Problem Statement - Binomial Distribution

A bank issues credit cards to customers under the scheme of Master Card. Based on the past data, the bank has found out that 60% of all accounts pay on time following the bill. If a sample of 7 accounts is selected at random from the current database, construct the Binomial Probability Distribution of accounts paying on time.

Work Out Using Python

```
In [1]: import numpy as np
        import scipy.stats as stats
        import matplotlib.pyplot as plt
```

```
In [2]: n = 7
```

```
In [3]: p = 0.6
```

```
In [4]: k = np.arange(0,8)
```

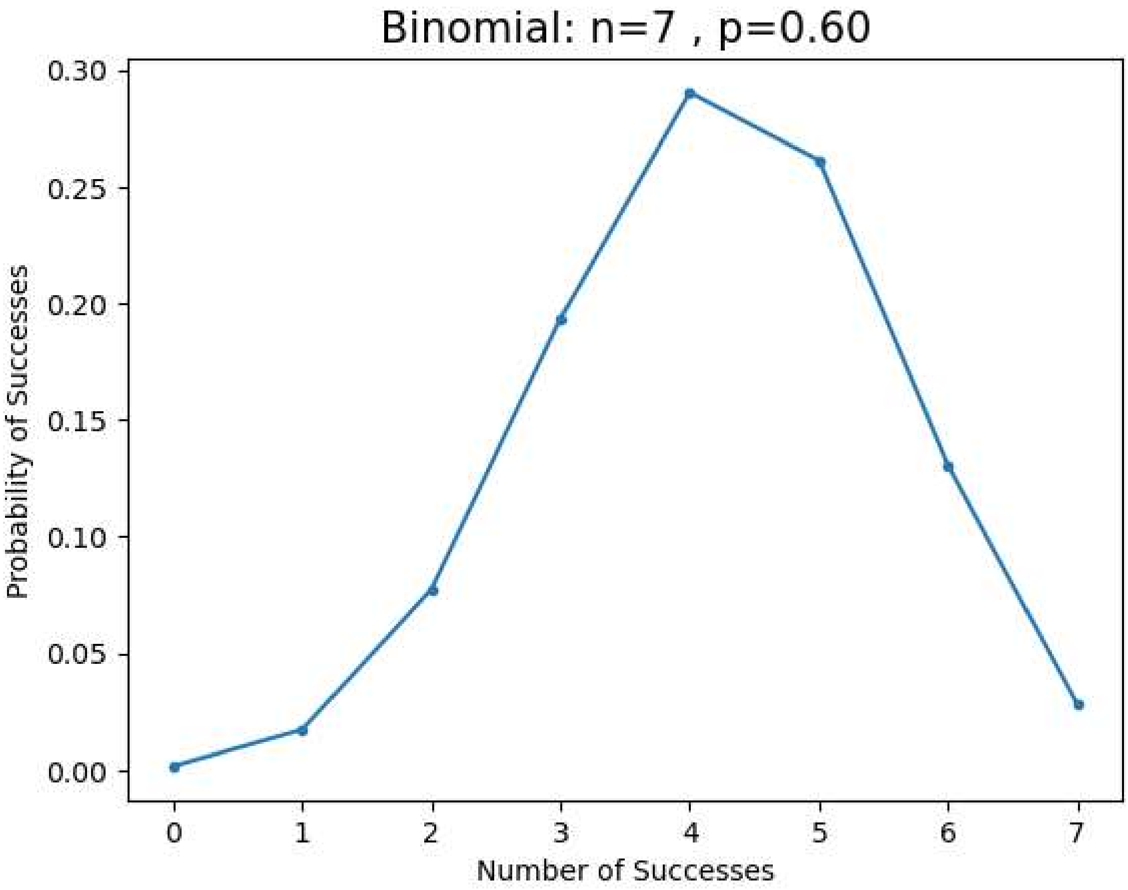
```
In [5]: binomial = stats.binom.pmf(k,n,p)
```

```
In [6]: binomial
```

```
Out[6]: array([0.0016384, 0.0172032, 0.0774144, 0.193536 , 0.290304 , 0.2612736,
               0.1306368, 0.0279936])
```

In [8]:

```
plt.plot(k,binomial, '.-')
plt.title('Binomial: n=%i , p=%.2f' % (n,p), fontsize=15)
plt.xlabel('Number of Successes')
plt.ylabel('Probability of Successes')
plt.show()
```



In [10]:

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

In [11]:

```
cumbinomial
```

Out[11]:

```
array([0.0016384, 0.0188416, 0.096256 , 0.289792 , 0.580096 , 0.8413696,
       0.9720064, 1.       ])
```

In [12]:

```
#
```

In [13]:

```
#
```

In [14]:

```
#
```

In [15]:

```
#
```

Problem Statement - Binomial Distribution

A Laptop assembly Unit manufactures a single type of Laptop product and sells it to its customers under multiple Brand names through its Channel Partners. From past audit and quality data, it knows that on an average, 10% of all manufactured units since inception have been defective. The manufacturer receives a wholesale order of 100 new laptops that are to have highest form of quality and no defectie units. However, it is the liability of the Manufacturer to inform the Client about the probability of a few units being defective out of the whole lot. Construct the Binomial Probability Distribution of Laptops that are defective out of the 100.

Work Out Using Python

In [16]:

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

In [18]:

```
p = 0.1
```

In [19]:

```
 n = 100
```

In [20]:

```
k = np.arange(0,101)
```

In [21]:

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

In [22]:

```
binomial
```

Out[22]:

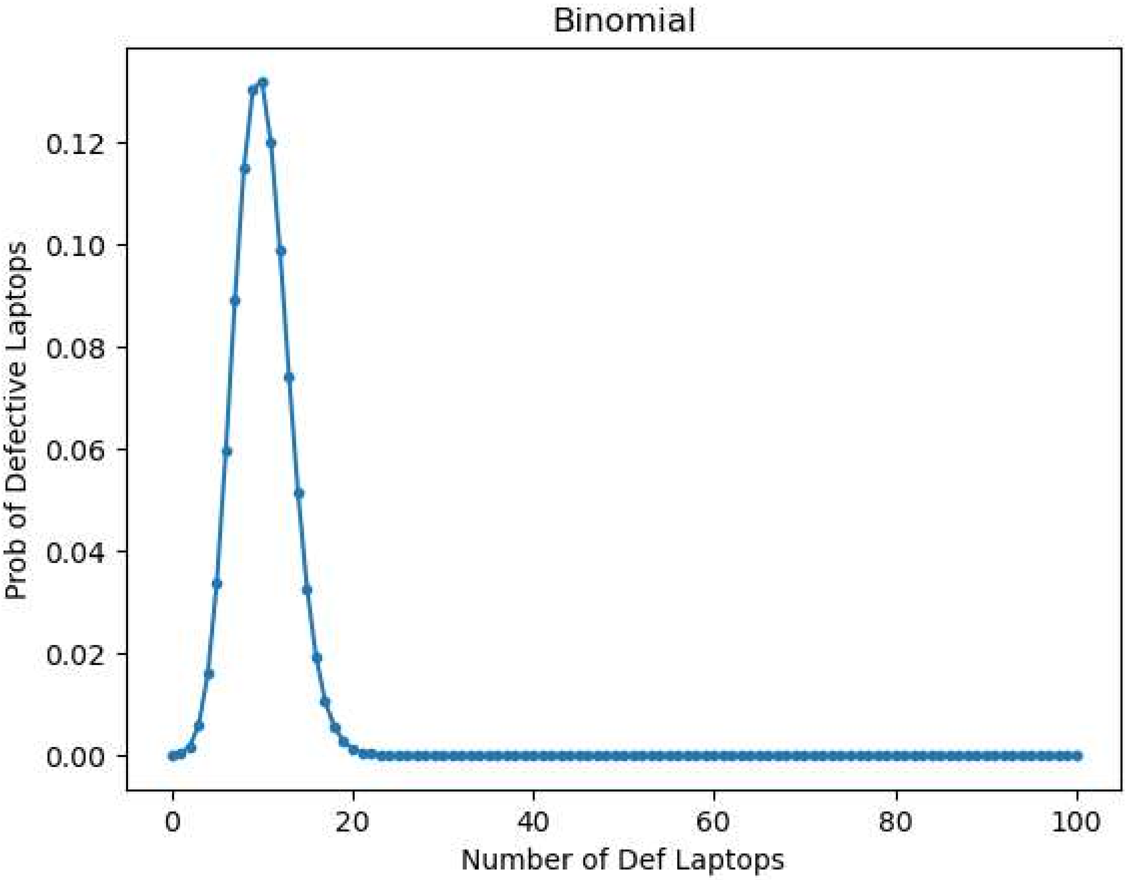
```
array([2.65613989e-005, 2.95126654e-004, 1.62319660e-003, 5.89160247e-003,
       1.58745955e-002, 3.38658038e-002, 5.95787289e-002, 8.88952464e-002,
       1.14823027e-001, 1.30416277e-001, 1.31865347e-001, 1.19877588e-001,
       9.87880124e-002, 7.43020948e-002, 5.13038273e-002, 3.26824382e-002,
       1.92917170e-002, 1.05915309e-002, 5.42652508e-003, 2.60219331e-003,
       1.17098699e-003, 4.95655869e-004, 1.97761685e-004, 7.45188959e-005,
       2.65646064e-005, 8.97293372e-006, 2.87594029e-006, 8.75800748e-007,
       2.53704185e-007, 6.99873614e-008, 1.84040839e-008, 4.61751209e-009,
       1.10627894e-009, 2.53289454e-010, 5.54588020e-011, 1.16199395e-011,
       2.33116070e-012, 4.48030885e-013, 8.25320051e-014, 1.45783029e-014,
       2.47021243e-015, 4.01660558e-016, 6.26930501e-017, 9.39585764e-018,
       1.35243405e-018, 1.87003227e-019, 2.48434239e-020, 3.17150093e-021,
       3.89096178e-022, 4.58798215e-023, 5.19971310e-024, 5.66417549e-025,
       5.93044015e-026, 5.96773852e-027, 5.77126976e-028, 5.36320018e-029,
       4.78857159e-030, 4.10715692e-031, 3.38329018e-032, 2.67604873e-033,
       2.03181478e-034, 1.48037507e-035, 1.03467074e-036, 6.93430129e-038,
       4.45432548e-039, 2.74112337e-040, 1.61514003e-041, 9.10692557e-043,
       4.91059712e-044, 2.53042042e-045, 1.24512751e-046, 5.84566904e-048,
       2.61611732e-049, 1.11493584e-050, 4.52001018e-052, 1.74104096e-053,
       6.36345379e-055, 2.20379352e-056, 7.22040612e-058, 2.23416223e-059,
       6.51630651e-061, 1.78773841e-062, 4.60257857e-064, 1.10905508e-065,
       2.49390692e-067, 5.21601447e-069, 1.01085552e-070, 1.80740450e-072,
       2.96669931e-074, 4.44449334e-076, 6.03573170e-078, 7.36963577e-080,
       8.01047367e-082, 7.65636671e-084, 6.33505520e-086, 4.44565277e-088,
       2.57271572e-090, 1.17879300e-092, 4.00950000e-095, 9.00000000e-098,
       1.00000000e-100])
```

In [23]:

```
plt.plot(k,binomial, '.-')
plt.title('Binomial')
plt.xlabel('Number of Def Laptops')
plt.ylabel('Prob of Defective Laptops')
```

Out[23]:

```
Text(0, 0.5, 'Prob of Defective Laptops')
```



In [ ]: