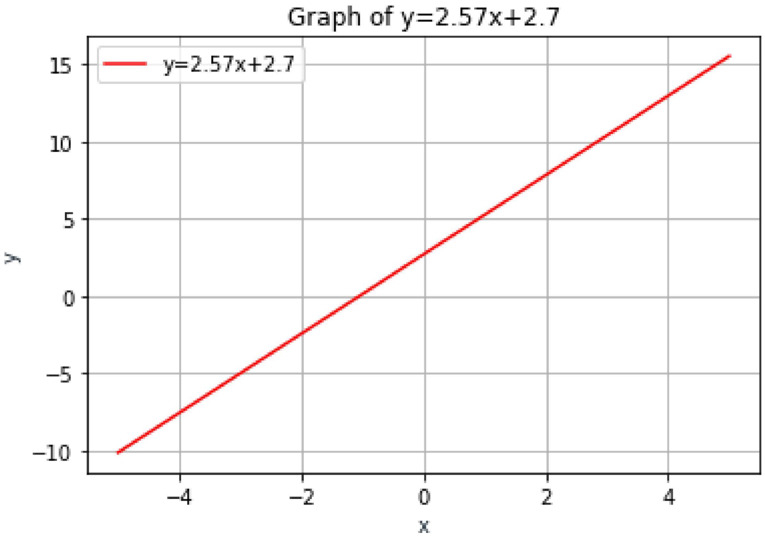
In [1]:

```
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)
for i in range(epochs):
  m_d=0
  c_d=0
  for j in range(len(x)):
    m_d=m_d+((-1.0)*((y[j]-m*x[j]-c)*x[j]))
    c_d=c_d+((-1.0)*(y[j]-m*x[j]-c))
  m_d=m_d/ns
  c_d=c_d/ns
  d_m=-l*m_d
  d_c=-l*c_d
  m=m+d_m
  c=c+d_c
print("The local minimum occurs at m =%.2f"%(m),", c = %.2f"%(c))
```

```
The local minimum occurs at m =2.57 , c = 2.70
```

In [2]:

```
import matplotlib.pyplot as plt
import numpy as np
x1=np.linspace(-5,5,100)
m=round(m,2)
c=round(c,2)
y1=m*x1+c
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)
```

```
Predicted values (yp) :  [3.2140000000000004, 3.728, 4.242, 4.756]
Mean Square Error :  0.016469999999999998
```

In [ ]: