5 week sales data : week 1 x1 y1 1.2 1.8 2.6 3.2 3.8 apply lr technique to find 7 th and 9th month

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

X = np.array([[1], [2], [3], [4], [5]])
y = np.array([1.2, 1.8, 2.6, 3.2, 3.8])

from sklearn.linear_model import LinearRegression
model = LinearRegression()
model.fit(X, y)

pred_7 = model.predict([[7]])
pred_9 = model.predict([[9]])

plt.scatter(X, y, color='blue', label='Original data')  # Original points

X_line = np.linspace(1, 10, 100).reshape(-1, 1)
y_line = model.predict(X_line)
plt.plot(X_line, y_line, color='black', label='Regression line')

plt.scatter([7], pred_7, color='green', label=f'Month 7: {pred_7[0]:.2f}', mark
plt.scatter([9], pred_9, color='red', label=f'Month 9: {pred_9[0]:.2f}', marker

plt.xlabel('Month')
plt.ylabel('Sales')
plt.title('Sales Forecast with Linear Regression')
plt.legend()
plt.grid(True)
plt.show()
```
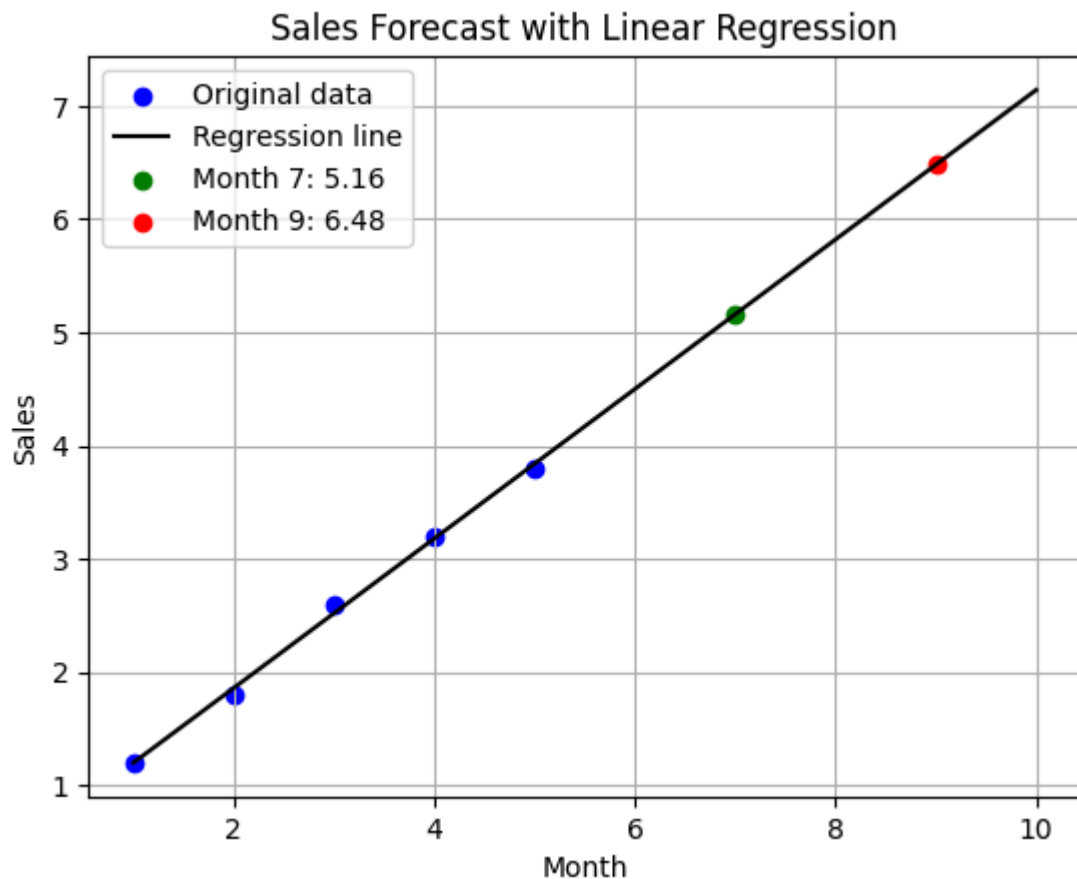
Sales Forecast with Linear Regression

Double-click (or enter) to edit

week 1 : 12 hours (study hours) 18 22 28 35 no of hours will be spent by the dayschlor in the 7 th and 9th week

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

X = np.array([[1], [2], [3], [4], [5]])
y = np.array([12, 18, 22, 28, 35])

model = LinearRegression()
model.fit(X, y)

pred_7 = model.predict([[7]])
pred_9 = model.predict([[9]])

plt.scatter(X, y, color='blue', label='Actual study hours')

X_line = np.linspace(1, 10, 100).reshape(-1, 1)
y_line = model.predict(X_line)
plt.plot(X_line, y_line, color='yellow', label='Regression line')

plt.scatter([7], pred_7, color='red', label=f'Week 7: {pred_7[0]:.2f}', marker=
plt.scatter([9], pred_9, color='red', label=f'Week 9: {pred_9[0]:.2f}', marker=
```

```
plt.xlabel('Week')
plt.ylabel('Study Hours')
plt.title('Study Hours Forecast with Linear Regression')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```