In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

In [2]:
```
data=pd.read_csv("aaaa.csv")

data.head()
```

Out[2]:

	no	values
0	1	2
1	2	4
2	3	6
3	4	8
4	5	10

In [3]:
```
real_x= data.iloc[:,0].values
real_y= data.iloc[:,1].values
real_x=real_x.reshape(-1,1)
real_y=real_y.reshape(-1,1)
```

In [4]:
```
training_x,testing_x,training_y,testing_y= train_test_split(real_x,real_y,test_size=0.3,random_state=0)
testing_x
```

Out[4]:
```
array([[ 3],
       [21],
       [24],
       [14],
       [12],
       [26],
       [28],
       [27],
       [ 6]], dtype=int64)
```

In [5]:
```
lin=LinearRegression()
lin.fit(training_x,training_y)
```

Out[5]:
```
LinearRegression()
```

In [6]:
```
pred_y=lin.predict(testing_x)
```

In [7]:
```
# y=mx+b
lin.coef_
```

Out[7]:
```
array([[2.]])
```

In [8]:
```
lin.intercept_
```

Out[8]:
```
array([1.0658141e-14])
```

In [9]:
```
2.* 3+1.0658141e-14
```

Out[9]:
```
6.000000000000011
```

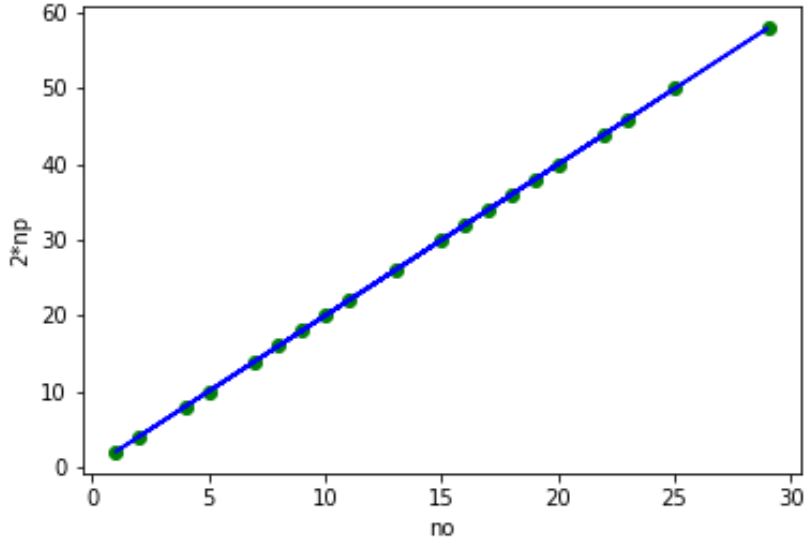
In [10]:
```
testing_y[5]
```

Out[10]:
```
array([52], dtype=int64)
```

In [11]:
```
pred_y[5]
```

Out[11]:
```
array([52.])
```

In [12]:
```
plt.scatter(training_x,training_y,color="g")
plt.plot(training_x,lin.predict(training_x),color="b")
plt.titel=("trining plot")
plt.xlabel("no")
plt.ylabel("2*np")
plt.show()
```



In [14]:
```
plt.scatter(testing_x,testing_y,color="green")
plt.plot(training_x,lin.predict(training_x),color="blue")
plt.titel=("testing plot")
plt.xlabel("no")
plt.ylabel("2*np")
plt.show()
```