```python
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
import pickle

# Load the dataset
dataset = pd.read_csv(r'C:\Users\A3MAX SOFTWARE TECH\Desktop\WORK\1. KODI
WORK\1. NARESH\2. EVENING BATCH\N_Batch -- 7.30PM\3. Sep\6th- slr\SIMPLE LINEAR
REGRESSION\Salary_Data.csv')

# Split the data into independent and dependent variables
X = dataset.iloc[:, :-1].values
y = dataset.iloc[:, 1].values

# Split the dataset into training and testing sets (80-20%)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20,
random_state=0)

# Train the model
regressor = LinearRegression()
regressor.fit(X_train, y_train)

# Predict the test set
y_pred = regressor.predict(X_test)

# Visualize the training set
plt.scatter(X_train, y_train, color='red')
plt.plot(X_train, regressor.predict(X_train), color='blue')
plt.title('Salary vs Experience (Training set)')
plt.xlabel('Years of Experience')
plt.ylabel('Salary')
plt.show()

# Visualize the test set
plt.scatter(X_test, y_test, color='red')
plt.plot(X_train, regressor.predict(X_train), color='blue')
plt.title('Salary vs Experience (Test set)')
plt.xlabel('Years of Experience')
plt.ylabel('Salary')
plt.show()

# Predict salary for 12 and 20 years of experience using the trained model
y_12 = regressor.predict([[12]])
y_20 = regressor.predict([[20]])
print(f"Predicted salary for 12 years of experience: ${y_12[0]:,.2f}")
print(f"Predicted salary for 20 years of experience: ${y_20[0]:,.2f}")

# Check model performance
bias = regressor.score(X_train, y_train)
variance = regressor.score(X_test, y_test)
train_mse = mean_squared_error(y_train, regressor.predict(X_train))
test_mse = mean_squared_error(y_test, y_pred)

print(f"Training Score (R^2): {bias:.2f}")
print(f"Testing Score (R^2): {variance:.2f}")
print(f"Training MSE: {train_mse:.2f}")
print(f"Test MSE: {test_mse:.2f}")

# Save the trained model to disk
filename = 'linear_regression_model.pkl'
with open(filename, 'wb') as file:
```

```
    pickle.dump(regressor, file)
print("Model has been pickled and saved as linear_regression_model.pkl")
```