In [1]:

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

In [2]:

```
df = pd.read_csv("advertising.csv")
```

In [3]:

```
df.head()
```

Out[3]:

	TV	Radio	Newspaper	Sales
0	230.1	37.8	69.2	22.1
1	44.5	39.3	45.1	10.4
2	17.2	45.9	69.3	12.0
3	151.5	41.3	58.5	16.5
4	180.8	10.8	58.4	17.9

In [4]:

```
df.describe()
```

Out[4]:

	TV	Radio	Newspaper	Sales
count	200.000000	200.000000	200.000000	200.000000
mean	147.042500	23.264000	30.554000	15.130500
std	85.854236	14.846809	21.778621	5.283892
min	0.700000	0.000000	0.300000	1.600000
25%	74.375000	9.975000	12.750000	11.000000
50%	149.750000	22.900000	25.750000	16.000000
75%	218.825000	36.525000	45.100000	19.050000
max	296.400000	49.600000	114.000000	27.000000

In [5]:

```
cdf = df[['TV','Radio','Newspaper','Sales']]
cdf.head(9)
```

Out[5]:

	TV	Radio	Newspaper	Sales
0	230.1	37.8	69.2	22.1
1	44.5	39.3	45.1	10.4
2	17.2	45.9	69.3	12.0
3	151.5	41.3	58.5	16.5
4	180.8	10.8	58.4	17.9
5	8.7	48.9	75.0	7.2
6	57.5	32.8	23.5	11.8

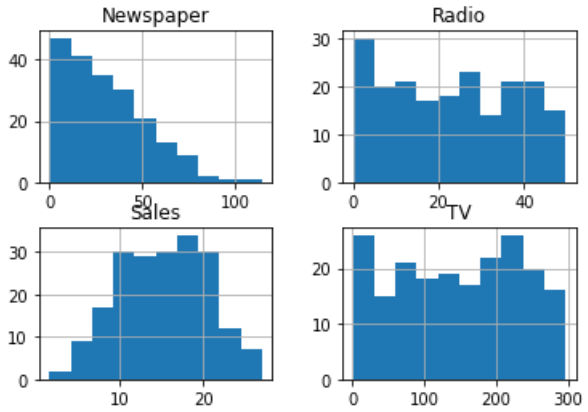
	TV	Radio	Newspaper	Sales
6	57.5	32.8	23.5	11.8
7	120.2	19.6	11.6	13.2
8	8.6	2.1	1.0	4.8

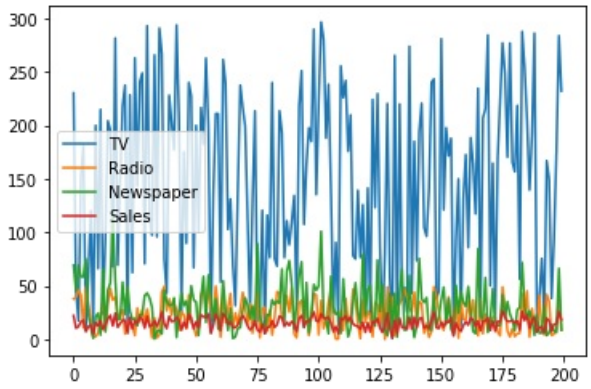
In [9]:

```
viz = cdf[['TV','Radio','Newspaper','Sales']]
viz.hist()
viz.plot()
```

Out[9]:

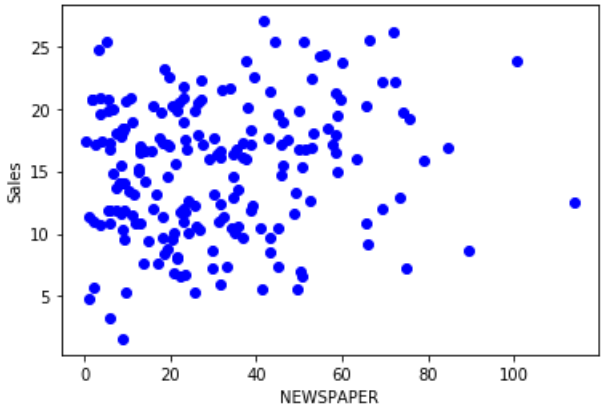
```
<matplotlib.axes._subplots.AxesSubplot at 0x1f29ff615c8>
```





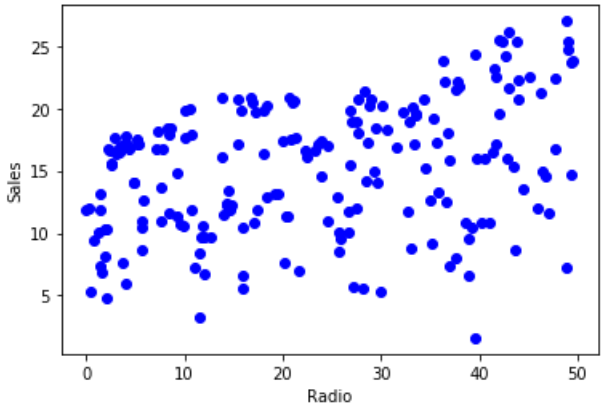
In [11]:

```
plt.scatter(cdf.Newspaper, cdf.Sales,  color='blue')
plt.xlabel("NEWSPAPER")
plt.ylabel("Sales")
plt.show()
```



In [12]:

```
plt.scatter(cdf.Radio, cdf.Sales,  color='blue')
plt.xlabel("Radio")
plt.ylabel("Sales")
plt.show()
```

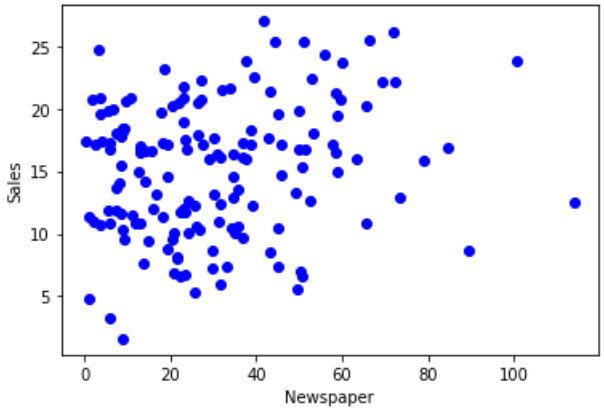


In [14]:

```
#train data distribution
msk = np.random.rand(len(df)) < 0.8
train = cdf[msk]
test = cdf[~msk]
```

In [15]:

```
plt.scatter(train.Newspaper, train.Sales,  color='blue')
plt.xlabel("Newspaper")
plt.ylabel("Sales")
plt.show()
```



In [16]:

```
from sklearn import linear_model
regr = linear_model.LinearRegression()
train_x = np.asanyarray(train[['Newspaper']])
train_y = np.asanyarray(train[['Sales']])
regr.fit (train_x, train_y)
# The coefficients
print ('Coefficients: ', regr.coef_)
print ('Intercept: ',regr.intercept_)
```

```
Coefficients:  [[0.04874876]]
Intercept:  [13.70930176]
```

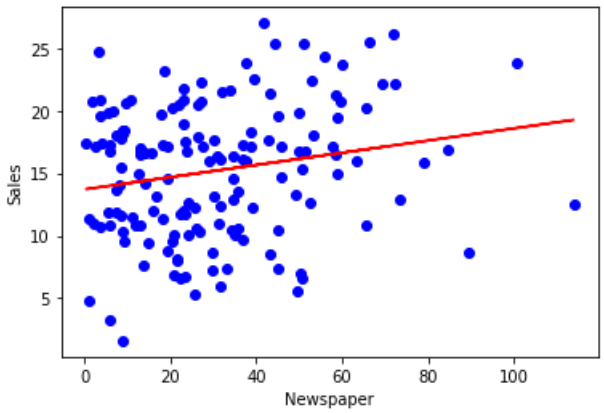
In [18]:

```
#plot the best fit line
plt.scatter(train.Newspaper, train.Sales,  color='blue')
plt.plot(train_x, regr.coef_[0][0]*train_x + regr.intercept_[0], '-r')
```

```
plt.xlabel("Newspaper")
plt.ylabel("Sales")
```

Out[18]:

```
Text(0, 0.5, 'Sales')
```



In [19]:

```
# to compare actual and predicted values and to find mse
from sklearn.metrics import r2_score

test_x = np.asanyarray(test[['Newspaper']])
test_y = np.asanyarray(test[['Sales']])
test_y_ = regr.predict(test_x)

print("Mean absolute error: %.2f" % np.mean(np.absolute(test_y_ - test_y)))
print("Residual sum of squares (MSE): %.2f" % np.mean((test_y_ - test_y) ** 2))
print("R2-score: %.2f" % r2_score(test_y_ , test_y) )
```

```
Mean absolute error: 4.16
Residual sum of squares (MSE): 26.27
R2-score: -22.52
```

In [ ]:

```
```