ESTADISTICA AVANZADA

PROF. JUAN IVAN NIETO HIPOLITO

REPORTE DE PRACTICA
INTERVALOS DE CONFIANZA PARA MEDIAS

KEVIN ALEJANDRO GONZALEZ TORRES
GRUPO 932

EJERCICIO 1:

```python
import scipy.stats as stats
import math

std_deviation = 40
sample_size = 30
sample_mean = 780
confidence_level = 0.96

standard_error_mean = std_deviation / math.sqrt(sample_size)

degrees_of_freedom = sample_size - 1
critical_value = stats.t.ppf((1 + confidence_level) / 2, df=degrees_of_freedom)

confidence_interval = (
    sample_mean - critical_value * standard_error_mean,
    sample_mean + critical_value * standard_error_mean
)

print(f"Confidence interval for {confidence_level * 100}% confidence level of the population mean:")
print(f"({confidence_interval[0]}, {confidence_interval[1]}) hours")

```

```
In [3]: runfile('C:/Users/kevin/Desktop/UABC/TercerSemestre/Estadistica Avanzada/
Practica 3/Practica3_1.py', wdir='C:/Users/kevin/Desktop/UABC/TercerSemestre/
Estadistica Avanzada/Practica 3')
Confidence interval for 96.0% confidence level of the population mean:
(764.2962459123092, 795.7037540876908) hours
```

EJERCICIO 2:

```python
import scipy.stats as stats
import math

std_deviation = 0.0015
sample_size = 75
sample_mean = 0.310
confidence_level = 0.95

standard_error_mean = std_deviation / math.sqrt(sample_size)

degrees_of_freedom = sample_size - 1
critical_value = stats.t.ppf((1 + confidence_level) / 2, df=degrees_of_freedom)

confidence_interval = (
    sample_mean - critical_value * standard_error_mean,
    sample_mean + critical_value * standard_error_mean
)

print(f"{confidence_level * 100}% confidence interval for the mean depth:")
print(f"({confidence_interval[0]:.5f}, {confidence_interval[1]:.5f}) inches")

```

```
In [6]: runfile('C:/Users/kevin/Desktop/UABC/TercerSemestre/Estadistica Avanzada/
Practica 3/Practica3_2.py', wdir='C:/Users/kevin/Desktop/UABC/TercerSemestre/
Estadistica Avanzada/Practica 3')
95.0% confidence interval for the mean depth:
(0.30965, 0.31035) inches
```

EJERCICIO 3:

```python
import scipy.stats as stats
import math

population_mean = 3
standard_deviation = 1.6
sample_size = 48
confidence_level = 0.95

standard_error_mean = standard_deviation / math.sqrt(sample_size)

degrees_of_freedom = sample_size - 1
critical_value = stats.t.ppf((1 + confidence_level) / 2, df=degrees_of_freedom)

confidence_interval = (
    population_mean - critical_value * standard_error_mean,
    population_mean + critical_value * standard_error_mean
)

print(f"{confidence_level * 100}% confidence interval for the mean price:")
print(f"(${confidence_interval[0]:.2f}, ${confidence_interval[1]:.2f}) per kilogram")

```

```
In [9]: runfile('C:/Users/kevin/Desktop/UABC/TercerSemestre/Estadistica Avanzada/
Practica 3/Practica3_3.py', wdir='C:/Users/kevin/Desktop/UABC/TercerSemestre/
Estadistica Avanzada/Practica 3')
95.0% confidence interval for the mean price:
($2.54, $3.46) per kilogram
```