In [1]:

```
import numpy as np
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
import matplotlib.pyplot as plt
```

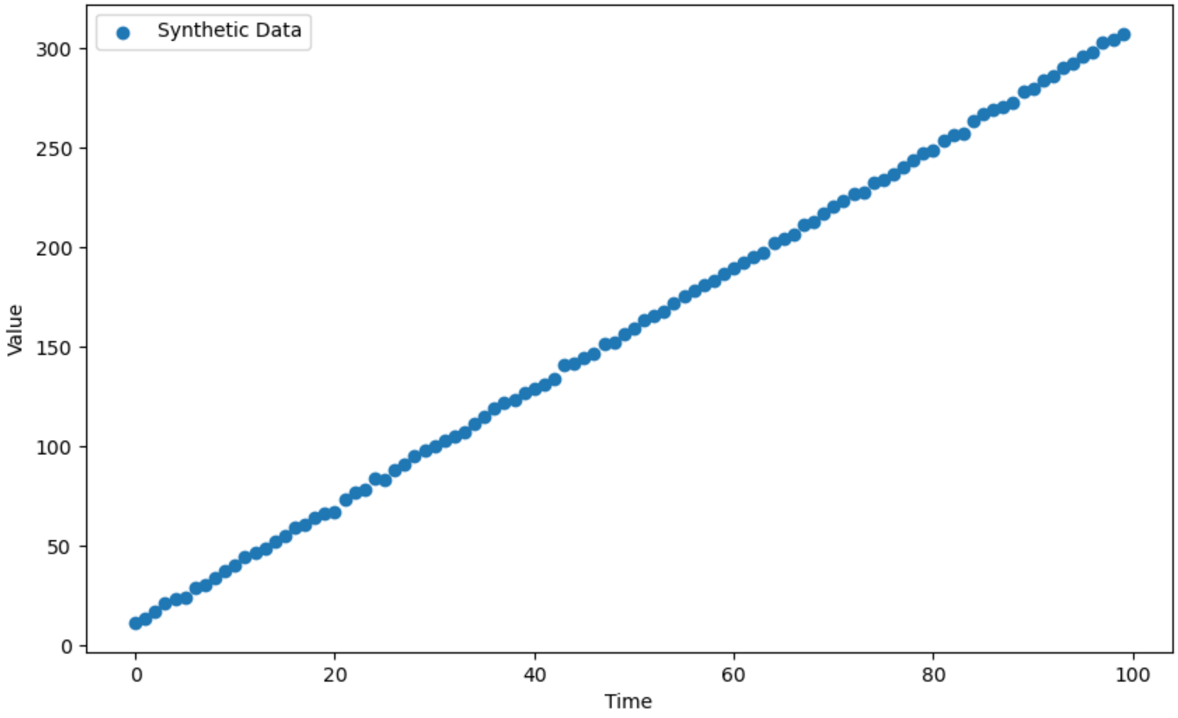
In [2]:

```
np.random.seed(0)
n_samples = 100
X = np.arange(n_samples).reshape(-1, 1)
y = 3 * X + 10 + np.random.randn(n_samples, 1)

print("X (Features):")
print(X[:5])
print("\ny (Labels):")
print(y[:5])
```

```
X (Features):
[[0]
 [1]
 [2]
 [3]
 [4]]

y (Labels):
[[11.76405235]
 [13.40015721]
 [16.97873798]
 [21.2408932 ]
 [23.86755799]]
```

In [3]:

```
import matplotlib.pyplot as plt

plt.figure(figsize=(10, 6))
plt.scatter(X, y, label="Synthetic Data")
plt.xlabel("Time")
plt.ylabel("Value")
plt.legend()
plt.show()
```



In [4]:

```
print("Mean of X:", np.mean(X))
print("Standard Deviation of X", np.std(X))
print("Mean of y:", np.mean(y))
print("Standard Deviation of y", np.std(y))

split_ratio = 0.8
split_index = int(split_ratio * n_samples)
X_train, X_test = X[:split_index], X[split_index:]
y_train, y_test = y[:split_index], y[split_index:]

print(f"Training set size: {len(X_train)} samples")
print(f"Testing set size: {len(X_test)} samples")
```

```
Mean of X: 49.5
Standard Deviation of X 28.86607004772212
Mean of y: 158.5598080155345
Standard Deviation of y 86.5173440412738
Training set size: 80 samples
Testing set size: 20 samples
```

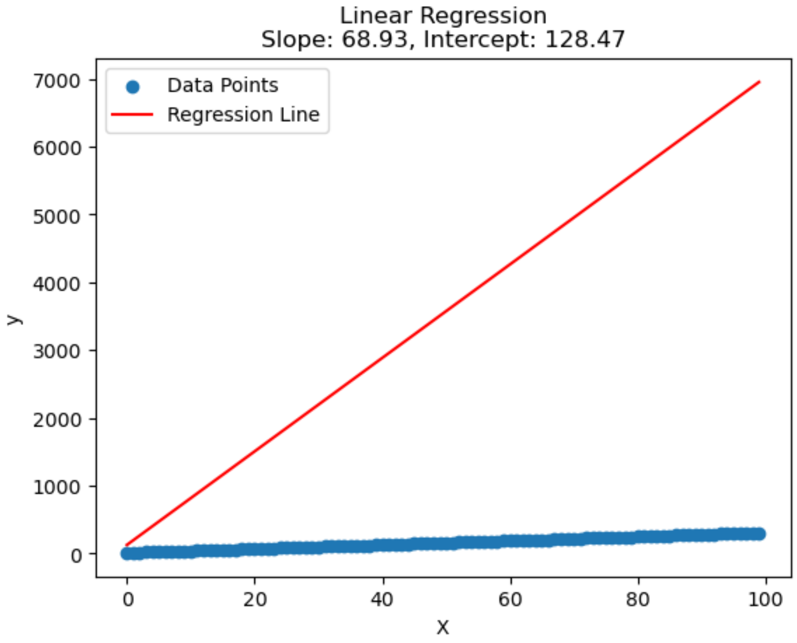
In [5]:

```
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

print(X_test_scaled[:5])
```

```
[[1.75383847]
 [1.79714312]
 [1.84044777]
 [1.88375243]
 [1.92705708]]
```

In [7]:

```
model = LinearRegression()
model.fit(X_train_scaled, y_train)

slope = model.coef_[0][0]
intercept = model.intercept_[0]

plt.scatter(X, y, label="Data Points")

regression_line = slope * X + intercept

plt.plot(X, regression_line, color='red', label="Regression Line")

plt.xlabel("X")
plt.ylabel("y")
plt.legend()
plt.title(f"Linear Regression\nSlope: {slope:.2f}, Intercept: {intercept:.2f}")
plt.show()
```



In [ ]: