In [1]:

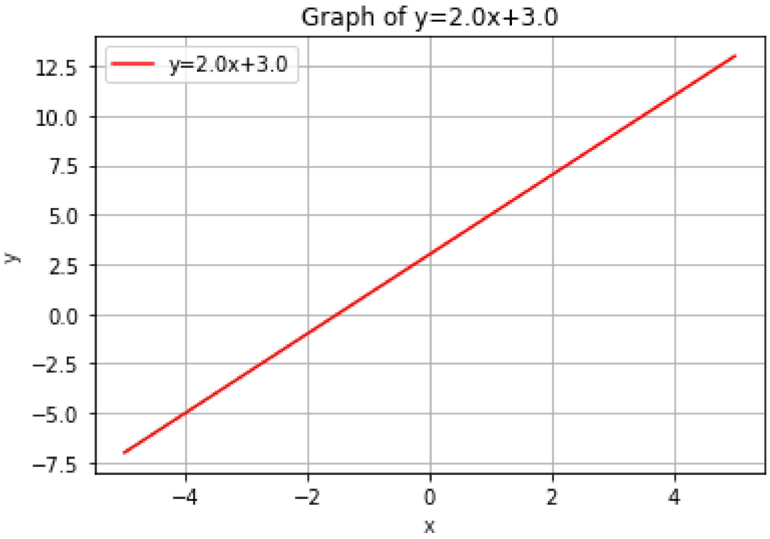
```
x=[0.2,0.4,0.6,0.8]
y=[3.4,3.8,4.2,4.6]
m=1
c=-1
epochs=100 # epochs
l=0.1      # learning rate
ns=len(x)
gama=0.9
vm=0
vc=0
for i in range(epochs):
  for j in range(len(x)):
    gm=((-1.0)*(y[j]-m*x[j]-c)*(x[j]))
    gc=((-1.0)*(y[j]-m*x[j]-c))
    vm=(gama*vm)-(l*gm)
    vc=(gama*vc)-(l*gc)
    m=m+vm
    c=c+vc
print("The local minimum occurs at m =%.2f"%(m),", c = %.2f"%(c))
```

```
The local minimum occurs at m =2.00 , c = 3.00
```

In [2]:

```

import matplotlib.pyplot as plt
import numpy as np
x1=np.linspace(-5,5,100)
y1=m*x1+c
m=round(m,2)
c=round(c,2)
plt.plot(x1,y1,'-r', label='y='+str(m)+'x+'+str(c))
plt.title('Graph of '+'y='+str(m)+'x+'+str(c))
plt.xlabel('x', color='#1C2833')
plt.ylabel('y', color='#1C2833')
plt.legend(loc='upper left')
plt.grid()
plt.show()
```



In [3]:

```
yp=[]
for i in range(len(x)):
  p=(m*x[i])+c
  yp.append(p)
print("Predicted values (yp) : ",yp)
sum=0
for i in range(len(x)):
  sum+=(y[i]-yp[i])**2
  mse=sum/len(x)
print("Mean Square Error (MSE) : ",mse)
```

```
Predicted values (yp) :  [3.4, 3.8, 4.2, 4.6]
Mean Square Error (MSE) :  0.0
```

In [ ]: