In [1]:
```python
m = 400
n = 2000
JointProbability = m/n
print("Joint probability of the people who planned to purchase and actually placed an
```

```
Joint probability of the people who planned to purchase and actually placed an order
=  0.2
```

In [2]:
```python
m = 400
n = 500
JointProbability = m/n
print("Joint probability of the people who planned to purchase and actually placed an
```

```
Joint probability of the people who planned to purchase and actually placed an order,
given that people planned to purchase =  0.8
```

In [3]:
```python
import numpy as np
import scipy.stats as stats
import matplotlib.pyplot as plt
print("Libraries: Numpy, Scipy.Stas & Matplotlib imported successfully")
```

```
Libraries: Numpy, Scipy.Stas & Matplotlib imported successfully
```

In [4]:
```python
p   =  0.05
n   =  10
k   =  np.arange(0,11)
print('Failure rate for manufactured items:p =',p,"\nSample size:n =",n,"\nArray of Sa
```

```
Failure rate for manufactured items:p = 0.05
Sample size:n = 10
Array of Samples:k = [ 0  1  2  3  4  5  6  7  8  9 10]
```

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

```
[5.98736939e-01 3.15124705e-01 7.46347985e-02 1.04750594e-02
 9.64808106e-04 6.09352488e-05 2.67259863e-06 8.03789062e-08
 1.58642578e-09 1.85546875e-11 9.76562500e-14]
```

In [6]:
```python
print('Probability that none of the items are defective is %1.4f' %binomial[0])
```

```
Probability that none of the items are defective is 0.5987
```

In [7]:
```python
print('Probability that exactly one of the items is defective is %1.4f' %binomial[1])
```

```
Probability that exactly one of the items is defective is 0.3151
```

In [8]:
```python
cumbinomial = stats.binom.cdf(k,n,p)
print(cumbinomial)
print("")
print('Probability that two or fewer of the items are defective is %1.4f' %cumbinomial
```

```
[0.59873694 0.91386164 0.98849644 0.9989715  0.99993631 0.99999725
 0.99999992 1.         1.         1.         1.        ]

Probability that two or fewer of the items are defective is 0.9885
```

In [9]:
```python
P = 1- cumbinomial[2]
print('Probability that three or more of the items are defective is %1.4f' % P)
```

```
Probability that three or more of the items are defective is 0.0115
```

In [10]:

```
rate =  3
n    =  np.arange(0,20)
cumpoisson = stats.poisson.cdf(n,rate)
print(cumpoisson)
```

```
[0.04978707 0.19914827 0.42319008 0.64723189 0.81526324 0.91608206
 0.96649146 0.9880955  0.99619701 0.99889751 0.99970766 0.99992861
 0.99998385 0.9999966  0.99999933 0.99999988 0.99999998 1.
 1.         1.        ]
```

In [11]:

```
P   = 1 - cumpoisson[0]
print('Probability that in a given week the salesman will sell some cars is %1.4f' % P
```

```
Probability that in a given week the salesman will sell some cars is 0.9502
```

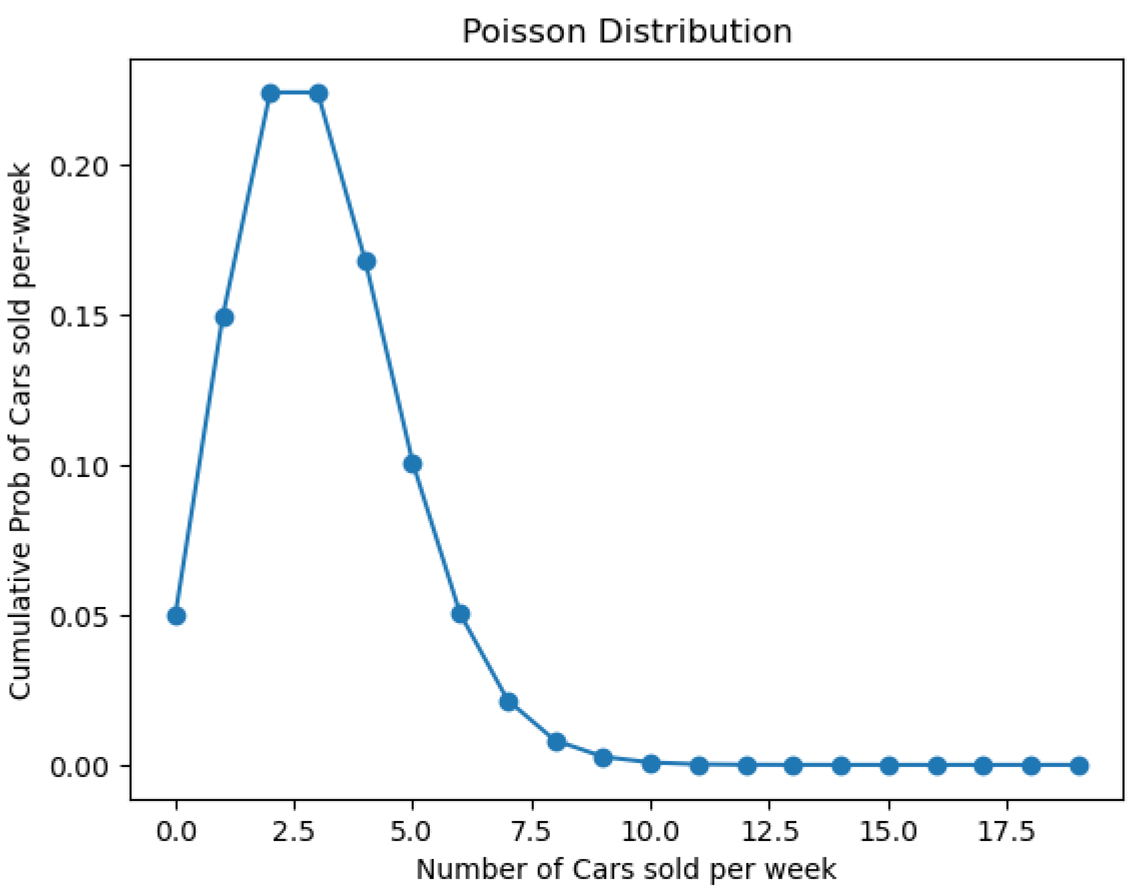
In [12]:

```
P1 = cumpoisson[4] # P(X >=4)
P2 = cumpoisson[1] # P(X >=1)
P = P1 - P2
print('Probability that in a given week the salesman will sell 2 or more but less than
```

```
Probability that in a given week the salesman will sell 2 or more but less than 5 car
s is 0.6161
```

In [13]:

```
poisson = stats.poisson.pmf(n,rate)
plt.plot(n,poisson, 'o-')
plt.title('Poisson Distribution')
plt.xlabel('Number of Cars sold per week')
plt.ylabel('Cumulative Prob of Cars sold per-week')
plt.show()
```



In [14]:
```
p1   =  0.868
n1   =  3
k1   =  np.arange(0,4)
print('Recognition:p =',p1,"\nSample size:n =",n1,"\nArray of Samples:k =",k1)
```

```
Recognition:p = 0.868
Sample size:n = 3
Array of Samples:k = [0 1 2 3]
```

In [15]:
```
binomial = stats.binom.pmf(k1,n1,p1)
print(binomial)
```

```
[0.00229997 0.0453721  0.2983559  0.65397203]
```

In [16]:
```
print('Probability that all three orders will be recognised correctly is %1.4f' %binom
```

```
Probability that all three orders will be recognised correctly is 0.6540
```

In [17]:
```
print('Probability that none of the three orders will be recognised correctly is %1.4f
```

```
Probability that none of the three orders will be recognised correctly is 0.0023
```

In [18]:
```
P_2 = binomial[2]
P_3 = binomial[3]
print("Probability that at least two of the three orders will be recognised correctly
```

```
Probability that at least two of the three orders will be recognised correctly is  0.
9523
```

In [ ]:
```
The number of vehicles produced per shift by a mass vehicle manufacturer is very impor

What is the probability to produce 'x' cars in a shift given the rate of production pe
What is the probability to produce cars between 'x' and 'y' in a shift, given the rate
```