EXCERSICE 5.1

In [12]:
```
import pandas as pd
```

In [13]:
```
# Load the dataset
df = pd.read_csv("C:/Users/user/Downloads/a22126551037.csv")
```

In [14]:
```
# Display  OF ROWS
print(df.head())
```

```
   Confirmed  Deaths  Recovered  Active  New cases  New deaths  New recovered  \
0      36263    1269      25198    9796        106          10             18
1       4880     144       2745    1991        117           6             63
2      27973    1163      18837    7973        616           8            749
3        907      52        803      52         10           0              0
4        950      41        242     667         18           1              0

   Deaths / 100 Cases  Recovered / 100 Cases  Deaths / 100 Recovered  \
0                3.50                  69.49                    5.04
1                2.95                  56.25                    5.25
2                4.16                  67.34                    6.17
3                5.73                  88.53                    6.48
4                4.32                  25.47                   16.94

   Confirmed last week  1 week change  1 week % increase
0                35526            737               2.07
1                 4171            709              17.00
2                23691           4282              18.07
3                  884             23               2.60
4                  749            201              26.84
```

In [15]:
```
#descriptive statistics
print("Mean:\n", df.mean())
print("\nMedian:\n", df.median())
print("\nMode:\n", df.mode().iloc[0])
print("\nStandard Deviation:\n", df.std())
print("\nVariance:\n",df.var())
print("\nRange:\n", df.max() - df.min())
print("\nSkewness:\n", df.skew())
print("\nKurtosis:\n", df.kurt())
```

```
Mean:
 Confirmed                 8.813094e+04
Deaths                    3.497519e+03
Recovered                 5.063148e+04
Active                    3.400194e+04
New cases                 1.222957e+03
New deaths                2.895722e+01
New recovered             9.338128e+02
Deaths / 100 Cases        3.019519e+00
Recovered / 100 Cases     6.482053e+01
Deaths / 100 Recovered             inf
Confirmed last week       7.868248e+04
1 week change             9.448460e+03
1 week % increase         1.360620e+01
dtype: float64

Median:
 Confirmed                 5059.00
Deaths                     108.00
Recovered                 2815.00
Active                    1600.00
New cases                   49.00
New deaths                   1.00
New recovered               22.00
Deaths / 100 Cases           2.15
Recovered / 100 Cases       71.32
Deaths / 100 Recovered       3.62
Confirmed last week       5020.00
1 week change              432.00
1 week % increase            6.89
dtype: float64

Mode:
 Confirmed                 24.0
Deaths                     0.0
Recovered                  0.0
Active                     0.0
New cases                  0.0
New deaths                 0.0
New recovered              0.0
Deaths / 100 Cases         0.0
Recovered / 100 Cases      0.0
Deaths / 100 Recovered     0.0
Confirmed last week       19.0
1 week change              0.0
1 week % increase          0.0
Name: 0, dtype: float64

Standard Deviation:
 Confirmed                 383318.663831
Deaths                     14100.002482
Recovered                 190188.189643
Active                    213326.173371
New cases                   5710.374790
New deaths                   120.037173
New recovered               4197.719635
Deaths / 100 Cases             3.454302
Recovered / 100 Cases         26.287694
Deaths / 100 Recovered              NaN
Confirmed last week       338273.676567
1 week change              47491.127684
1 week % increase             24.509838
dtype: float64

Variance:
 Confirmed                 1.469332e+11
Deaths                    1.988101e+08
Recovered                 3.617155e+10
Active                    4.550806e+10
New cases                 3.260838e+07
New deaths                1.440892e+04
New recovered             1.762085e+07
Deaths / 100 Cases        1.193221e+01
Recovered / 100 Cases     6.910429e+02
Deaths / 100 Recovered             NaN
Confirmed last week       1.144291e+11
1 week change             2.255407e+09
1 week % increase         6.007321e+02
dtype: float64

Range:
 Confirmed                 4290249.00
Deaths                     148011.00
Recovered                 1846641.00
Active                    2816444.00
New cases                   56336.00
New deaths                   1076.00
New recovered               33728.00
Deaths / 100 Cases             28.56
Recovered / 100 Cases         100.00
Deaths / 100 Recovered           inf
Confirmed last week       3834667.00
1 week change              455629.00
1 week % increase             230.16
dtype: float64

Skewness:
 Confirmed                  8.725676
Deaths                     7.464481
Recovered                  6.983644
Active                    12.182067
New cases                  7.720320
New deaths                 5.970033
New recovered              6.769567
Deaths / 100 Cases         3.352173
Recovered / 100 Cases     -0.823366
Deaths / 100 Recovered          NaN
Confirmed last week        8.865198
1 week change              7.692012
1 week % increase          6.114613
dtype: float64

Kurtosis:
 Confirmed                   86.096572
Deaths                      66.480494
Recovered                   55.600771
Active                     157.921665
New cases                   65.022330
New deaths                  40.101549
New recovered               47.910082
Deaths / 100 Cases          17.541183
Recovered / 100 Cases       -0.115728
Deaths / 100 Recovered            NaN
Confirmed last week         89.376884
1 week change               61.662738
1 week % increase           45.808865
dtype: float64
C:\Users\user\AppData\Local\Programs\Python\Python312\Lib\site-packages\pandas\core\nanops.py:1016: RuntimeWarning: invalid value encountered in subtract
  sqr = _ensure_numeric((avg - values) ** 2)
C:\Users\user\AppData\Local\Programs\Python\Python312\Lib\site-packages\pandas\core\nanops.py:1256: RuntimeWarning: invalid value encountered in subtract
  adjusted = values - mean
C:\Users\user\AppData\Local\Programs\Python\Python312\Lib\site-packages\pandas\core\nanops.py:1344: RuntimeWarning: invalid value encountered in subtract
  adjusted = values - mean
```

In [16]:
```
# Import libraries
import numpy as np
from scipy import stats
```

In [17]:
```
chosen_value = 1000
new_cases = df['New cases'].dropna()
t_stat, p_value = stats.ttest_1samp(new_cases, chosen_value)
print(f"T-statistic: {t_stat}")
print(f"P-value: {p_value}")
if p_value < 0.05:
    print("The average number of new cases is significantly different from 1000.")
else:
    print("The average number of new cases is not significantly different from 1000.")
```

```
T-statistic: 0.5339218930257723
P-value: 0.5940330965326368
The average number of new cases is not significantly different from 1000.
```

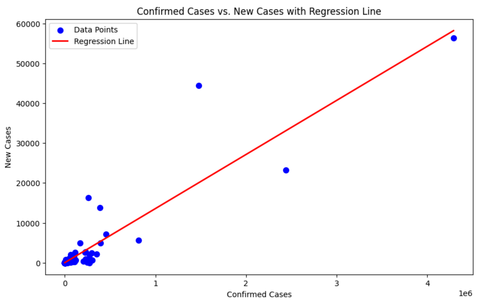
In [18]:
```
# Extract the Active cases column
active_cases = df['Active'].dropna()
# Calculate mean and standard error
mean = active_cases.mean()
std_err = stats.sem(active_cases)
```

In [25]:
```
# Compute 95% confidence interval
confidence_interval = stats.norm.interval(0.95, loc=mean, scale=std_err)
print(f"Mean of Active cases: {mean}")
print(f"95% Confidence Interval: {confidence_interval}")
```

```
Mean of Active cases: 34001.935828877
95% Confidence Interval: (np.float64(3426.586273930501), np.float64(64577.2853838235))
```

EXCERSICE 5.2

In [26]:
```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
# Scatter plot
plt.figure(figsize=(10, 6))
plt.scatter(df['Confirmed'], df['New cases'], color='blue', s=50, label='Data Points')
# Fit a linear regression model using NumPy
# X and y values
X = df['Confirmed'].values
y = df['New cases'].values
# Add a constant to the model (intercept)
X_with_const = np.vstack([np.ones_like(X), X]).T
# Calculate the coefficients using the Ordinary Least Squares (OLS) method
coefficients = np.linalg.lstsq(X_with_const, y, rcond=None)[0]
intercept, slope = coefficients
# Generate regression line values
X_range = np.linspace(X.min(), X.max(), 100)
y_pred = intercept + slope * X_range
# Plot regression line
plt.plot(X_range, y_pred, color='red', linewidth=2, label='Regression Line')# Add labels and title
plt.xlabel('Confirmed Cases')
plt.ylabel('New Cases')
plt.title('Confirmed Cases vs. New Cases with Regression Line')
plt.legend()
# Show plot
plt.show()
```


Confirmed Cases vs. New Cases with Regression Line
Data Points
Regression Line
New Cases
Confirmed Cases
1e6
0
10000
20000
30000
40000
50000
60000
0
1
2
3
4

In [ ]: