In [2]:

```
import numpy as np
import matplotlib.pyplot as plt
import random
```

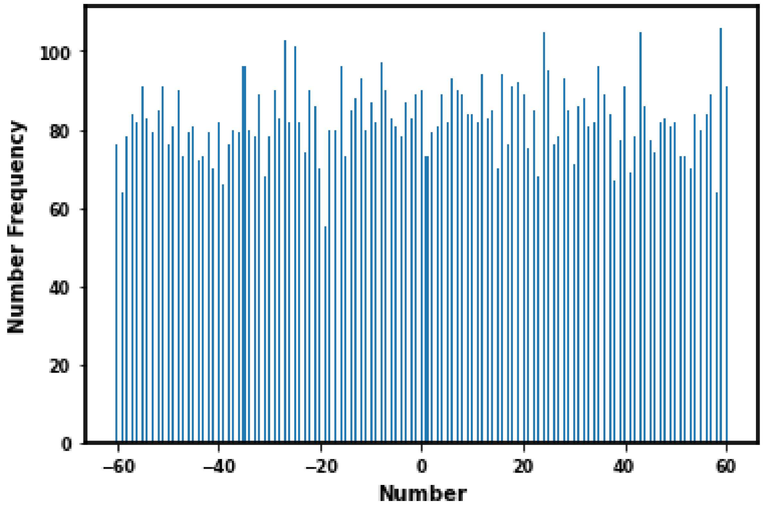
In [55]:

```
n_population=10000  #size of population
A=np.zeros(n_population)  # A define as population
for i in range(n_population):
    A[i]=random.randint(-60,60)   #randomly generated numbers from -60 to 60

A_mean=np.mean(A)
A_median=np.median(A)
A_std=np.std(A)

print("Mean:",A_mean)
print("Median:",A_median)
print("Standard Deviation:",A_std)

x_bar1=np.unique(A)
n1=np.size(x_bar1)
```

```
Mean: 0.404
Median: 1.0
Standard Deviation: 34.64516104739593
```

In [57]:

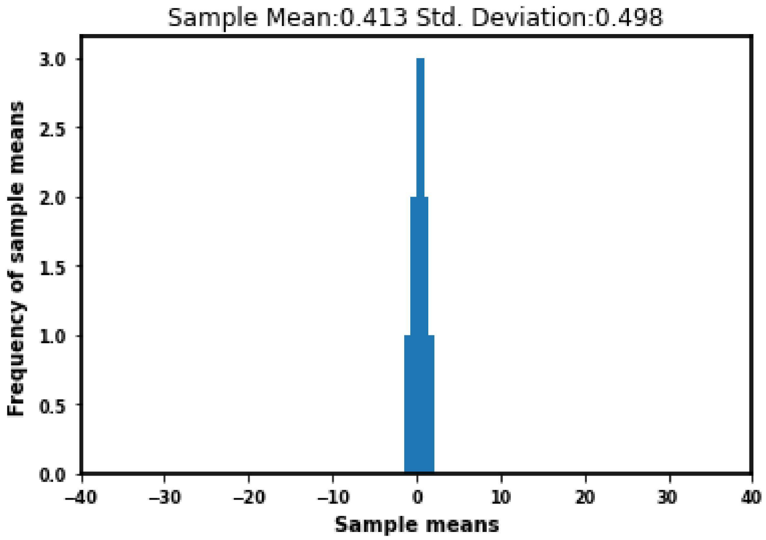
```
A_freq=np.zeros(n1)
for i in range(n1):
    for j in range(n_population):
        if (A[j]==x_bar1[i]):
            A_freq[i]+=1
#print(x_bar1)
#print(A_freq)

plt.bar(x_bar1,A_freq,width=0.4)
#plt.savefig("Population.pdf")
#plt.savefig("Population.png")
plt.xticks(weight='bold',fontsize='8')
plt.yticks(weight='bold',fontsize='8')
plt.xlabel("Number",fontsize='10',fontweight="bold")
plt.ylabel("Number Frequency",fontsize='10',fontweight="bold")
plt.savefig("population.pdf",bbox_inches="tight",pad_inches=0.3,transparent=True)
plt.rcParams["axes.linewidth"]  = 1.8
#plt.legend(prop={'weight':'bold'})
plt.show()
```



In [87]:

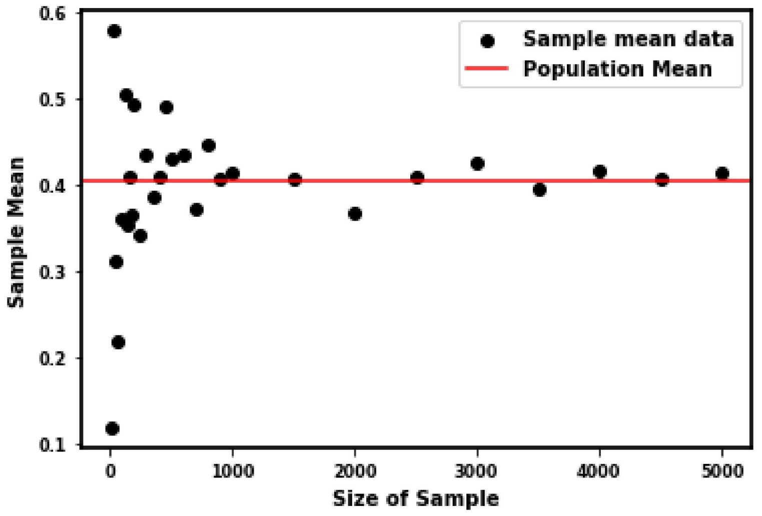
```
n_sample=5000 #sample size  #you have to change sample size to get various distributio
n2=1000  #Number of sampling
A_bar=np.zeros(n2)
for i in range(n2):
    a=[]
    for j in range(n_sample):
        a.append(random.choice(A)) #Sample generated randomly from population(A)
    A_bar[i]=np.mean(a)
#print(A_bar)
x_bar2=np.unique(A_bar)
#print(x_bar2)
n3=np.size(x_bar2)
Freq_A_bar=np.zeros(n3)
for i in range(n3):
    for j in range(n2):
        if A_bar[j]==x_bar2[i]:
            Freq_A_bar[i]+=1
A_bar_mean=round(np.mean(A_bar),3)
A_bar_std=round(np.std(A_bar),3)

print("Sample Mean:",np.mean(A_bar))
print("Sample Std. Deaviation:",np.std(A_bar))
#print(Freq_A_bar)
plt.bar(x_bar2,Freq_A_bar,width=0.4)
#plt.xlabel("Sample means")
#plt.ylabel("Frequency of sample means")
plt.title("Sample Mean:"+str(A_bar_mean)+" Std. Deviation:"+str(A_bar_std))
plt.xlim(-40,40)
#plt.tight_layout()
#plt.savefig("Sample_p10000_S1000.pdf")
plt.xticks(weight='bold',fontsize='8')
plt.yticks(weight='bold',fontsize='8')
plt.xlabel("Sample means",fontsize='10',fontweight="bold")
plt.ylabel("Frequency of sample means",fontsize='10',fontweight="bold")
#plt.savefig("Sample_P5000_S1000.pdf",bbox_inches="tight",pad_inches=0.3,transparent=T
plt.rcParams["axes.linewidth"]  = 1.8
#plt.legend(prop={'weight':'bold'})
plt.show()
```

```
Sample Mean: 0.41335400000000005
Sample Std. Deaviation: 0.49835476781505766
```



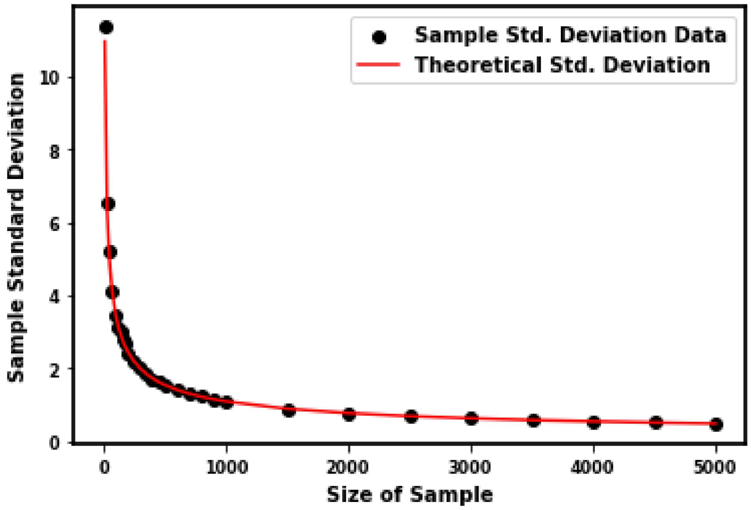
In [93]:

```
#you have make list of means and smaple size, obtain from the above simulation by char
Means=[0.118,0.5784,0.311,0.217,0.361,0.505,0.354,0.41,0.364,0.493,0.341,0.434,0.386,0
n=[10,30,50,70,100,120,140,160,180,200,250,300,350,400,450,500,600,700,800,900,1000,15
plt.scatter(n,Means,color='k',label="Sample mean data")
plt.axhline(0.404,color='red',label="Population Mean")
plt.xticks(weight='bold',fontsize='8')
plt.yticks(weight='bold',fontsize='8')
plt.xlabel("Size of Sample",fontsize='10',fontweight="bold")
plt.ylabel("Sample Mean",fontsize='10',fontweight="bold")
plt.rcParams["axes.linewidth"]  = 1.8
plt.legend(prop={'weight':'bold'})
plt.savefig("MeanVsSize.pdf",bbox_inches="tight",pad_inches=0.3,transparent=True)
plt.show()
```



In [95]:

```
#you have make list of standard deviation and smaple size, obtain from the above simul
std_sample=[11.356,6.518,5.222,4.099,3.443,3.106,2.988,2.78,2.693,2.401,2.214,2.035,1.
n=[10,30,50,70,100,120,140,160,180,200,250,300,350,400,450,500,600,700,800,900,1000,15
ns=np.size(n)
std_theory=np.zeros(ns)
for i in range(ns):
    std_theory[i]=A_std/np.sqrt(n[i])

plt.scatter(n,std_sample,color='k',label='Sample Std. Deviation Data')
plt.plot(n,std_theory,color='red',label='Theoretical Std. Deviation')
plt.xticks(weight='bold',fontsize='8')
plt.yticks(weight='bold',fontsize='8')
plt.xlabel("Size of Sample",fontsize='10',fontweight="bold")
plt.ylabel("Sample Standard Deviation",fontsize='10',fontweight="bold")
plt.rcParams["axes.linewidth"]  = 1.8
plt.legend(prop={'weight':'bold'})
plt.savefig("StdVsSize.pdf",bbox_inches="tight",pad_inches=0.3,transparent=True)
plt.show()
```



In [ ]: