Exp. No.16: TO STUDY THE CHARACTERISTICS OF GM COUNTER AND IT'S RELIABILITY.

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

```
df = pd.read_csv('https://raw.githubusercontent.com/jsdhami/Python-For-Research/main/Data/PhysicsExp16.csv')
print(df.head())
```

```
    x    f
0  90  1.0
1  91  NaN
2  92  1.0
3  93  NaN
4  94  2.0
```

```
print(df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 51 entries, 0 to 50
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   x       51 non-null     int64
 1   f       47 non-null     float64
dtypes: float64(1), int64(1)
memory usage: 944.0 bytes
None
```

```
df.dropna(inplace = True)
df = df[df.f != 1]
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 37 entries, 4 to 45
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   x       37 non-null     int64
 1   f       37 non-null     float64
dtypes: float64(1), int64(1)
memory usage: 888.0 bytes
```

```
# To Find λ (Where, λ is mean)
N = df.f.sum()
print("Total No of Observation =", N)
λ = ((df.f * df.x).sum()/N)
print("Value of Σfx/N =", λ)

# Calculating the Poisson's distribution frequency
def p_f(x, λ, N):
  return (N*λ**x*np.exp(-λ))/math.factorial(x)

# add new column
df['p_f'] = df['x'].apply(lambda x: p_f(x, λ, N))

print(df.head())
```

```
Total No of Observation = 192.0
Value of Σfx/N = 114.44791666666667
    x    f       p_f
4  94  2.0  1.127116
5  95  3.0  1.357853
7  97  2.0  1.909966
8  98  2.0  2.230527
9  99  2.0  2.578577
```

```
# pot
fig, ax = plt.subplots()
fig.set_size_inches(20, 14)
ax.plot(df.x, df.f, 'o-', label = 'Counts Vs Frequencies')
ax.plot(df.x, df.p_f, 'o-', label = 'Counts Vs Poisson\'s Frequencies', color='red')
ax.bar(df.x, df.p_f, label = 'Poisson\'s Frequencies', color='yellow')
ax.set_xlabel('Counts from GM Counter')
ax.set_ylabel('frequency | p_f(x) frequency')
ax.set_title('Reliability of GM Counter')
ax.grid()
ax.legend()
plt.show()
```