In [1]:

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

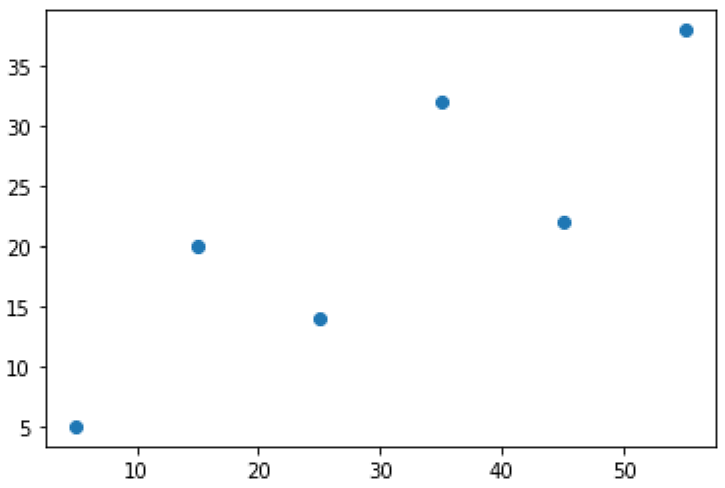
In [2]:

```
x=np.array([5,15,25,35,45,55])
print(x)
x=np.array([5,15,25,35,45,55]).reshape(-1,1)
print(x)
y=np.array([5,20,14,32,22,38])
print(y)
plt.scatter(x,y)
```

```
[ 5 15 25 35 45 55]
[[ 5]
 [15]
 [25]
 [35]
 [45]
 [55]]
[ 5 20 14 32 22 38]
```

Out[2]:

```
<matplotlib.collections.PathCollection at 0x2732feed190>
```



In [4]:

```
model=LinearRegression()
model.fit(x,y)
print(model.intercept_)
print(model.coef_)
print(model.score(x,y))
```

```
5.633333333333329
[0.54]
0.7158756137479542
```

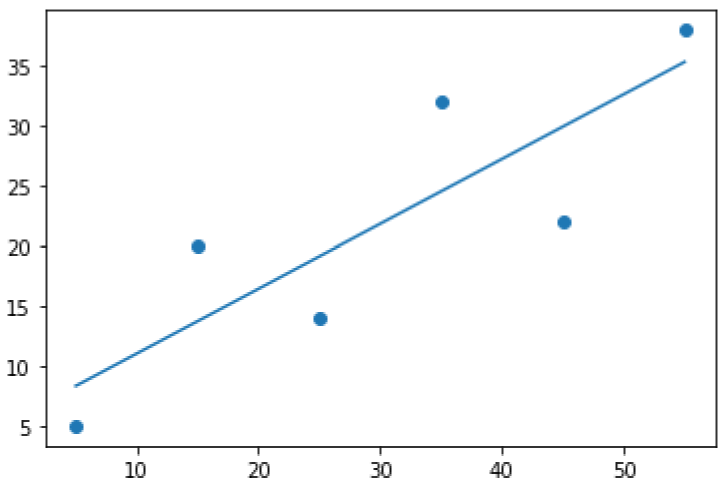
In [8]:

```
y_pred=model.predict(x)
print(y_pred)
print(model.score(x,y_pred))
plt.scatter(x,y)
plt.plot(x,y_pred)
```

```
[ 8.33333333 13.73333333 19.13333333 24.53333333 29.93333333 35.33333333]
1.0
```

Out[8]:

```
[<matplotlib.lines.Line2D at 0x27330330c70>]
```



In [9]:

```
print(model.intercept_+model.coef_*x)
```

```
[[ 8.33333333]
 [13.73333333]
 [19.13333333]
 [24.53333333]
 [29.93333333]
 [35.33333333]]
```

In [4]:

```
import pandas as pd
import numpy as np
from sklearn.linear_model import LinearRegression
x = np.array([[1,2,3,4]])
y = np.array([1,4,9,16])
model = LinearRegression()
model.fit(x,y)
y_predict=model.predict(x)
print(y_predict)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-4-85146f0c860a> in <module>
      5 y = np.array([1,4,9,16])
      6 model = LinearRegression()
----> 7 model.fit(x,y)
      8 y_predict=model.predict(x)
      9 print(y_predict)

~\anaconda\lib\site-packages\sklearn\linear_model\_base.py in fit(self, X, y, sample_weight)
    503
    504         n_jobs_ = self.n_jobs
--> 505         X, y = self._validate_data(X, y, accept_sparse=['csr', 'csc', 'coo'],
    506                                    y_numeric=True, multi_output=True)
    507

~\anaconda\lib\site-packages\sklearn\base.py in _validate_data(self, X, y, reset, validate_separately, **check_params)
    430                 y = check_array(y, **check_y_params)
    431             else:
--> 432                 X, y = check_X_y(X, y, **check_params)
    433             out = X, y
    434

~\anaconda\lib\site-packages\sklearn\utils\validation.py in inner_f(*args, **kwargs)
     70                           FutureWarning)
     71         kwargs.update({k: arg for k, arg in zip(sig.parameters, args)})
---> 72         return f(**kwargs)
     73     return inner_f
     74

~\anaconda\lib\site-packages\sklearn\utils\validation.py in check_X_y(X, y, accept_sparse, accept_large_sparse, dtype, order, copy, force_all_finite, ensure_2d, allow_nd, multi_output, ensure_min_samples, ensure_min_features, y_n
umeric, estimator)
    810         y = y.astype(np.float64)
    811
--> 812     check_consistent_length(X, y)
    813
    814     return X, y

~\anaconda\lib\site-packages\sklearn\utils\validation.py in check_consistent_length(*arrays)
    253     uniques = np.unique(lengths)
    254     if len(uniques) > 1:
--> 255         raise ValueError("Found input variables with inconsistent numbers of"
    256                          " samples: %r" % [int(l) for l in lengths])
    257

ValueError: Found input variables with inconsistent numbers of samples: [1, 4]
```

In [ ]:

In [ ]:

In [ ]:

In [ ]: