DAY1

In [27]:

```
import scipy.stats as stats
import numpy as np
```

Q1 A leb bulb manufacturing company regulary conduct qantity check at specified periods on its products. Historically, the failure rate for the leb Bulb is 0.05.
Suppose a random 10 LEB bulb is seleced. Find the probability:

1. None of the Leb bulb are defective
2. Excatly one of the LED bulb is Defective
3. Two or fewer LED Bulbs are DEfective
4. Three or more of the LED bulb are Defecive

In [3]:

```
#1. None of the Leb bulb are defective
# K ~ r in Binomial Formal | the value for which Prob is found.
k = 0
p = 0.05
n = 10
```

In [4]:

```
stats.binom.pmf(k,n,p)
```

Out[4]:

```
0.5987369392383789
```

In [5]:

```
round(stats.binom.pmf(k,n,p)*100 , 3)
```

Out[5]:

```
59.874
```

In [133]:

```
#2. Excatly one of the LED bulb is Defective
k = 1
p = 0.05
n = 10
round(stats.binom.pmf(k,n,p)*100 , 3)
```

Out[133]:

```
31.512
```

In [16]:

```
#3 Two or fewer LED Bulbs are Defective
k = [0,1,2]

stats.binom.pmf(k,n,p)*100

print((stats.binom.pmf(k,n,p)*100).sum())
```

```
98.84964426207036
```

In [134]:

```
k = 2

stats.binom.cdf(k,n,p)*100

print((stats.binom.cdf(k,n,p)*100)) # CDF is meant for comulative
```

```
98.84964426207031
```

In [26]:

```
#4. Three or more of the LED bulb are Defecive
k = 2
two = stats.binom.cdf(k,n,p)
1 - two
```

Out[26]:

```
0.01150355737929687
```

In [28]:

```
#Probability Distribution of Above question
n = 10
k = np.arange(11)
p = 0.05
```

In [32]:

```
pmf = stats.binom.pmf(k,n,p)
pmf
```

Out[32]:

```
array([5.98736939e-01, 3.15124705e-01, 7.46347985e-02, 1.04750594e-02,
       9.64808106e-04, 6.09352488e-05, 2.67259863e-06, 8.03789063e-08,
       1.58642578e-09, 1.85546875e-11, 9.76562500e-14])
```

In [33]:

```
cdf = stats.binom.cdf(k,n,p)
cdf
```

Out[33]:

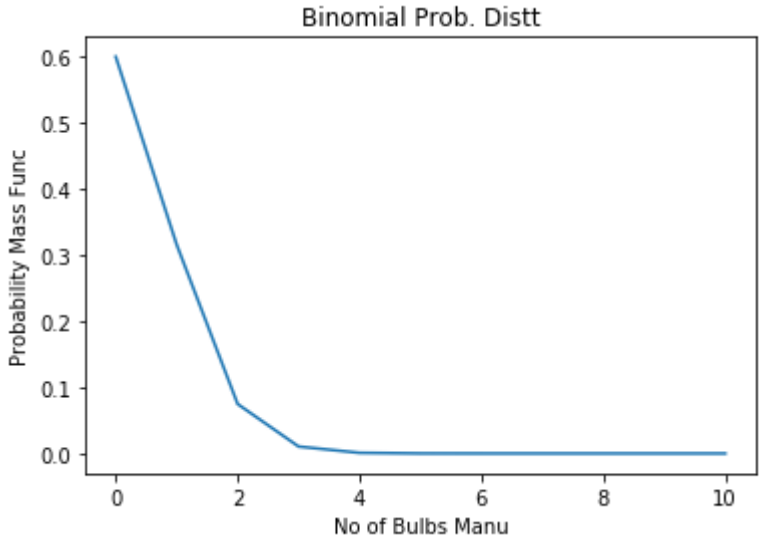
```
array([0.59873694, 0.91386164, 0.98849644, 0.9989715 , 0.99993631,
       0.99999725, 0.99999992, 1.        , 1.        , 1.        ,
       1.        ])
```

In [35]:

```
import matplotlib.pyplot as plt
```

In [41]:

```
plt.plot(k,pmf )
plt.xlabel("No of Bulbs Manu")
plt.ylabel("Probability Mass Func")
plt.title("Binomial Prob. Distt")
plt.show()
```

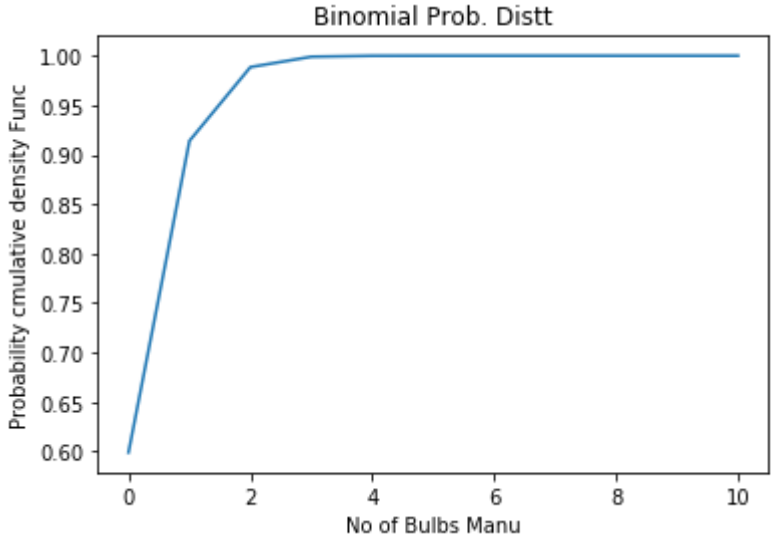


In [42]:

```
#as sample inc probability desc
```

In [45]:

```
plt.plot(k,cdf )
plt.xlabel("No of Bulbs Manu")
plt.ylabel("Probability cmulative density Func")
plt.title("Binomial Prob. Distt")
plt.show()
```



In [146]:

```
p = 0.868
n = 3
k = [0,1,2,3]
```

In [147]:

```
none = stats.binom.pmf(0,n,p)
none
```

Out[147]:

```
0.002299968
```

In [148]:

```
three = stats.binom.pmf(3,n,p)
three
```

Out[148]:

```
0.653972032
```

In [149]:

```
1-stats.binom.pmf(1,n,p) #At least 2 of 3 orders are filled
```

Out[149]:

```
0.954627904
```

In [151]:

```
stats.binom.pmf(2,n,p)+stats.binom.pmf(3,n,p)
```

Out[151]:

```
0.9523279359999999
```

In [75]:

```
# Mean and SD of the order filled correctly.

print('Mean' , n*p)
print('STD' , np.sqrt(n*p*(1-p)))
# On average 2.6 are filled outoff 3
```

```
Mean 2.604
STD 0.5862832080146932
```

According to the Harris International survey conducted for world vision
relesed in Feb 2009, it is found that 56% of the teens in USA volunteer for
charitable cuases. A sample of 60 teens is selected. Let x be the no of teens selected as sample for the cuse
Find the mean and standard deviation of prob dist of x

In [87]:

```
n = 60
p = 0.56
q = 1-p
```

In [93]:

```
mean = n*p #exected  value
std = np.sqrt(n*p*q)
print(mean) #average 34 people can volunteer out of 60
print(std)  # varies +4 or -4
```

```
33.6
3.8449967490233328
```

In [94]:

```
#Observed value the comes after the experiment
#Expected value the ecpected value after experiment
```

Poisson Probability

1. like in Binomal, the experimnet are repeated in nature however, this is not in the case of Poisson
2. In binomal, I need to have the prior probability of the event howerer, it is not required for Poisson.
3. x (Value that we want to find out) should be random variable.
4. The occurenece of the events should be random (patient without appoment) and independent
5. The Avg No of occurences are needed to find the probability.(Lambda)
6. The Poisson Distribution follows an Interval - Time, Space, Volume etc.

Q On an average a household receives 9.5 telemarketing phone calls per Week.
Find the probability that a randomly selected household receives excatly 6 marketing phone calls

In [97]:

```
lambdaa  = 9.5 #average
x = 6

(lambdaa**x)*(2.718)**(-lambdaa)/720
```

Out[97]:

```
0.07649610765306605
```

In [98]:

```
stats.poisson.pmf(x,lambdaa)
```

Out[98]:

```
0.07642079604012611
```

Q Cynthia's mail order company provides free examination of its products for 7 days.
If not satisfied, return and get the refund. Acc to past record of the company , an average of 2 every 10 products sold by the co. are return for a refund.
Find the prob that 6 out of 40 products sold will be returned for refund.

In [99]:

```
# probability 2/10 = 0.2
# 0.2X40 = 8
```

In [100]:

```
stats.poisson.pmf(6,8)
```

Out[100]:

```
0.12213821545677205
```

Q A Insurance agent sells on an avg 3 life insurance policies using Poisson, Find the prob that in given week, he will sell

1. Some Policies (when some we take max)
2. 2 or more but less than 5 policies.

Given that the sample size is 16

In [153]:

```
#1
avg = 3
cumpoisson = stats.poisson.cdf(np.arange(17),avg)

1 - cumpoisson[0] #Some Policies Sold
#0 policy sale no sense
```

Out[153]:

```
0.950212931632136
```

In [154]:

```
#2
#less than 5 --> 4
#p(4) - p(1)

stats.poisson.cdf(4,avg) - stats.poisson.cdf(1,avg)
```

Out[154]:

```
0.6161149710523164
```

In [116]:

```
#2
( stats.poisson.pmf(np.arange(5),avg).sum() ) - (stats.poisson.pmf(np.arange(2),avg).sum()
```

Out[116]:

```
0.6161149710523163
```

In [117]:

```
#In Poisson Mean and Std are equal (that is lambda)
```

In [118]:

```
### 3rd Distribution
#Continuous Probability Distribution ~ Gaussian
```

Q 4 According to the credit bureau data (2008), college students caaries an average 0f 3173 dollars dept on their credit cards. Suppose that current current debt for the cards is 3173 dollars follwed with a stndard deviation of 800 dollars . find the probability that a credit card depbt randomly selected for a college student is betwwen 2109 dollars and 3605 dollars. Find

In [120]:

```
mu = 3173
sd = 800

zscore1 = (2109 - mu)/sd
zscore2 = (3605 - mu)/sd

stats.norm.cdf(zscore2) - stats.norm.cdf(zscore1)
```

Out[120]:

```
0.6136423481340212
```

Q 5 ABC corp produces many types of soft drinks including Coke. The filling machines are adjusted to pour 12 ounces of soda into each of 12 ounce of soff drink. However, the actual qantity of soda is not excatly 12 ounces. Its Been onserved that net amount of soda follows gaussian curve pattern with a mean of 12 ounces and a SD of 0.015 ounces.

1. Find the probability that arandomly selected can of soda contains 11.97 to 11.99 ounces of soda
2. What %age of the can contains 12.02 to 12.07 ounces of Soda

In [125]:

```
#1
mu = 12
sd = 0.015

zsocre1 = (11.97 - mu)/sd
zsocre2 = (11.99 - mu)/sd

stats.norm.cdf(zsocre2) - stats.norm.cdf(zsocre1)
```

Out[125]:

```
0.2297424055987459
```

In [127]:

```
#2
mu = 12
sd = 0.015

zsocre1 = (12.02 - mu)/sd
zsocre2 = (12.07 - mu)/sd

(stats.norm.cdf(zsocre2) - stats.norm.cdf(zsocre1))*100
```

Out[127]:

```
9.120968909913596
```

Q 5 The life span of monkeys in zoo is 13.1 years on an avg
with as std of 1.5 years

find the prob of a monkey living longer than 14.6 years

In [129]:

```
z1 = (14.6 - 13.1)/1.5

1 - stats.norm.cdf(z1)
```

Out[129]:

```
0.15865525393145707
```

Q 6 A survey on the use of samrt phones was conducted and
it was observed that a smart phone user spends 68 minutes in a day
on his cellphone with a sd of 12 minutes.

1. What proportion of smart phone users are spending more than 90 minutes in sending messages daily
2. What Proportion of customers spends less than 20 minutes.
3. What proportion of customers are spending between 50 mins to 100 mins.

In [130]:

```
mu = 68
sd = 12

#1
z1 = (90-68)/12

1 - stats.norm.cdf(z1)
```

Out[130]:

```
0.03337650758481725
```

In [131]:

```
#2
z2 = (20-68)/12
stats.norm.cdf(z2)
```

Out[131]:

```
3.167124183311986e-05
```

In [132]:

```
#3
stats.norm.cdf((100 - 68)/12) - stats.norm.cdf((50 - 68)/12)
```

Out[132]:

```
0.9293624181635521
```

In [ ]:

```

```