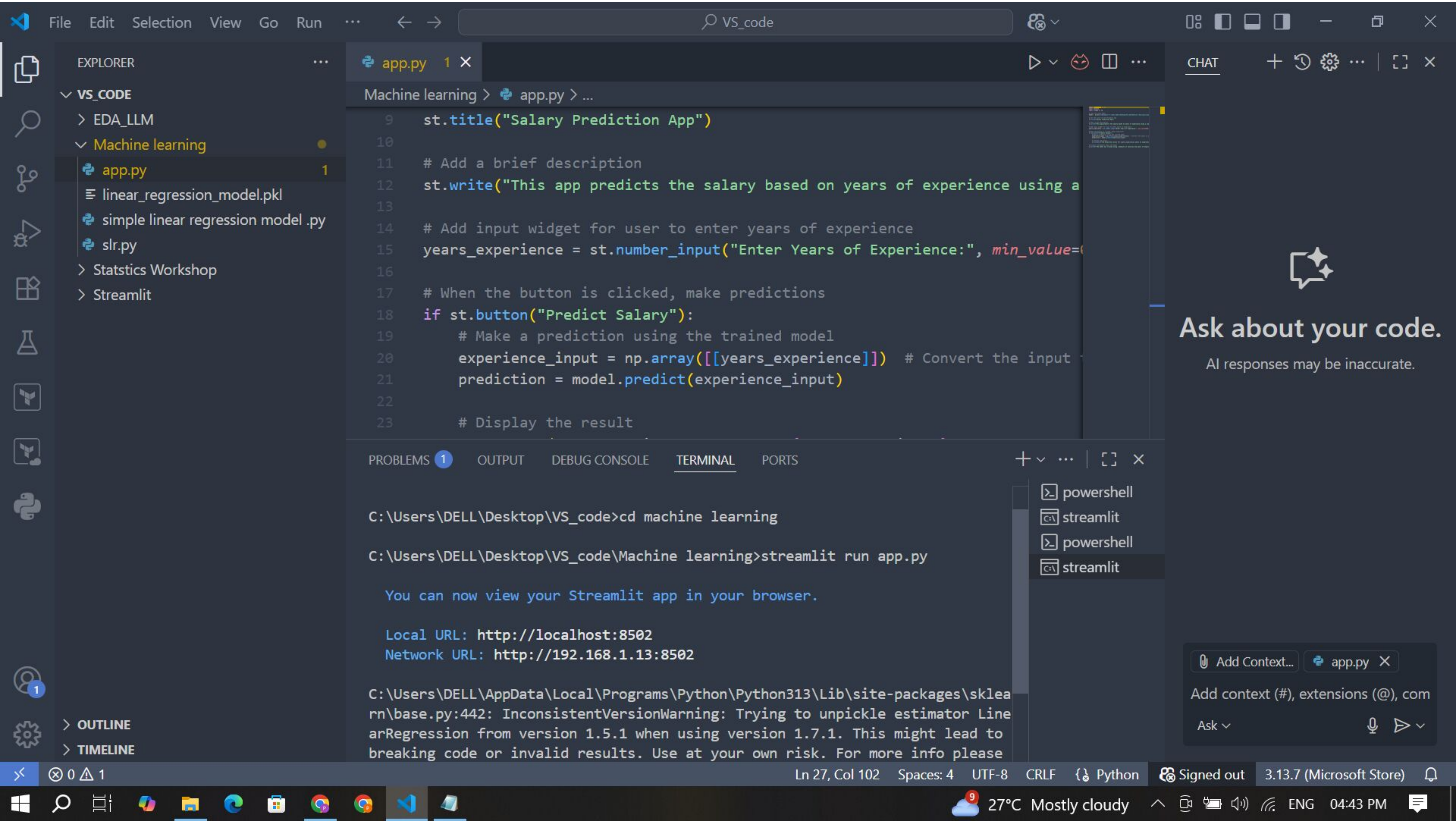


```
st.title("Salary Prediction App")

# Add a brief description
st.write("This app predicts the salary based on years of experience using a

# Add input widget for user to enter years of experience
years_experience = st.number_input("Enter Years of Experience:", min_value=(

# When the button is clicked, make predictions
if st.button("Predict Salary"):
    # Make a prediction using the trained model
    experience_input = np.array([[years_experience]])  # Convert the input
    prediction = model.predict(experience_input)

    # Display the result
```

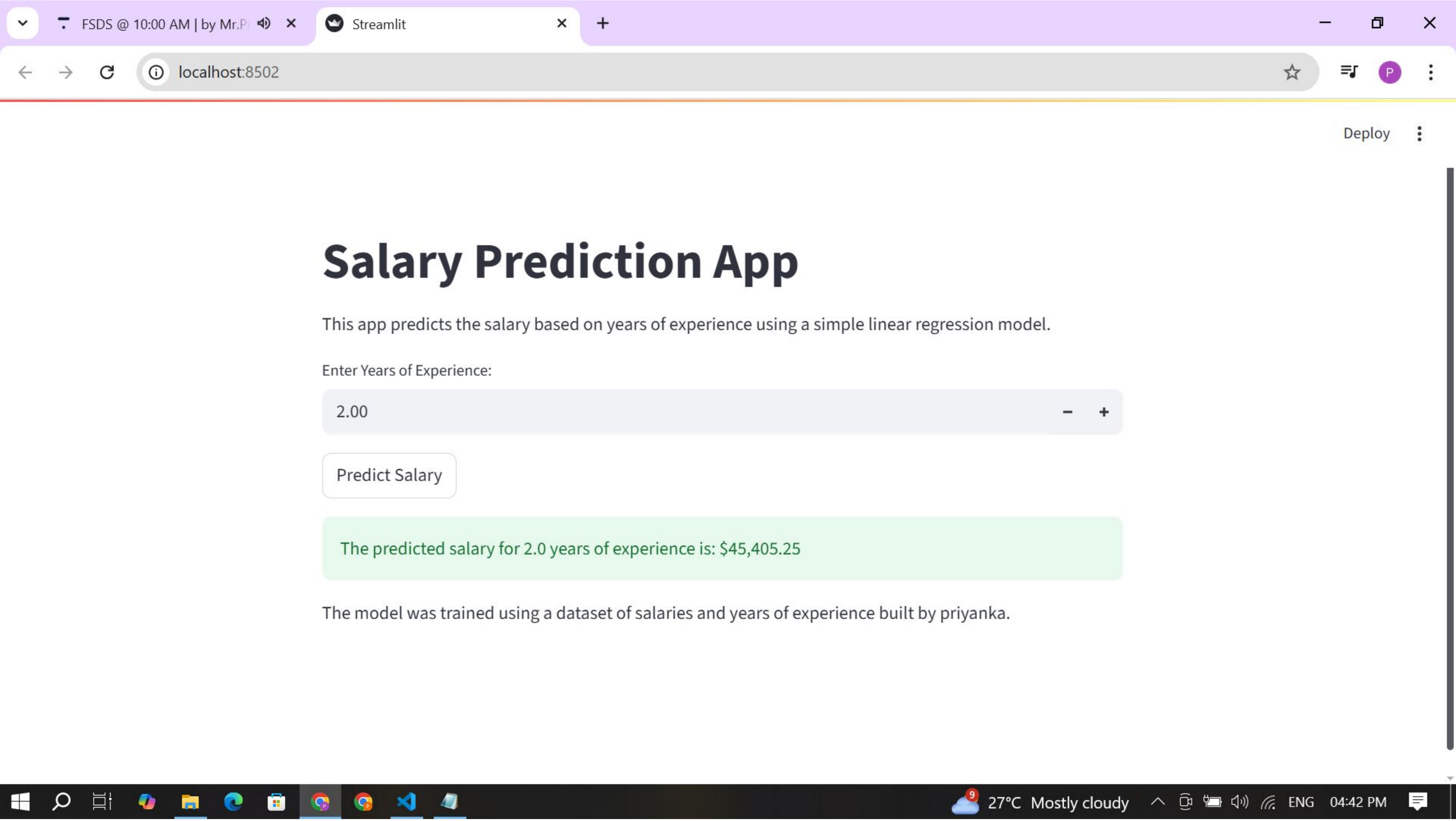
```
C:\Users\DELL\Desktop\VS_code>cd machine learning

C:\Users\DELL\Desktop\VS_code\Machine learning>streamlit run app.py

  You can now view your Streamlit app in your browser.

  Local URL: http://localhost:8502
  Network URL: http://192.168.1.13:8502

C:\Users\DELL\AppData\Local\Programs\Python\Python313\Lib\site-packages\sklea
rn\base.py:442: InconsistentVersionWarning: Trying to unpickle estimator Line
arRegression from version 1.5.1 when using version 1.7.1. This might lead to
breaking code or invalid results. Use at your own risk. For more info please
```



Salary Prediction App

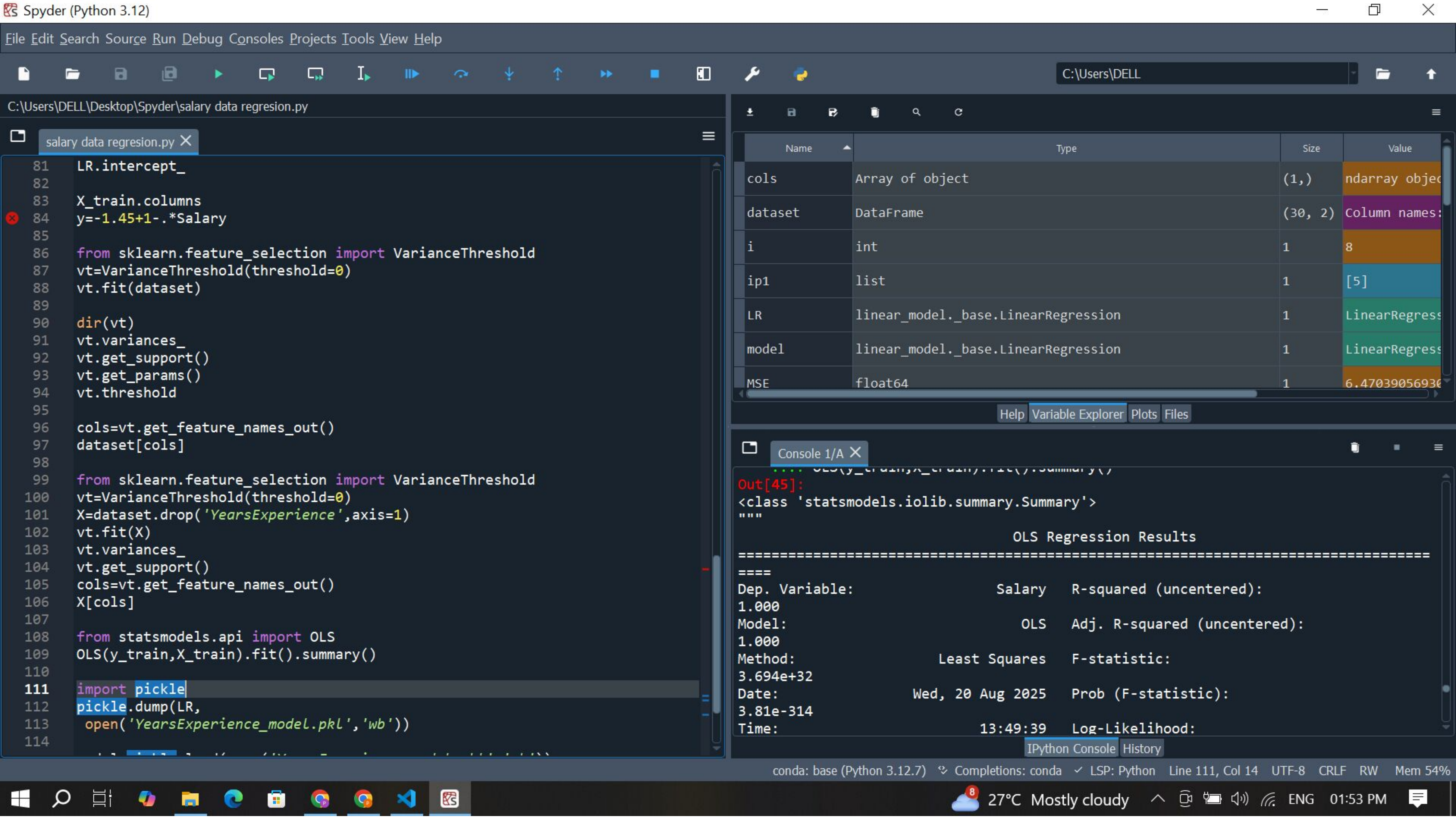
This app predicts the salary based on years of experience using a simple linear regression model.

Enter Years of Experience:

2.00

The predicted salary for 2.0 years of experience is: $45,405.25

The model was trained using a dataset of salaries and years of experience built by priyanka.



Spyder (Python 3.12)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\DELL
C:\Users\DELL\Desktop\Spyder\salary data regresion.py
salary data regresion.py
Help Variable Explorer Plots Files
Console 1/A
IPython Console History
conda: base (Python 3.12.7) Completions: conda LSP: Python Line 111, Col 14 UTF-8 CRLF RW Mem 54%
27°C Mostly cloudy ENG 01:53 PM

```python
LR.intercept_

X_train.columns
y=-1.45+1-.*Salary

from sklearn.feature_selection import VarianceThreshold
vt=VarianceThreshold(threshold=0)
vt.fit(dataset)

dir(vt)
vt.variances_
vt.get_support()
vt.get_params()
vt.threshold

cols=vt.get_feature_names_out()
dataset[cols]

from sklearn.feature_selection import VarianceThreshold
vt=VarianceThreshold(threshold=0)
X=dataset.drop('YearsExperience',axis=1)
vt.fit(X)
vt.variances_
vt.get_support()
cols=vt.get_feature_names_out()
X[cols]

from statsmodels.api import OLS
OLS(y_train,X_train).fit().summary()

import pickle
pickle.dump(LR,
 open('YearsExperience_model.pkl','wb'))

```

Name	Type	Size	Value
cols	Array of object	(1,)	ndarray objec
dataset	DataFrame	(30, 2)	Column names:
i	int	1	8
ip1	list	1	[5]
LR	linear_model._base.LinearRegression	1	LinearRegress
model	linear_model._base.LinearRegression	1	LinearRegress
MSE	float64	1	6.47039056936

```
Out[45]:
<class 'statsmodels.iolib.summary.Summary'>
"""
                                 OLS Regression Results
===================================================================================
====
Dep. Variable:                 Salary   R-squared (uncentered):
1.000
Model:                            OLS   Adj. R-squared (uncentered):
1.000
Method:                 Least Squares   F-statistic:
3.694e+32
Date:                Wed, 20 Aug 2025   Prob (F-statistic):
3.81e-314
Time:                        13:49:39   Log-Likelihood:
```