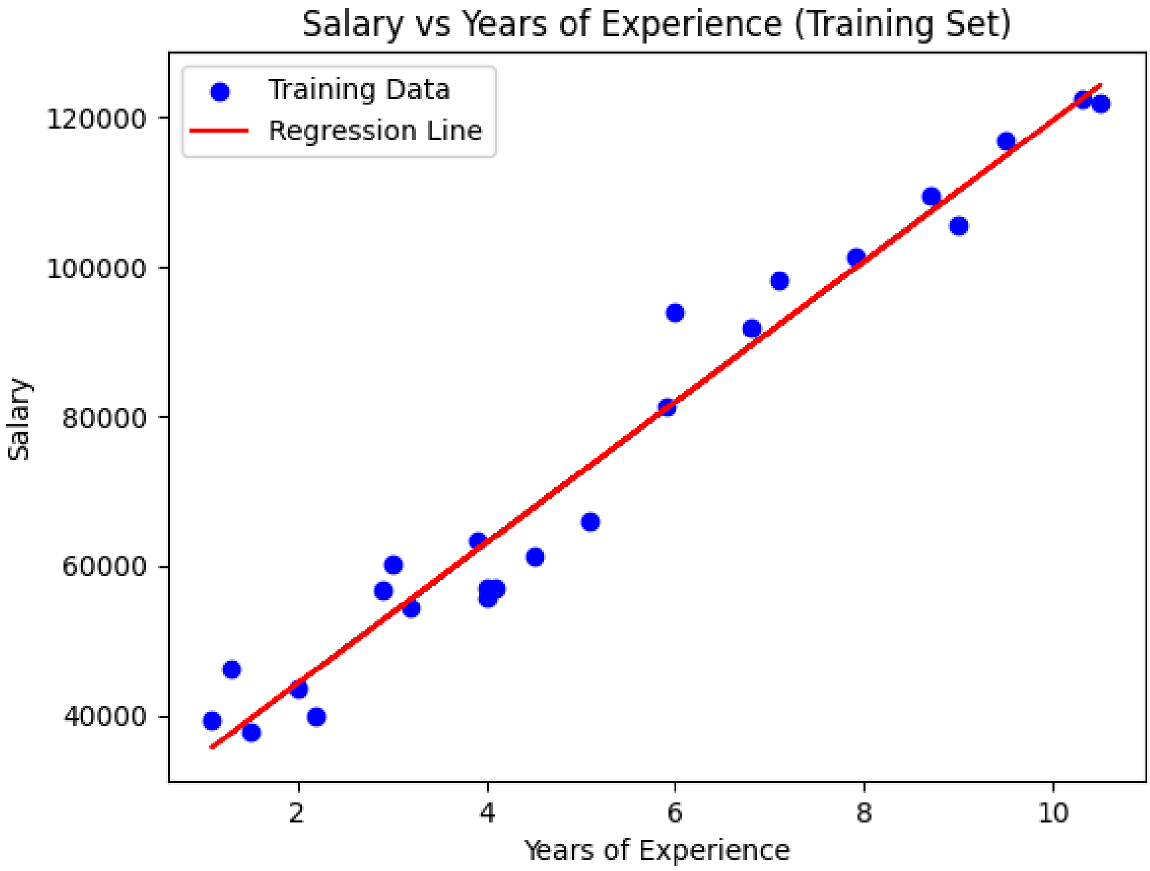
In [3]:

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
import matplotlib.pyplot as plt

# Load the dataset
salary_data = pd.read_csv('Salary_Data.csv')  # Update with the correct p

# Separate the features (YearsExperience) and target variable (Salary)
X = salary_data[['YearsExperience']]  # Features (2D array)
y = salary_data['Salary']  # Target variable

# Split the dataset into training and testing sets (80% training, 20% tes
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,

# Create a linear regression model
model = LinearRegression()

# Train the model using the training data
model.fit(X_train, y_train)

# Make predictions using the testing data
y_pred = model.predict(X_test)

# Evaluate the model
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print(f"Mean Squared Error: {mse}")
print(f"R^2 Score: {r2}")

# Visualize the results
plt.scatter(X_train, y_train, color='blue', label='Training Data')
plt.plot(X_train, model.predict(X_train), color='red', label='Regression
plt.xlabel('Years of Experience')
plt.ylabel('Salary')
plt.title('Salary vs Years of Experience (Training Set)')
plt.legend()
plt.show()

# Take user input for prediction
years_experience = float(input("Enter years of experience: "))

# Convert the user input to a DataFrame with the same column name
new_data = pd.DataFrame({'YearsExperience': [years_experience]})

# Predict the salary for the given input
predicted_salary = model.predict(new_data)
print(f"Predicted Salary for {years_experience} years of experience: {pre
```

```
Mean Squared Error: 49830096.85590839
R^2 Score: 0.9024461774180497
```



```
Predicted Salary for 2.11 years of experience: 45205.83334337217
```

```
In [ ]:
```