

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline

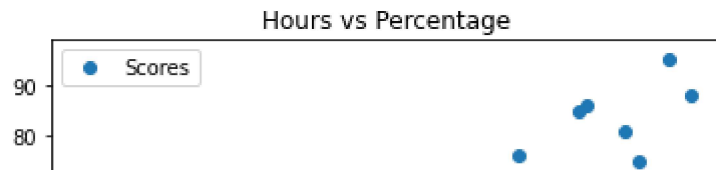
url = "http://bit.ly/w-data"
s_data = pd.read_csv(url)
print("Data imported successfully")

s_data.head(10)
```

☞ Data imported successfully

	Hours	Scores
0	2.5	21
1	5.1	47
2	3.2	27
3	8.5	75
4	3.5	30
5	1.5	20
6	9.2	88
7	5.5	60
8	8.3	81
9	2.7	25

```
s_data.plot(x='Hours', y='Scores', style='o')
plt.title('Hours vs Percentage')
plt.xlabel('Hours Studied')
plt.ylabel('Percentage Score')
plt.show()
```



```
X = s_data.iloc[:, :-1].values
```

```
y = s_data.iloc[:, 1].values
```

```
from sklearn
```

```
from sklearn.model_selection import train_test_split
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y,  
                                                    test_size=0.2, random_state=0)
```

```
from sklearn.linear_model import LinearRegression
```

```
regressor = LinearRegression()
```

```
regressor.fit(X_train, y_train)
```

```
print("Training complete.")
```

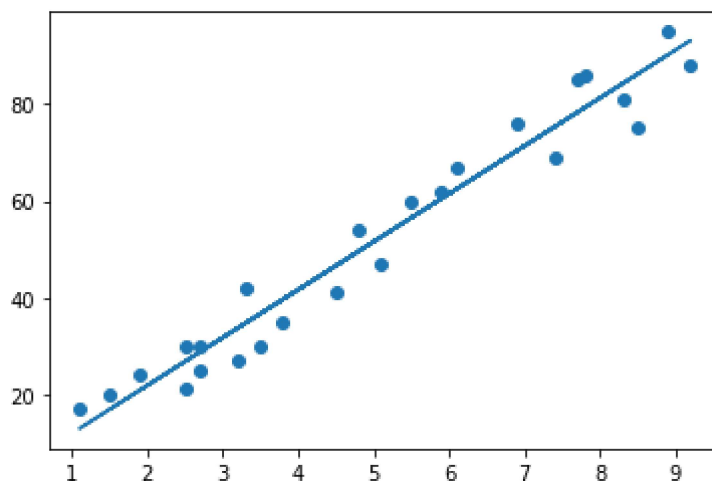
```
Training complete.
```

```
line = regressor.coef_*X+regressor.intercept_
```

```
plt.scatter(X, y)
```

```
plt.plot(X, line);
```

```
plt.show()
```



```
print(X_test) # Testing data - In Hours
```

```
y_pred = regressor.predict(X_test) # Predicting the scores
```


```
[[1.5]
```

```
[3.2]
```

```
[7.4]
```

```
[2.5]  
[5.9]]
```

```
df = pd.DataFrame({'Actual': y_test, 'Predicted': y_pred})  
df
```

	Actual	Predicted	
0	20	16.884145	
1	27	33.732261	
2	69	75.357018	
3	30	26.794801	
4	62	60.491033	

```
from sklearn import metrics  
print('Mean Absolute Error:',  
      metrics.mean_absolute_error(y_test, y_pred))
```

Mean Absolute Error: 4.183859899002982