In [4]:

```
#Z-score for given area

import matplotlib.pyplot as plt
import numpy as np
import scipy.stats as sp

area = 0.75     #area under the graph
#convert values to z-score
print('z-score:',sp.norm.ppf(area))
```

```
z-score: 0.6744897501960817
```

In [6]:

```
#Given X ~ N(mu, var)
#Finding Probabilities

import matplotlib.pyplot as plt
import numpy as np
import scipy.stats as sp

mu = 40
x = 47
sigma = 8
z = (x - mu)/sigma
print('Finding P(z <',z,')')
print('Area to left of x : ',sp.norm.cdf(z))

print('Finding P(z >', z,')')
print('Area to right of x : ',1 - sp.norm.cdf(z))
#since total area under the graph is 1
```

```
Finding P(z < 0.875 )
Area to left of x :  0.8092130471474893
Finding P(z > 0.875 )
Area to right of x :  0.19078695285251068
```

In [7]:

```
#Area in an interval between two values a and b
import scipy.stats as sp
def prob(a,b):
    z1 = (a - mu)/sigma
    z2 = (b - mu)/sigma
    print('Finding P(',z1,'< z <', z2,')')
    print('Area to left of a : ',sp.norm.cdf(z1))
    print('Area to left of b : ',sp.norm.cdf(z2))
    print('Area to between a and b: ',sp.norm.cdf(z2) - sp.norm.cdf(z1))

mu = 50
sigma = 5
prob(44.5, 55.5)
```

```
Finding P( -1.1 < z < 1.1 )
Area to left of a :  0.13566606094638267
Area to left of b :  0.8643339390536173
Area to between a and b:  0.7286678781072347
```

In [8]:

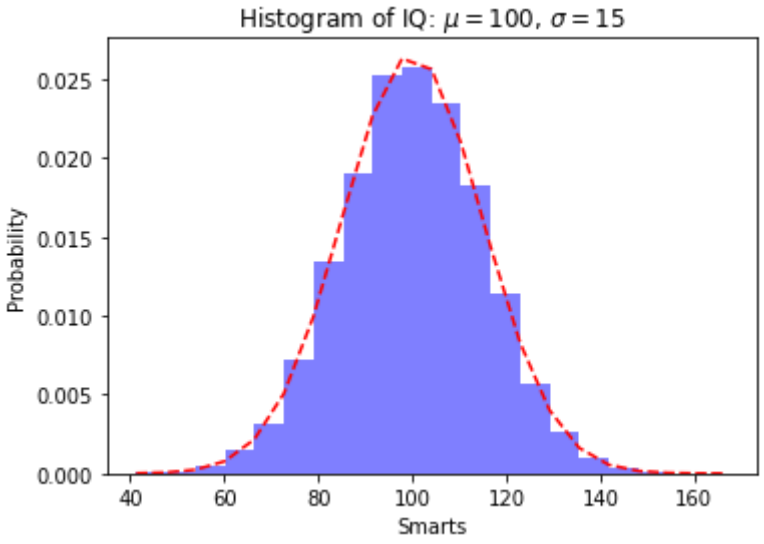
```
import scipy.stats as sp
def fun(z1, z2):
    return sp.norm.cdf(z1) - sp.norm.cdf(z2)

a = 65
b = 32
mu=25
```

```
sigma=5
z1 = (a-mu)/sigma
z2 = (b-mu)/sigma
area = fun(z1, z2)
print("required area is :",area)
```

```
required area is : 0.0807566592337704
```

In [9]:

```
#!/usr/bin/env python

import numpy as np
import matplotlib.mlab as mlab
import matplotlib.pyplot as plt
import scipy.stats as sp

# example data
mu = 100 # mean of distribution
sigma = 15 # standard deviation of distribution
x = mu + sigma * np.random.randn(10000)

num_bins = 20
# the histogram of the data
n, bins, patches = plt.hist(x, num_bins, facecolor='blue', density=True, alpha=0.5)

# add a 'best fit' line
y = sp.norm.pdf(bins, mu, sigma)
plt.plot(bins, y, 'r--')
plt.xlabel('Smarts')
plt.ylabel('Probability')
plt.title(r'Histogram of IQ: $\mu=100$, $\sigma=15$')

# Tweak spacing to prevent clipping of ylabel
plt.subplots_adjust(left=0.15)
plt.show()
```



In [ ]: