In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import math
```

In [2]:
```
df = pd.read_csv('Linear Regression Data.csv')
```

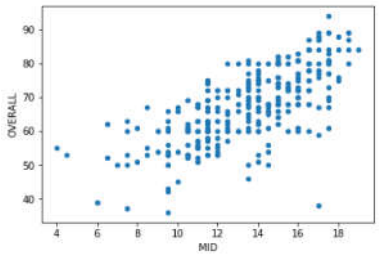
In [3]:
```
df.head()
```

Out[3]:

	MID	FINAL	OVERALL
0	15.0	28.5	66
1	7.0	23.0	50
2	13.5	37.5	73
3	14.0	27.0	62
4	16.0	25.0	60

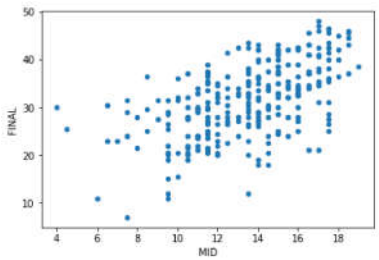
In [13]:
```
df.plot.scatter(x='MID',y='OVERALL')
```

Out[13]:
```
<matplotlib.axes._subplots.AxesSubplot at 0xd24d67e630>
```



In [15]:
```
df.plot.scatter(x='MID',y='FINAL')
```

Out[15]:
```
<matplotlib.axes._subplots.AxesSubplot at 0xd24ddff208>
```



In [4]:
```
x = df.MID
y = df.OVERALL
```

In [6]:
```
n = np.size(x)

# mean of x and y vector
m_x = np.mean(x)
m_y = np.mean(y)

# calculating cross-deviation and deviation about x
SS_xy = np.sum(y*x) - n*m_y*m_x
SS_xx = np.sum(x*x) - n*m_x*m_x
SS_yy = np.sum(y*y) - n*m_y*m_y

# calculating regression coefficients
b = SS_xy / SS_xx
a = m_y - b*m_x
```

In [7]:
```
print("Slope:",b)
print("Intercept:",a)
```

```
Slope: 2.6728276838411533
Intercept: 31.25185229357586
```

In [8]:
```
#computing r and r_squared
r = SS_xy/math.sqrt(SS_xx*SS_yy)
r_squared = b*SS_xy/SS_yy
```

In [9]:
```
print("R:",r)
print("R Squared:",r_squared)
```

```
R: 0.7071705369595216
R Squared: 0.5000901683436181
```

In [21]:
```
#computing Errors
y_pred = np.array([b*i+a for i in x])
MAE = np.sum(np.abs(y_pred-y))/len(x)
MSE = np.sum((y_pred-y)**2)/len(x)
RMSE = np.sqrt(MSE)
```

In [22]:
```
print("Mean Absolute Error: ",MAE)
print("Mean Squared Error: ",MSE)
print("Root Mean Squared Error: ",RMSE)
```

```
Mean Absolute Error:  6.0966939633116874
Mean Squared Error:  60.08993186951793
Root Mean Squared Error:  7.751769596003092
```

In [ ]: