Taylor approximation

import library

In [12]:

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

define my function $f(x)$

In [19]:

```
def myfunction(x):

    y = (np.exp(x))/3 + (3*(x**3)) - (2*(x**2))

    return y
```

```
git commit -a -m "define my function"
git push origin master
```

define derivative of my function $f'(x)$

In [20]:

```
def derivative_myfunction(x):

    y_prime = (np.exp(x))/3 + (9*(x**2)) - (4*x)

    return y_prime
```

```
git commit -a -m "define derivative of my function"
git push origin master
```

define 1st order Taylor approxation of my function $\hat{f}(x) = f(a) + f'(a)(x - a)$

In [21]:

```
def taylor(x, a):

    y_approximate = myfunction(a) + (derivative_myfunction(a)*x) - (derivative_myfunction(a)*a)

    return y_approximate
```

```
git commit -a -m "define Taylor approximation"
git push origin master
```

define functions for the visualization

In [34]:

```
x = np.linspace(0,5,10)
y = myfunction(x)

a = 3
b = myfunction(a)

t = taylor(x, a)

def plot_myfunction(x, y):

    plt.plot(x,y, 'b')
    plt.xlim([0,5])
    plt.ylim([0,400])

    plt.show()

def plot_myfunction_and_taylor(x, y, t, a, b):

    plt.plot(x,y, 'b')
    plt.plot(x,t, 'r')
    plt.plot(a,b, 'go')
    plt.xlim([0,5])
    plt.ylim([0,400])

    plt.show()
```

```
git commit -a -m "define functions for the visualization"
git push origin master
```

##

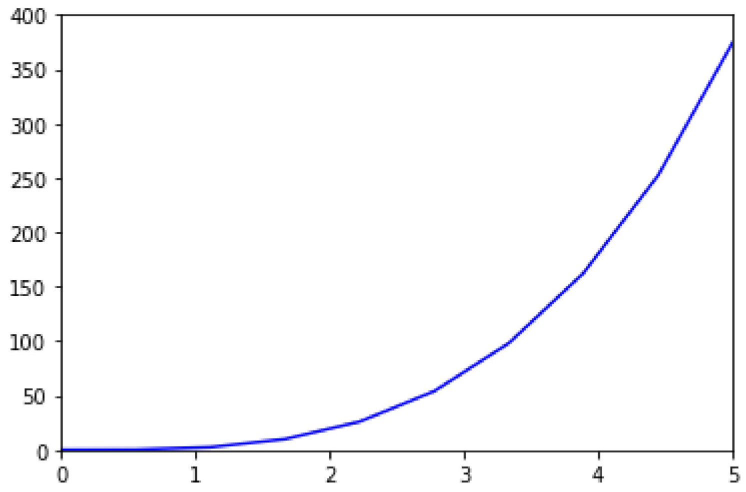
results

##

01. plot my function $f(x)$

In [35]:

```
plot_myfunction(x, y)
```



02. plot my function $f(x)$ & Taylor approxation $\hat{f}(x)$

In [36]:

```
plot_myfunction_and_taylor(x, y, t, a, b)
```