In [17]:

```
import scipy.stats as stats
```

In [3]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

file_name = r"/Users/      /Documents/Work/Narela/AC-1-voterlist.xlsx"

sheet = 'VoterList'
df = pd.read_excel(io=file_name, sheet_name=sheet, sep='\s*,\s*')
```

In [4]:

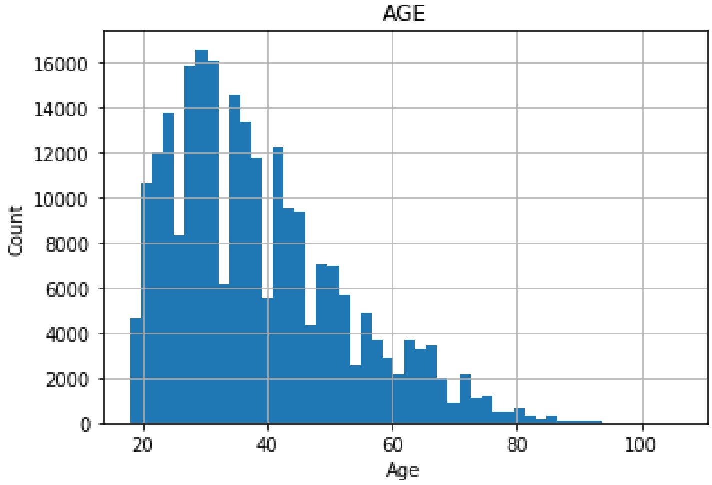
```
df.head()
```

Out[4]:

	ACNo	PartNo	SlNo	EName	Sex	RName	RType	AGE	IDCardNo	STATUSTYPE	VHo
0	1	1	1	BIMLA DEVI	F	JAGDISH CHANDER	H	64	XVX0000026	N	1
1	1	1	2	JAGDISH CHAND BHARA	M	MITTHAN LAL	F	61	XVX0000018	N	1
2	1	1	3	SANDEEP	M	JAGDISH CHAND	F	37	XVX0000034	N	1
3	1	1	4	SARITA DEVI	F	SANDEEP KUMAR	H	34	XVX1521111	N	1
4	1	1	5	RAJA MISHRA	M	NISHIKANT MISHRA	F	21	XVX2556710	N	1

In [10]:

```
df.hist(column='AGE', bins = 50)
plt.xlabel('Age')
plt.ylabel('Count')
plt.show()
```



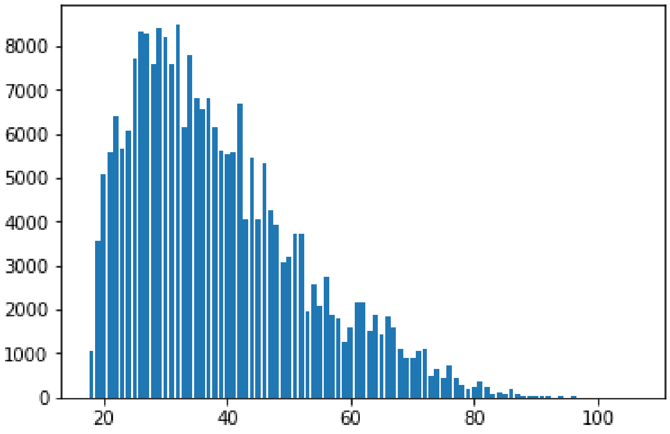
In [78]:

```
df.AGE.value_counts().sort_index().head()
```

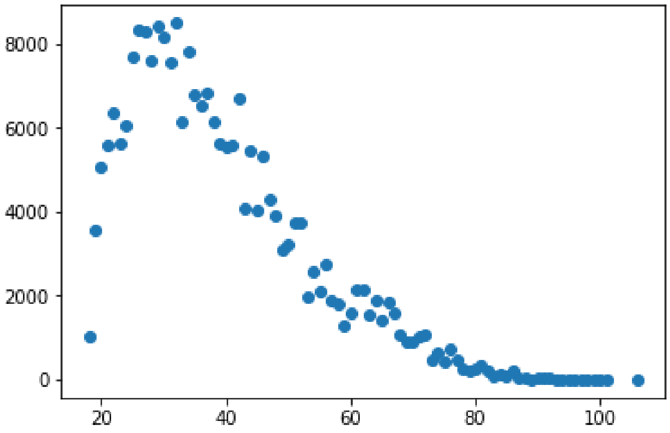
Out[78]:

```
18    1044
19    3584
20    5088
21    5595
22    6382
Name: AGE, dtype: int64
```

In [15]:
```
plt.bar(df.AGE.value_counts().index, df.AGE.value_counts().values)
plt.show()
```



In [52]:
```
plt.scatter(df.AGE.value_counts().sort_index().index, df.AGE.value_counts().sor
t_index().values)
plt.show()
```

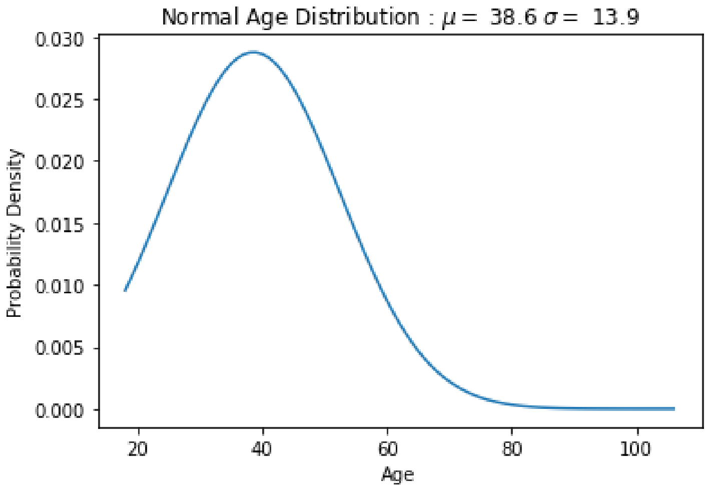


In [25]:
```
sigma = df.AGE.std()
mean = df.AGE.mean()
print("sigma " + str(sigma))
print("mean " + str(mean))
```

```
sigma 13.864475891832324
mean 38.5884051357574
```

In [71]:
```
y = stats.norm.pdf(df.AGE.value_counts().sort_index().index,mean,sigma)
```

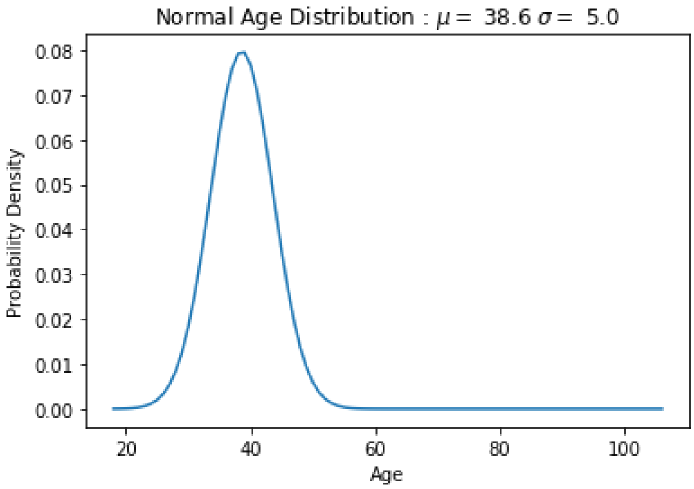
In [72]:
```
plt.plot(df.AGE.value_counts().sort_index().index, y)
plt.title("Normal Age Distribution : $\mu = $ %.1f $\sigma = $ %.1f" %(mean,sig
ma))
plt.xlabel("Age")
plt.ylabel("Probability Density")
plt.show()
```



POSITIVE SKEW

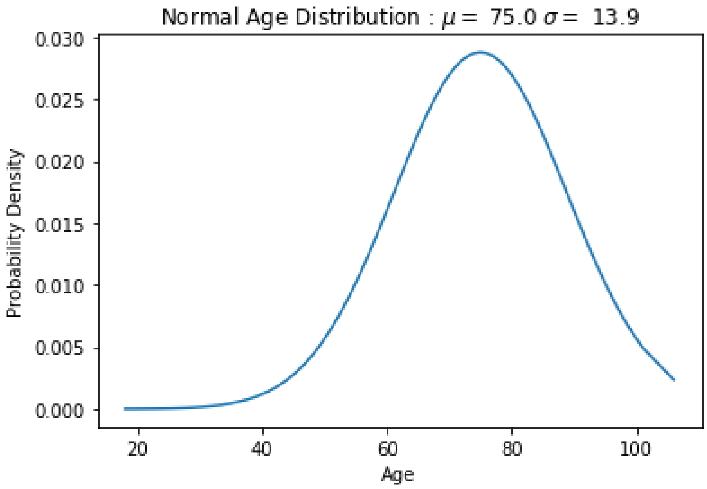
In [60]:
```
y = stats.norm.pdf(df.AGE.value_counts().sort_index().index,mean,5)
```

In [61]:
```
plt.plot(df.AGE.value_counts().sort_index().index, y)
plt.title("Normal Age Distribution : $\mu = $ %.1f $\sigma = $ %.1f" %(mean,5))
plt.xlabel("Age")
plt.ylabel("Probability Density")
plt.show()
```



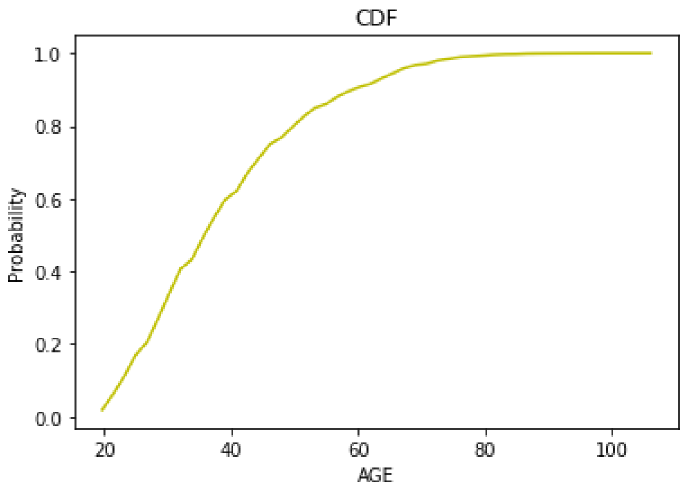
In [66]:
```
y = stats.norm.pdf(df.AGE.value_counts().sort_index().index,75,sigma)
```

```
In [67]: plt.plot(df.AGE.value_counts().sort_index().index, y)
         plt.title("Normal Age Distribution : $\mu = $ %.1f $\sigma = $ %.1f" %(75,sigma
         ))
         plt.xlabel("Age")
         plt.ylabel("Probability Density")
         plt.show()
```



NEGATIVE SKEW

```
In [74]: num_bins = 20
         counts, bin_edges = np.histogram (df.AGE, bins=50, normed=True)
         cdf = np.cumsum (counts)
         plt.plot (bin_edges[1:], cdf/cdf[-1], color ='y')
         plt.xlabel('AGE')
         plt.ylabel('Probability')
         plt.title('CDF')
         plt.show()
```



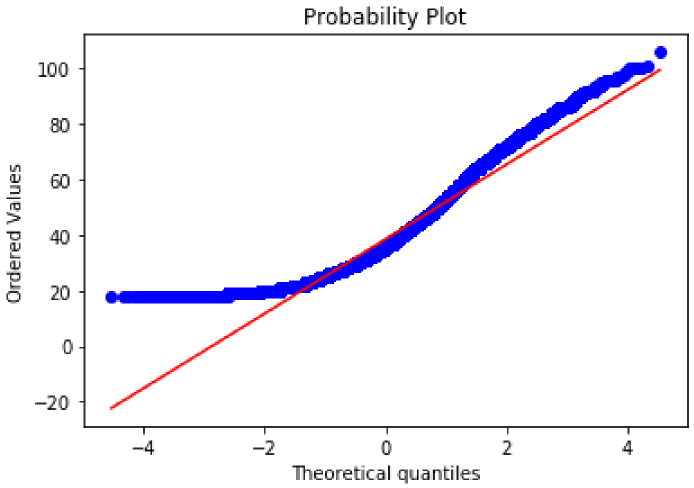
CDF

```
In [75]: df.AGE.mode()

Out[75]: 0    32
         dtype: int64
```

```
In [76]: df.AGE.median()

Out[76]: 36.0
```

TASK: What is a Q-Q Plot?

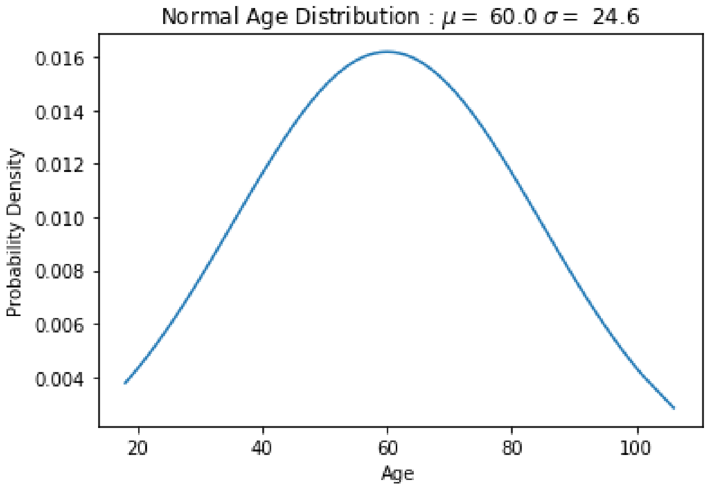
In [85]:
```
stats.probplot(df.AGE, dist="norm", plot=plt)
plt.show()
```



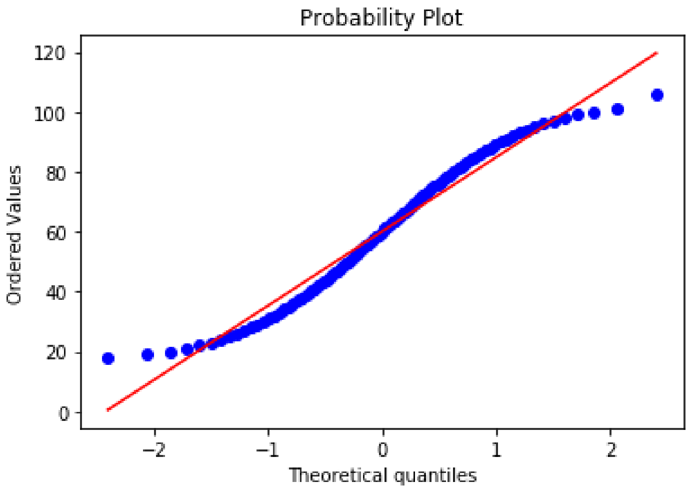
To check the normality of the given data.

Say we have a normal distribution, then the corresponding QQ Plot is:

In [90]:
```
mean = df.AGE.value_counts().index.values.mean()
sigma = df.AGE.value_counts().index.values.std()
y = stats.norm.pdf(df.AGE.value_counts().sort_index().index,mean,sigma)
plt.plot(df.AGE.value_counts().sort_index().index, y)
plt.title("Normal Age Distribution : $\mu = $ %.1f $\sigma = $ %.1f" %(mean,sig
ma))
plt.xlabel("Age")
plt.ylabel("Probability Density")
plt.show()
```



```
In [91]: stats.probplot(df.AGE.value_counts().index, dist="norm", plot=plt)
         plt.show()
```


Probability Plot
Ordered Values
Theoretical quantiles