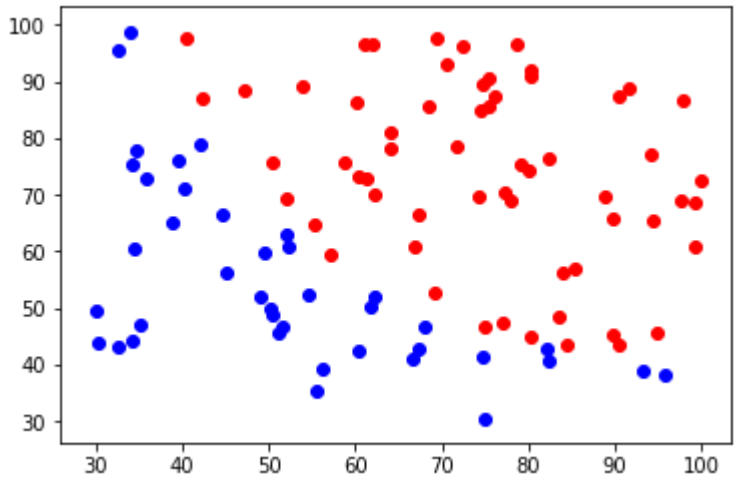
LOKESH SHARMA
SERIAL NO. - 18
BATCH-3

In [96]:
```
import numpy as np
from matplotlib import pyplot as plt
import pandas as pd
data=np.loadtxt("C:/Users/CDAC/Downloads/university_admission.txt",delimiter=','
#print(data)
```

In [97]:
```
x=data[:,0:2]
y=data[:,2]
j=[]
#print(x)
```

In [98]:
```
positives=np.where(y==1)
negatives=np.where(y==0)
#print(positives)
```

In [99]:
```
plt.scatter(x[positives,0],x[positives,1],c='r')
plt.scatter(x[negatives,0],x[negatives,1],c='b')
plt.show()
```



In [100]:
```
def sigmoid(z):
    return 1/(1+np.exp(-z))
def cost(theta,x,y):
    error=0
    prediction=sigmoid(np.dot(x,theta))
    prediction[prediction==1]=0.9
    error=(-y*np.log(prediction))-((1-y)*np.log(1-prediction))
    cost=np.sum(error)/m
    grad=np.dot(x.transpose(),(prediction-y))/m
    return cost,grad
def gradient(X,y,theta,alpha,iters):

    for i in range(1000):
        cst,grad=cost(theta,X,y)
        theta=theta-alpha*cst
        j.append(cst)
    return j

m,n=x.shape[0],x.shape[1]
X=[]
X=np.append(np.ones((m,1)),x,axis=1)
alpha=0.0001
y=y.reshape(m,1)

theta=np.zeros((n+1,1))
iters=np.arange(0,1000,1)
j=gradient(X,y,theta,alpha,iters)

plt.plot(iters,j)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\ipykernel_launcher.py:7: RuntimeWarn
ing: divide by zero encountered in log
  import sys
C:\ProgramData\Anaconda3\lib\site-packages\ipykernel_launcher.py:7: RuntimeWarn
ing: invalid value encountered in multiply
  import sys
```

Out[100]:
```
[<matplotlib.lines.Line2D at 0xab0efd0>]
```