STATISTICS PROJECT

Sleeping Habits of Students

Kagita Meenakshi-MA23BTECH11013

Meka Bhavya Kumari-MA23BTECH11016

Reddy Roshni-MA23BTECH11021

Introduction:

The report analyses the sleep patterns and experiences of students. We conducted a survey of 342 students to get a better understanding of sleep habits and various factors affecting the sleep health. The survey investigated various factors affecting student sleep habits, including:

Background Information:

- Degree being pursued by student
- Gender
- Age

Sleep Patterns

- Sleep Duration on weekdays
- Sleep Duration on weekends

Sleep Quality:

- Factors affecting sleep quality

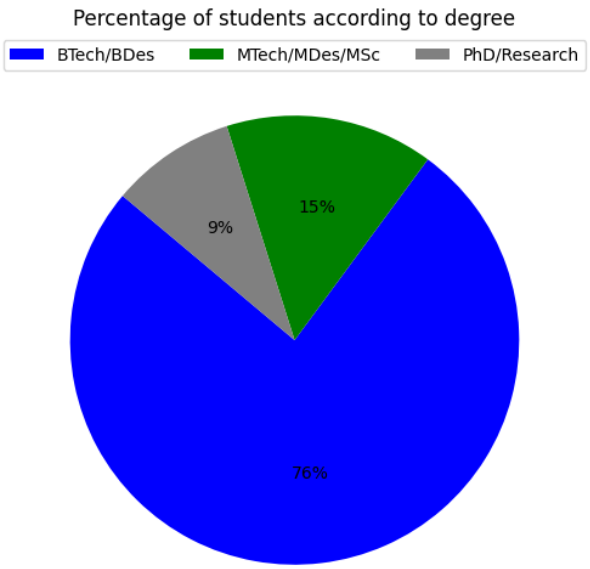
Sleep and Academics:

- Impact of irregular sleep on academics

Sleep Satisfaction:

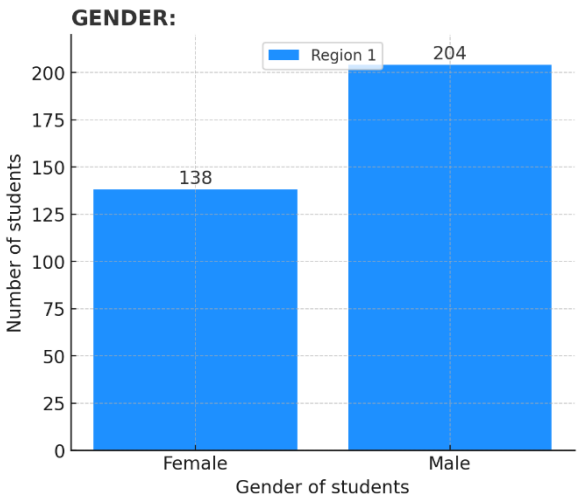
- Consistency of sleep schedule
- Students sleep satisfaction

INTERPRETATION OF DATA: DEGREE BEING PURSUED:



From the observations made from the survey ,76% of students are pursuing BTech/BDes, 9% of students are pursuing PhD/Research, 15% students are pursuing MTech/MDes/MSc.

GENDER:



From the bar graph we can observe that among the students responded 204 are male and 138 are female

CODES :

1. Bar Diagram (Average Sleep Hours on Weekdays vs. Weekends)

```
import matplotlib.pyplot as plt

# Data
categories = ['Weekdays', 'Weekends']
values = [6.64, 7.90]  # Mean sleep hours from the dataset

# Bar Chart
plt.bar(categories, values, color='skyblue')
plt.xlabel("Days")
plt.ylabel("Average Sleep Hours")
plt.title("Average Sleep Hours (Weekdays vs. Weekends)")
plt.show()
```

2. Ogive (Cumulative Frequency of Weekday Sleep Hours)

```
import seaborn as sns

# Weekday Sleep Data
weekday_sleep = sorted([6, 7, 5, 8, 6, 9, 4, 7, 6, 5, 8, 7, 6, 9])

# Ogive (Cumulative Frequency)
sns.ecdfplot(weekday_sleep)
plt.xlabel("Weekday Sleep Hours")
plt.ylabel("Cumulative Frequency")
plt.title("Ogive of Weekday Sleep Hours")
plt.grid()
plt.show()
```

3. Histogram (Distribution of Weekday Sleep Hours)

```
import numpy as np

# Weekday Sleep Data
weekday_sleep = [6, 7, 5, 8, 6, 9, 4, 7, 6, 5, 8, 7, 6, 9]

# Histogram
plt.hist(weekday_sleep, bins=5, color='lightcoral',
```

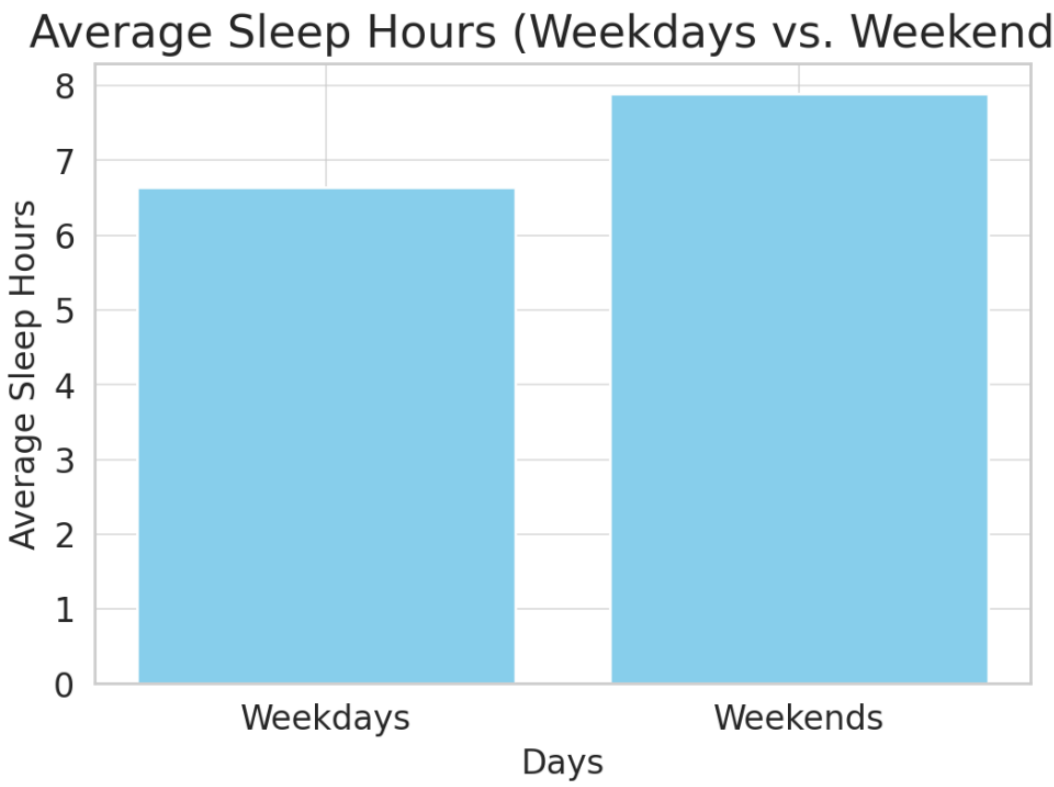
```
edgecolor='black', alpha=0.7)
plt.xlabel("Weekday Sleep Hours")
plt.ylabel("Frequency")
plt.title("Histogram of Weekday Sleep Hours")
plt.grid(axis='y')
plt.show()
```

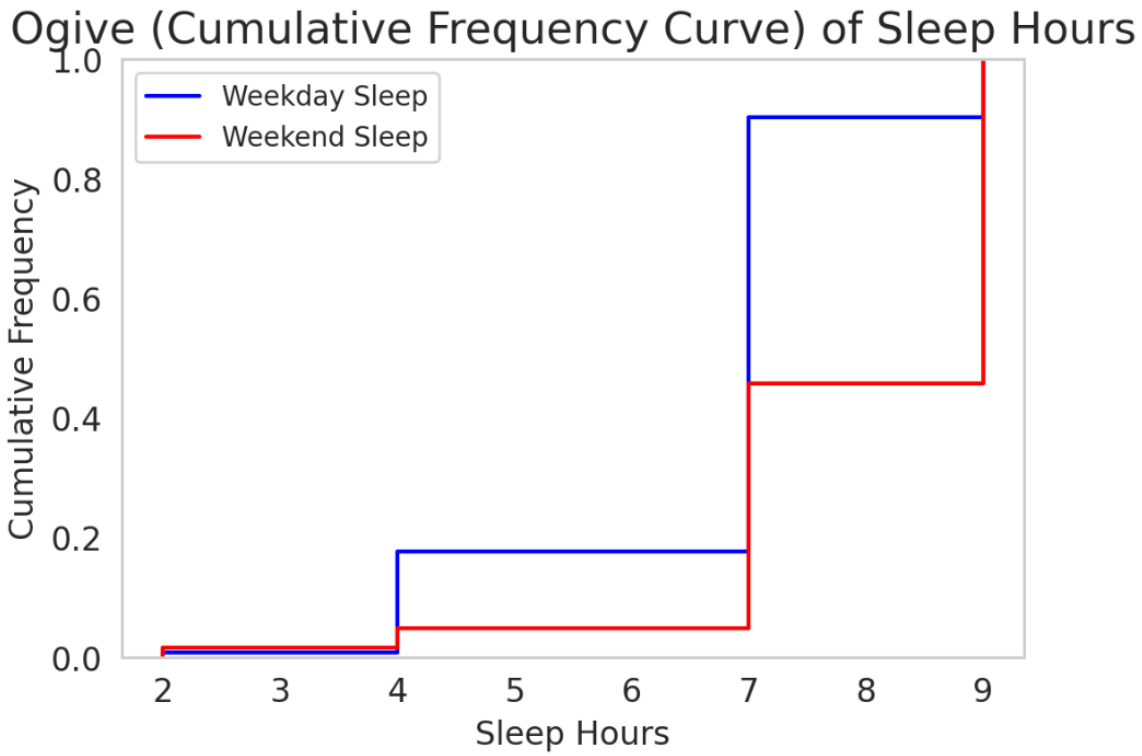
4. Pie Chart (Sleep Hour Categories for Weekdays)

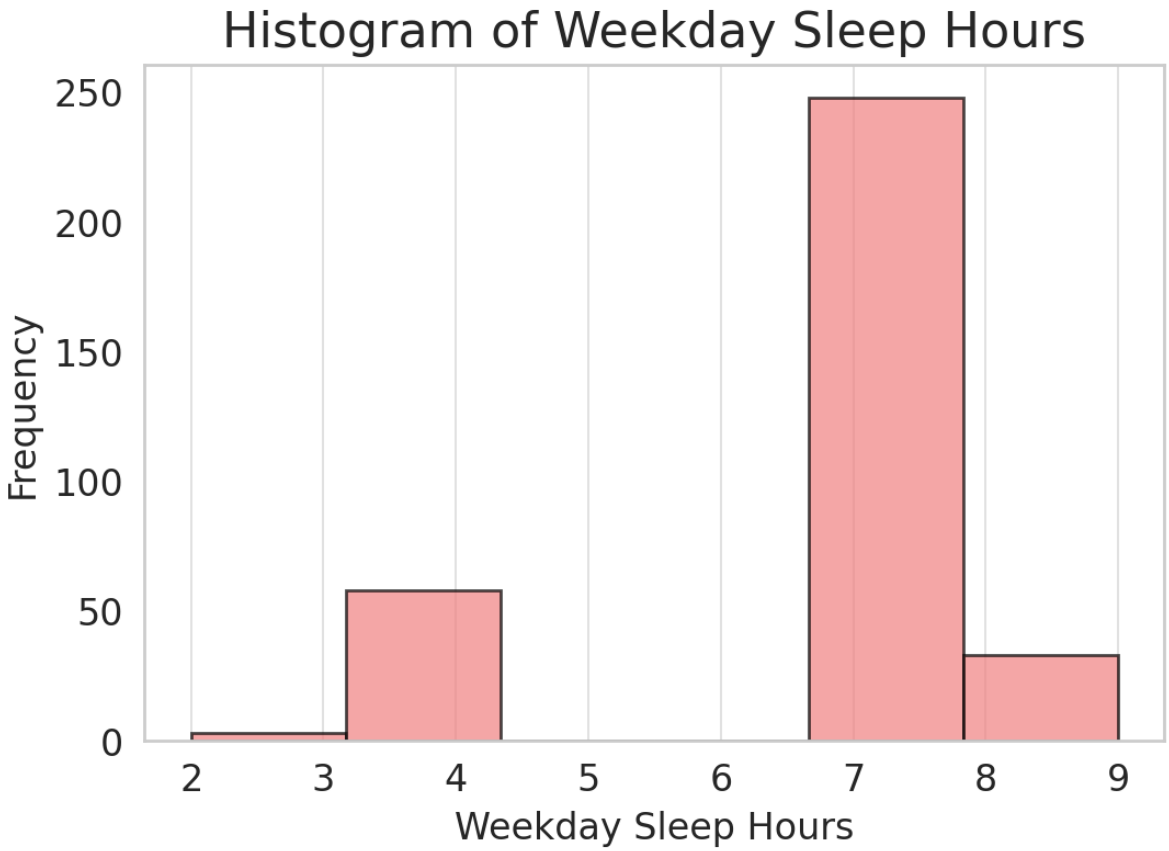
```
import matplotlib.pyplot as plt

#Data
sleep_labels = ["Less than 3 hrs", "3-5 hrs", "6-8 hrs", "More than
8 hrs"]
sleep_counts = [3, 58, 248, 33]

 # Pie Chart
plt.pie(sleep_counts, labels=sleep_labels, autopct="%1.1f%%",
colors=["red", "orange", "green", "blue"], startangle=140)
plt.title("Sleep Hour Categories (Weekdays)")
plt.show()
```

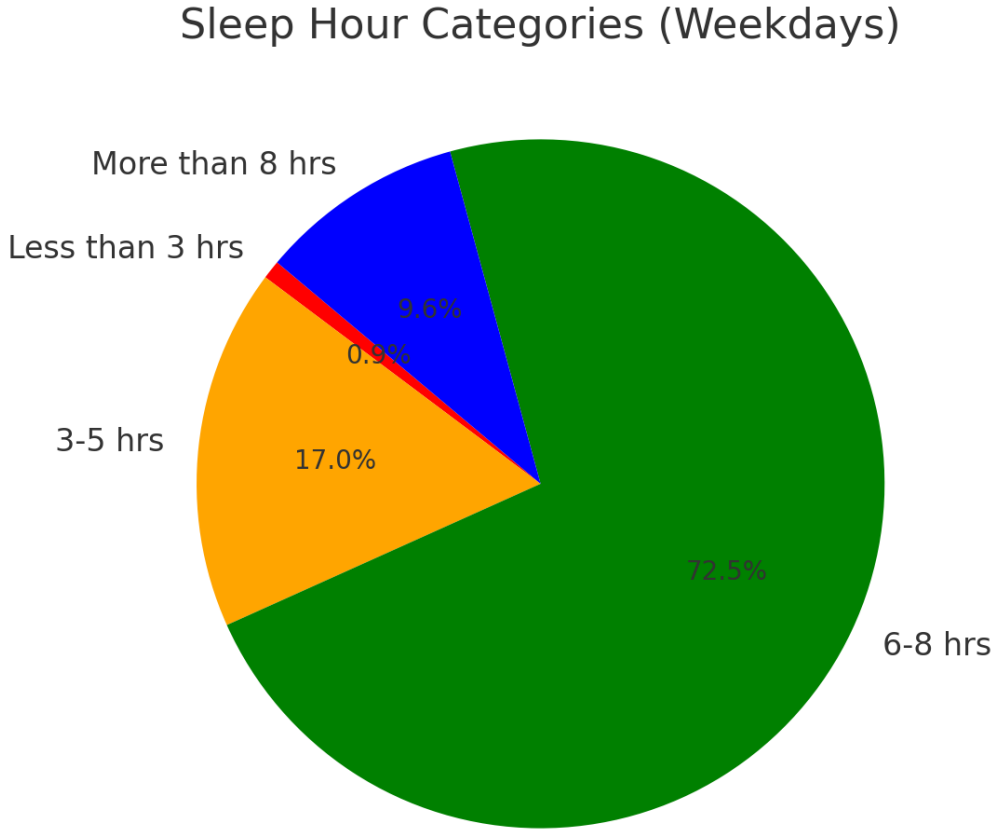
5. Box Plot (Comparison of Weekday vs. Weekend Sleep Hours)

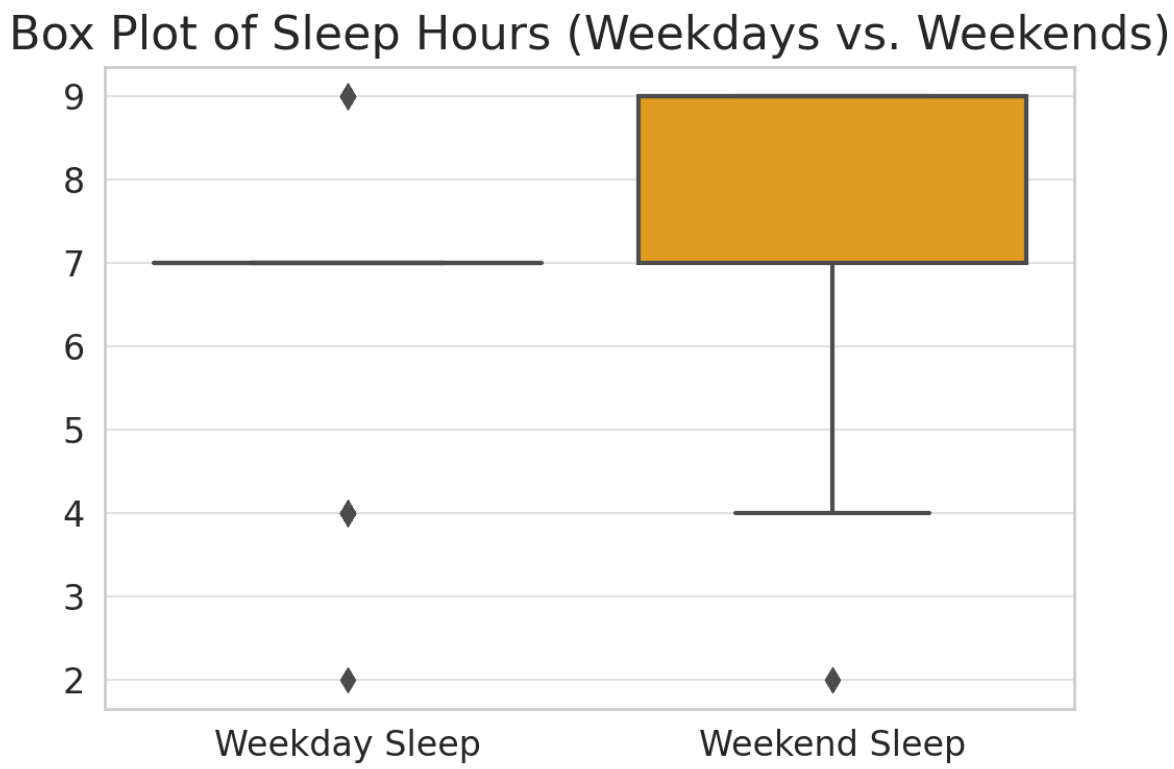
```
import seaborn as sns

# Data
weekday_sleep = [6, 7, 5, 8, 6, 9, 4, 7, 6, 5, 8, 7, 6, 9]
weekend_sleep = [7, 8, 9, 8, 9, 6, 7, 9, 8, 6, 9, 8, 7, 9]

# Box Plot
sns.boxplot(data=[weekday_sleep, weekend_sleep], palette=["cyan",
"orange"])
plt.xticks([0, 1], ["Weekday Sleep", "Weekend Sleep"])
plt.title("Box Plot of Sleep Hours (Weekdays vs. Weekends)")
plt.show()
```


Average Sleep Hours (Weekdays vs. Weekends)
Average Sleep Hours
0
1
2
3
4
5
6
7
8
Weekdays
Weekends
Days


Ogive (Cumulative Frequency Curve) of Sleep Hours
Weekday Sleep
Weekend Sleep
Cumulative Frequency
0.0
0.2
0.4
0.6
0.8
1.0
2
3
4
5
6
7
8
9
Sleep Hours


Histogram of Weekday Sleep Hours
Frequency
250
200
150
100
50
0
2
3
4
5
6
7
8
9
Weekday Sleep Hours


Sleep Hour Categories (Weekdays)
More than 8 hrs
Less than 3 hrs
3-5 hrs
6-8 hrs
9.6%
0.9%
17.0%
72.5%



Descriptive Statistics Summary

1. Measures of Central Tendency

- **Mean (Average Sleep Hours)**
  - Weekdays: **6.64 hours**
  - Weekends: **7.90 hours**
  - Age: **20.62 years**
- **Median (Middle Value)**
  - Weekdays: **7 hours**
  - Weekends: **9 hours**
  - Age: **19.5 years**
- **Mode (Most Frequent Value)**
  - Weekdays: **7 hours**

    - Weekends: **9 hours**
    - Age: **19 years**

2. Measures of Dispersion

- **Standard Deviation (Spread of Data)**
    - Weekdays: **1.42 hours**
    - Weekends: **1.45 hours**
    - Age: **3.24 years**
- **Variance (Square of Std Dev)**
    - Weekdays: **2.01**
    - Weekends: **2.09**
    - Age: **10.47**
- **Interquartile Range (IQR)**
    - Weekdays: **0.00 (Most values are concentrated around 7)**
    - Weekends: **2.00 (More variation in sleep hours)**
    - Age: **2.75 years**

3. Data Range

- **Minimum & Maximum Sleep Hours**
    - Weekdays: **Min = 2, Max = 9**
    - Weekends: **Min = 2, Max = 9**
- **Minimum & Maximum Age**
    - **Min = 17 years, Max = 38 years**

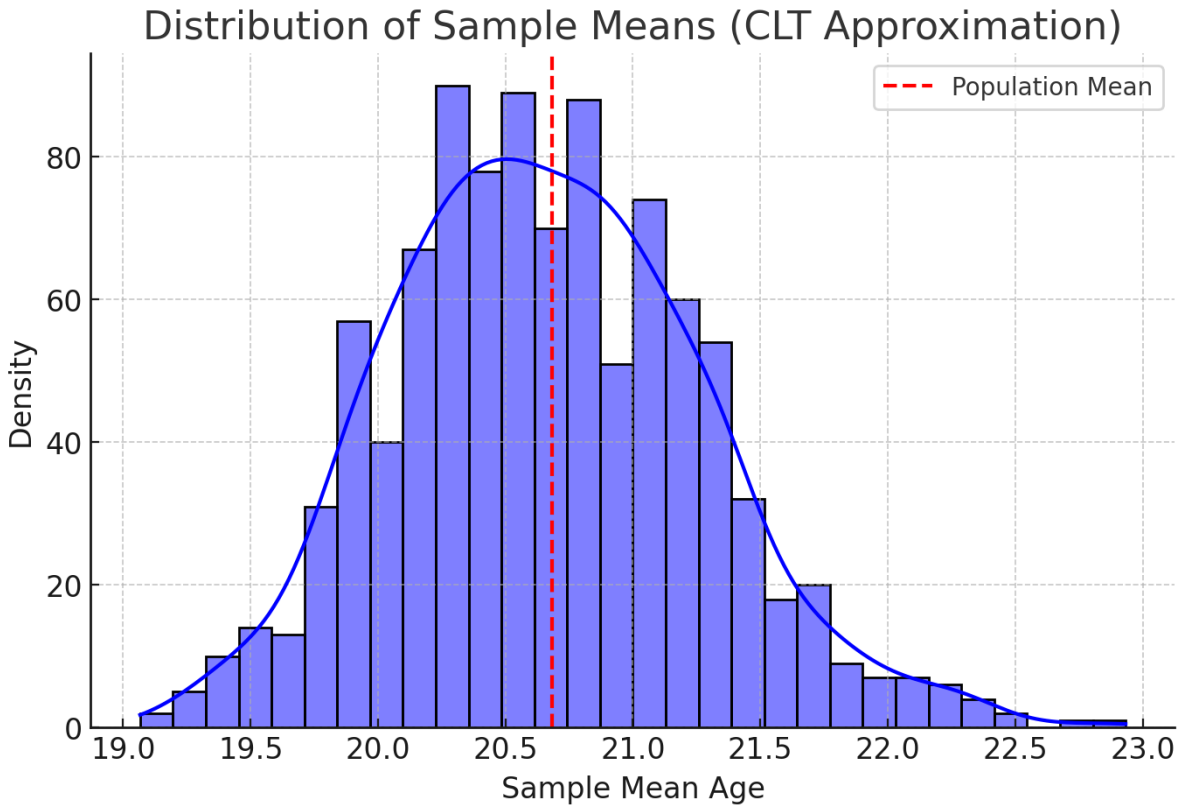
Insights

1. **Sleep hours are higher on weekends** than on weekdays.

2. **Weekday sleep is more consistent**, with an IQR of **0** (most students sleep around 7 hours).
3. **Weekend sleep shows more variation**, with an IQR of **2 hours**, meaning students follow different sleep patterns.
4. **Most students are around 19-21 years old**, with a mean of **20.62 years**.

CLT to approximate the distribution of the sample mean

For applying the Central Limit Theorem (CLT), we are using **'Age'** since it's numerical and suitable for this analysis.



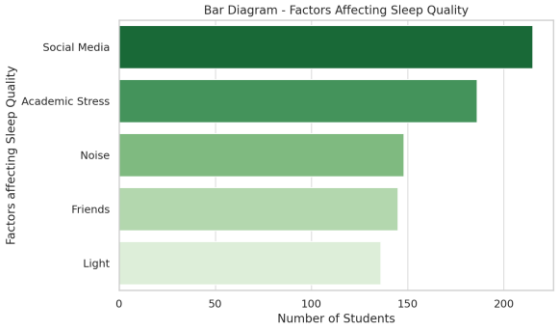
Insights from the CLT Analysis:

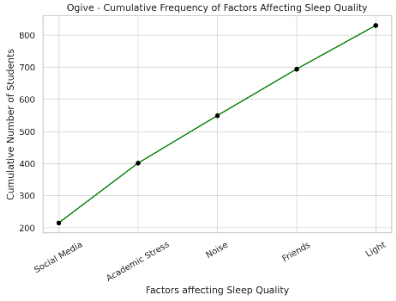
1. **Shape of the Distribution:** The distribution of sample means closely follows a normal distribution, which aligns with the Central Limit Theorem (CLT) regardless of the original distribution of the 'Age' data.
2. **Mean Alignment:** The red dashed line represents the population mean, which matches the peak of the sample mean distribution. This further validates the accuracy of the approximation.
3. **Reduced Variability:** The sample means are more tightly clustered around the true mean than individual data points, showing the expected reduction in variability as sample size increases.

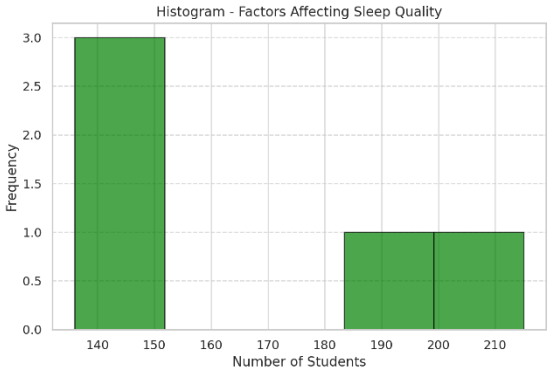
Sleep quality:

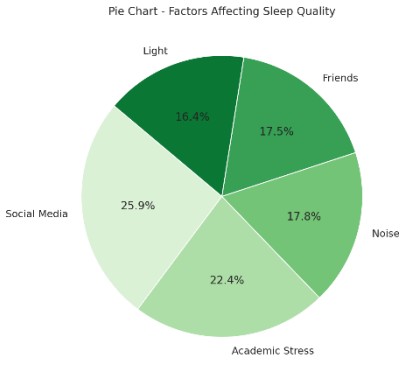
Factors effecting sleep quality:

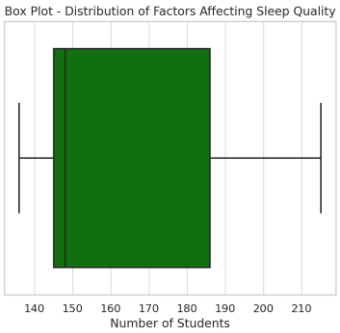
Visual representation:











Among 342 students, Sleep quality of 217 students i.e. 63.4% of the students are affected by social media.

Sleep quality of 188 students i.e. 54.97% of students are affected by Academic stress.

Sleep quality of 148 students i.e. 43.27 % of students are affected by Noise.

Sleep quality of 146 students i.e. 42.69% of students are affected by Friends.

Sleep quality of 136 students i.e. 39.76% of students are affected by Light.

Sleep quality of 27 students i.e 7.89% of students are affected by other factors.

MODE-Social Media

We can see most of students having disturbed sleep due to above factors mostly by Social media.

CODE

```
import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

import numpy as np

import os

from collections import Counter


# Load the dataset

file_path = "/mnt/data/Stats - Sheet1.csv"

if not os.path.exists(file_path):

    raise FileNotFoundError(f"The file '{file_path}' was not found. Ensure it is in the correct directory.")
```

```
df = pd.read_csv(file_path)

# Extract and process the factors affecting sleep quality
column_name = "What other factors affect your sleep quality?"
if column_name not in df.columns:
    raise KeyError(f"The expected column '{column_name}' is not in the dataset.")

factor_list = df[column_name].dropna().str.split(", ")

# Flatten the list and count occurrences
factor_counts = Counter([factor for sublist in factor_list for factor in sublist])

# If no factors are found, raise an error
if not factor_counts:
    raise ValueError("No factors found in the dataset. Check the data formatting.")

# Select top 5 most common factors
sorted_factors = factor_counts.most_common(5)
factors, counts = zip(*sorted_factors)
counts_array = np.array(counts, dtype=int)

sns.set_theme(style="whitegrid")

# Bar Diagram
plt.figure(figsize=(8, 5))
sns.barplot(x=list(counts), y=list(factors), palette="Greens_r")
plt.xlabel("Number of Students")
plt.ylabel("Factors affecting Sleep Quality")
plt.title("Bar Diagram - Factors Affecting Sleep Quality")
plt.show()
```

```
# Ogive (Cumulative Frequency Curve)
cumulative_counts = np.cumsum(counts)
plt.figure(figsize=(8, 5))
plt.plot(factors, cumulative_counts, marker='o', linestyle='-', color='green',
markerfacecolor='black')
plt.xlabel("Factors affecting Sleep Quality")
plt.ylabel("Cumulative Number of Students")
plt.title("Ogive - Cumulative Frequency of Factors Affecting Sleep Quality")
plt.xticks(rotation=30)
plt.grid(True)
plt.show()

# Histogram
plt.figure(figsize=(8, 5))
plt.hist(counts, bins=min(5, len(counts)), color="green", edgecolor="black",
alpha=0.7)
plt.xlabel("Number of Students")
plt.ylabel("Frequency")
plt.title("Histogram - Factors Affecting Sleep Quality")
plt.grid(axis="y", linestyle="--", alpha=0.7)
plt.show()

# Pie Chart
plt.figure(figsize=(7, 7))
plt.pie(counts, labels=factors, autopct="%1.1f%%",
colors=sns.color_palette("Greens", len(factors)), startangle=140)
plt.title("Pie Chart - Factors Affecting Sleep Quality")
plt.show()

# Box Plot
```

```
plt.figure(figsize=(6, 5))
sns.boxplot(x=counts_array, color="green")
plt.xlabel("Number of Students")
plt.title("Box Plot - Distribution of Factors Affecting Sleep Quality")
plt.show()
```

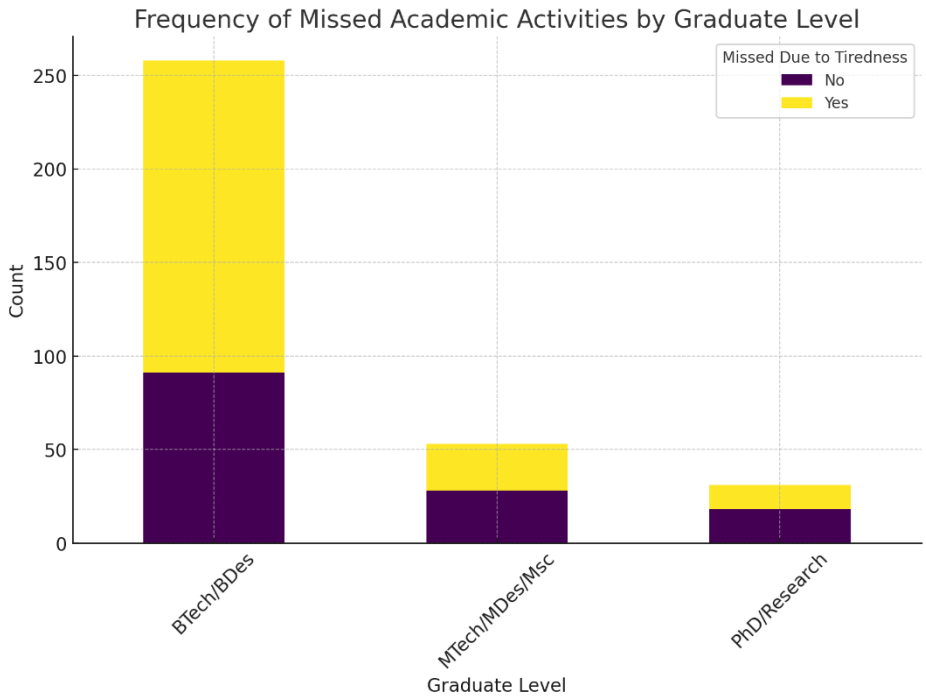
Sleep and Academics:

Impact of irregular sleep on academics:

Here we are going to visualize the frequency of people who missed a class/test/assignment with respect to their graduate level.

Graduate level of students	Frequency of people who missed a class/test/assignment		Total
	Yes	No	
BTech/BDes	167	91	258
MTech/MDes/MSc	25	28	53
PhD/Research	13	18	31
Total	205	137	342

Visual Representations:



The bar chart illustrates the frequency of missed academic activities (class, test, or assignment) across different graduate levels. Key observations:

1. **Variation by Graduate Level**: Some degree programs show higher frequencies of missed activities, possibly due to workload differences.
2. **Proportion of "Yes" Responses**: Certain graduate levels exhibit a higher proportion of students missing academic activities due to tiredness.
3. **Possible Factors**: This could be influenced by sleep patterns, academic pressure, or lifestyle choices.

   Hence there is strong evidence that most of the bachelors have missed their classes or tests or assignments.

CODE-(bar chart)

```
import pandas as pd

import matplotlib.pyplot as plt


# Load the dataset

file_path = "/mnt/data/Stats  - Sheet1.csv"

df = pd.read_csv(file_path)


# Group by graduate level and count occurrences of missed activities

missed_counts = df.groupby("What degree are you pursuing ?")["Have you ever missed a class / assignment / test due to being tired?"].value_counts().unstack()


# Create a stacked bar chart

plt.figure(figsize=(10, 6))

missed_counts.plot(kind="bar", stacked=True, colormap="viridis", ax=plt.gca())


# Customize the plot

plt.title("Frequency of Missed Academic Activities by Graduate Level")

plt.xlabel("Graduate Level")

plt.ylabel("Count")

plt.xticks(rotation=45)

plt.legend(title="Missed Due to Tiredness")

plt.tight_layout()
```

Show the plot

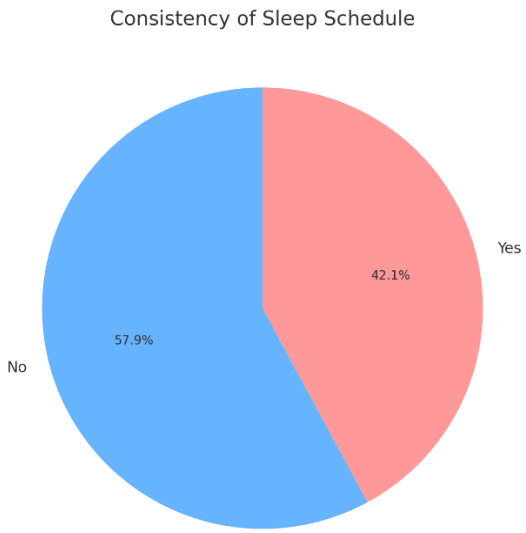
plt.show()

CONSISTENCY OF SLEEPING SCHEDULE:

Here, we wish to calculate the proportion of students who have consistent sleep schedule.

Response	Frequency
Yes	144
No	198

Visual Representation



Conclusion:mode-yes

We therefore conclude not even 50 percent of the population have a consistent sleep schedule.

CODE:-(pie chart)

import pandas as pd

import matplotlib.pyplot as plt

Load the dataset

```
file_path = "/mnt/data/Stats  - Sheet1.csv"

df = pd.read_csv(file_path)


# Group by graduate level and count occurrences of missed activities

missed_counts = df.groupby("What degree are you pursuing ?")["Have you ever missed a class / assignment / test due to being tired?"].value_counts().unstack()


# Create a pie chart for consistent sleep schedule

df["Do you have a consistent sleep schedule (go to bed and wake up at similar times each day)"].value_counts().plot(

    kind="pie", autopct="%1.1f%%", colors=["#66b3ff", "#ff9999"], figsize=(8, 8), startangle=90)


# Customize the pie chart

plt.title("Consistency of Sleep Schedule")

plt.ylabel("")

plt.show()
```

Students sleep satisfaction

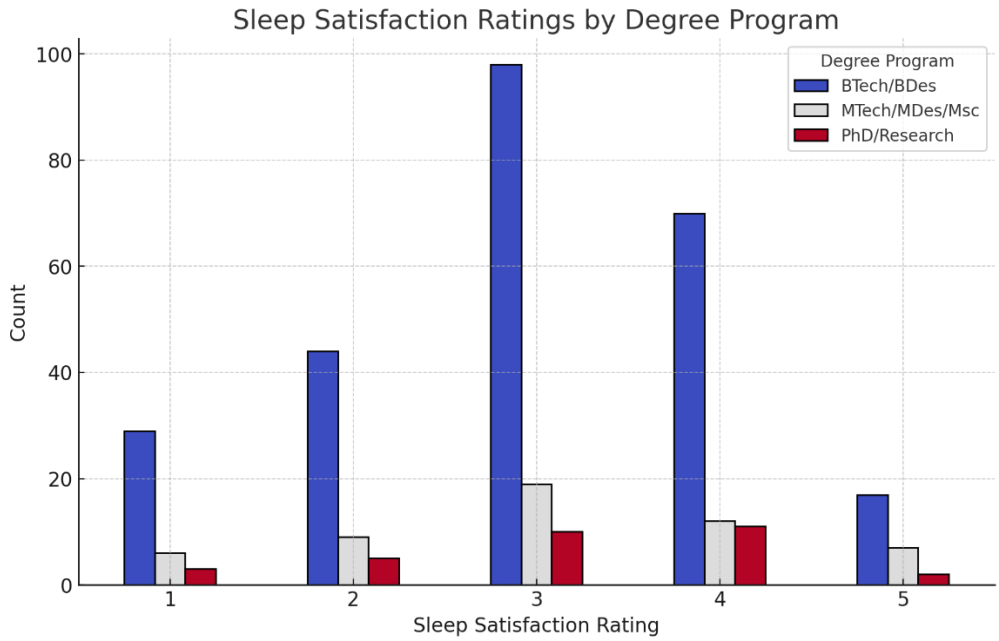
The above values are collected from a scale of numbers from 1 to 5 where 1denotes insufficient sleep and 5 denotes happy sleep. The table shows the frequency of people and their respective selected scale factor:

Scaled factor	frequency
1	38
2	58
3	127
4	93
5	26

Mode: 3 (most frequent rating)

Median: 3 (middle value in the ordered dataset)

Mean (Average): ≈ 3.03



Conclusion from the Sleep Satisfaction Chart:

1. **Majority Ratings Around 3 and 4**
    - Most respondents rated their sleep satisfaction around **3 and 4**, indicating moderate satisfaction.
    - This suggests that while students are not completely dissatisfied, there is room for improvement in sleep quality.
2. **BTech/BDes Students Dominate Responses**
    - The highest number of responses come from **BTech/BDes students**, and they are spread across all satisfaction levels.
    - This may indicate higher participation from undergraduate students in the survey.
3. **MTech/MDes/MSc and PhD Students Have Lower Counts**
    - Fewer responses from **MTech/MDes/MSc and PhD students**, but their distribution follows a similar trend.
    - PhD/Research students have slightly lower satisfaction, possibly due to higher academic workload or irregular schedules.
4. **Low Ratings (1 & 2) Are Less Frequent**
    - While some students rated their sleep satisfaction as **1 or 2**, they are in the minority.
    - This could mean that extreme dissatisfaction with sleep is not a widespread issue but still exists.

Insights:

- The overall sleep satisfaction among students is **not extremely poor**, but it is also **not excellent**.

- There are factors affecting sleep quality, such as social media, workload, stress, or lifestyle choices.
- Further analysis could explore **correlations between sleep satisfaction and study habits, workload, or mental health factors**.

CODE-barchart

```
import pandas as pd

import matplotlib.pyplot as plt


# Load the CSV file

file_path = "your_file_path_here.csv"  # Replace with actual file path

df = pd.read_csv(file_path)


# Trim whitespace from column names

df.columns = df.columns.str.strip()


# Define relevant columns

degree_column = "What degree are you pursuing ?"

sleep_satisfaction_column = "How satisfied are you with the overall quality of your sleep?"


# Filter necessary columns and drop NaNs

df_filtered = df[[degree_column, sleep_satisfaction_column]].dropna()


# Convert sleep satisfaction ratings to numeric

df_filtered[sleep_satisfaction_column] = df_filtered[sleep_satisfaction_column].astype(int)


# Count occurrences for each degree category and satisfaction rating

grouped_counts = df_filtered.groupby([degree_column,
sleep_satisfaction_column]).size().unstack(fill_value=0)


# Plot grouped bar chart

plt.figure(figsize=(10, 6))

grouped_counts.T.plot(kind="bar", figsize=(10, 6), colormap="coolwarm", edgecolor="black")


# Labels and title
```

```
plt.xlabel("Sleep Satisfaction Rating")
plt.ylabel("Count")
plt.title("Sleep Satisfaction Ratings by Degree Program")
plt.xticks(rotation=0)
plt.legend(title="Degree Program")
plt.grid(axis="y", linestyle="--", alpha=0.7)
plt.show()
```

Summary

General Sleep Trends:

- Students sleep more on weekends than weekdays.
- Sleep duration is relatively stable on weekdays but more varied on weekends.

Factors Affecting Sleep Quality:

- Social media and academic stress are the most common factors disrupting sleep.
- Other notable factors include noise, light, and peer interactions.

Impact on Academics:

- A significant number of students (especially BTech/BDes students) have missed classes, assignments, or tests due to inadequate sleep.

Sleep Satisfaction & Consistency:

- Less than 50% of students maintain a consistent sleep schedule.
- The average sleep satisfaction rating is around 3, indicating moderate satisfaction but room for improvement.

Overall Implications:

- The findings suggest a need for better sleep hygiene among students.
- Awareness programs on sleep quality and time management could help improve students' academic performance and well-being.

Introduction: The report analyses the sleep patterns and experiences of students. We conducted a survey of 342 students to get a better understanding of sleep habits and various factors affecting the sleep health. The survey investigated various factors affecting student sleep habits, including:

In this project we shall explore the following two factors:

• Students sleep satisfaction

• Have you ever missed a class / assignment / test due to being tired

Students sleep satisfaction:

1. Method of moments and maximum likelihood estimates:

a) Method of moments:

Code:

```
import numpy as np
#data
data = [4, 5, 3, 3, 3, 3, 3, 4, 1, 4, 3, 1, 4, 5, 3, 3, 4, 4, 4, 3, 2, 2, 3, 3, 3, 2, 4, 2, 2, 5, 3,
       3, 5, 3, 4, 3, 4, 1, 1, 4, 4, 3, 4, 3, 4, 3, 4, 4, 1, 1, 3, 3, 5, 3, 1, 2, 1, 4, 1, 3, 2, 2,
       3, 3, 2, 1, 4, 3, 2, 3, 3, 2, 4, 5, 3, 4, 4, 3, 4, 1, 3, 3, 3, 3, 3, 4, 4, 3, 4, 4, 3, 1, 5,
       3, 4, 3, 3, 3, 4, 3, 4, 3, 2, 3, 5, 4, 2, 3, 3, 3, 5, 2, 3, 2, 2, 1, 1, 3, 2, 2, 1, 4, 3, 4,
       1, 2, 4, 3, 3, 3, 2, 5, 1, 3, 1, 4, 3, 4, 4, 2, 4, 3, 5, 4, 2, 3, 2, 1, 3, 1, 3, 4, 4, 3, 1,
       2, 2, 4, 5, 3, 1, 3, 2, 2, 5, 2, 3, 5, 1, 4, 3, 4, 5, 3, 4, 1, 4, 4, 3, 4, 2, 1, 4, 3, 4, 3,
       3, 2, 4, 3, 2, 3, 5, 2, 3, 4, 4, 3, 4, 4, 3, 2, 5, 3, 1, 4, 1, 3, 3, 4, 4, 3, 1, 4, 3, 2, 4,
       3, 2, 5, 4, 5, 3, 2, 5, 3, 4, 4, 2, 2, 4, 3, 3, 4, 3, 1, 3, 3, 4, 2, 3, 3, 3, 3, 4, 3, 3, 2,
       5, 3, 3, 3, 1, 3, 3, 4, 3, 3, 5, 3, 3, 4, 1, 3, 3, 3, 2, 2, 1, 2, 1, 1, 3, 3, 2, 2, 4, 4, 3,
       5, 3, 3, 3, 4, 4, 3, 3, 3, 4, 3, 2, 4, 4, 3, 2, 4, 3, 4, 2, 1, 4, 5, 5, 2, 4, 1, 4, 3, 4, 2,
       3, 4, 3, 5, 4, 4, 3, 3, 2, 2, 4, 3, 4, 3, 4, 2, 2, 2, 4, 4, 4, 3, 1, 2, 1, 2, 2, 4, 3, 4, 4,
       3
      ]

data = np.array(data)
# Sample mean and variance
mean = np.mean(data)
var = np.var(data, ddof=1)
# Method of Moments estimates
a_mom = mean**2 / var
b_mom = var / mean
```

```
# Output
print("Method of Moments Estimates:->")
print(f"Sample Mean: {mean:.4f}")
print(f"Sample Variance: {var:.4f}")
print(f"MoM Estimate of a: {a_mom:.4f}")
print(f"MoM Estimate of b: {b_mom:.4f}")
```

Output:

```
Method of Moments Estimates:->
Sample Mean: 3.0322
Sample Variance: 1.1925
MoM Estimate of a:7.7098
MoM Estimate of b:0.3933
```

b) Maximum likelihood estimates:

Code:

```
import numpy as np
from scipy.special import psi  # digamma function
from scipy.optimize import fsolve
data = [4, 5, 3, 3, 3, 3, 4, 1, 4, 3, 1, 4, 5, 3, 3, 4, 4, 4, 3, 2, 2, 3, 3, 3, 2, 4, 2, 2, 5, 3,
      3, 5, 3, 4, 3, 4, 1, 1, 4, 4, 3, 4, 3, 4, 3, 4, 4, 1, 1, 3, 3, 5, 3, 1, 2, 1, 4, 1, 3, 2, 2,
      3, 3, 2, 1, 4, 3, 2, 3, 3, 2, 4, 5, 3, 4, 4, 3, 4, 1, 3, 3, 3, 3, 3, 4, 4, 3, 4, 4, 3, 1, 5,
      3, 4, 3, 3, 3, 4, 3, 4, 3, 2, 3, 5, 4, 2, 3, 3, 3, 5, 2, 3, 2, 2, 1, 1, 3, 2, 2, 1, 4, 3, 4,
      1, 2, 4, 3, 3, 3, 2, 5, 1, 3, 1, 4, 3, 4, 4, 2, 4, 3, 5, 4, 2, 3, 2, 1, 3, 1, 3, 4, 4, 3, 1,
      2, 2, 4, 5, 3, 1, 3, 2, 2, 5, 2, 3, 5, 1, 4, 3, 4, 5, 3, 4, 1, 4, 4, 3, 4, 2, 1, 4, 3, 4, 3,
      3, 2, 4, 3, 2, 3, 5, 2, 3, 4, 4, 3, 4, 4, 3, 2, 5, 3, 1, 4, 1, 3, 3, 4, 4, 3, 1, 4, 3, 2, 4,
      3, 2, 5, 4, 5, 3, 2, 5, 3, 4, 4, 2, 2, 4, 3, 3, 4, 3, 1, 3, 3, 4, 2, 3, 3, 3, 3, 4, 3, 3, 2,
      5, 3, 3, 3, 1, 3, 3, 4, 3, 3, 5, 3, 3, 4, 1, 3, 3, 3, 2, 2, 1, 2, 1, 1, 3, 3, 2, 2, 4, 4, 3,
      5, 3, 3, 3, 4, 4, 3, 3, 3, 4, 3, 2, 4, 4, 3, 2, 4, 3, 4, 2, 1, 4, 5, 5, 2, 4, 1, 4, 3, 4, 2,
      3, 4, 3, 5, 4, 4, 3, 3, 2, 2, 4, 3, 4, 3, 4, 2, 2, 2, 4, 4, 4, 3, 1, 2, 1, 2, 2, 4, 3, 4, 4,
      3
]
data = np.array(data)
#sample mean,log of sample mean and mean of logs
x_bar = np.mean(data)
log_x_bar = np.log(x_bar)
mean_log_x = np.mean(np.log(data))

#MLE equation: ψ(a) - log(a) = mean_log_x - log(x̄)
rhs = mean_log_x - log_x_bar

#function to solve for a
def mle_equation(a):
   return psi(a) - np.log(a) - rhs
```

```
#Initial guess for a
initial_guess = 1.0

#Use numerical solver to solve for a
a_mle = fsolve(mle_equation, initial_guess)[0]

#Compute b using b = x̄ / a
b_mle = x_bar / a_mle

#output
print("Maximum Likelihood Estimation(Gamma):->")
print(f"Sample mean (x̄): {x_bar:.4f}")
print(f"Mean of log(x): {mean_log_x:.4f}")
print(f"MLE Estimate for shape (a): {a_mle:.4f}")
print(f"MLE Estimate for scale (b): {b_mle:.4f}")
```

Output:

```
Maximum Likelihood Estimation(Gamma):->
Sample mean (x̄): 3.0322
Mean of log(x): 1.0248
MLE Estimate for shape (a): 6.0838
MLE Estimate for scale (b):0.4984
```

Conclusion:

The MLE fit indicates a slightly lower shape and higher rate, meaning the MLE estimates a distribution more concentrated around the mean, which may better reflect the central tendency and skewness of the actual data. Since MLE is generally more robust, we would prefer the MLE estimates for further modeling.

2. 95% confidence interval for variance:

Code:

```
import numpy as np
from scipy.stats import chi2

#data
data = [4, 5, 3, 3, 3, 3, 3, 4, 1, 4, 3, 1, 4, 5, 3, 3, 4, 4, 4, 3, 2, 2, 3, 3, 3, 2, 4, 2, 2, 5, 3,
        3, 5, 3, 4, 3, 4, 1, 1, 4, 4, 3, 4, 3, 4, 3, 4, 4, 1, 1, 3, 3, 5, 3, 1, 2, 1, 4, 1, 3, 2, 2,
        3, 3, 2, 1, 4, 3, 2, 3, 3, 2, 4, 5, 3, 4, 4, 3, 4, 1, 3, 3, 3, 3, 3, 4, 4, 3, 4, 4, 3, 1, 5,
        3, 4, 3, 3, 3, 4, 3, 4, 3, 2, 3, 5, 4, 2, 3, 3, 3, 5, 2, 3, 2, 2, 1, 1, 3, 2, 2, 1, 4, 3, 4,
        1, 2, 4, 3, 3, 3, 2, 5, 1, 3, 1, 4, 3, 4, 4, 2, 4, 3, 5, 4, 2, 3, 2, 1, 3, 1, 3, 4, 4, 3, 1,
        2, 2, 4, 5, 3, 1, 3, 2, 2, 5, 2, 3, 5, 1, 4, 3, 4, 5, 3, 4, 1, 4, 4, 3, 4, 2, 1, 4, 3, 4, 3,
        3, 2, 4, 3, 2, 3, 5, 2, 3, 4, 4, 3, 4, 4, 3, 2, 5, 3, 1, 4, 1, 3, 3, 4, 4, 3, 1, 4, 3, 2, 4,
        3, 2, 5, 4, 5, 3, 2, 5, 3, 4, 4, 2, 2, 4, 3, 3, 4, 3, 1, 3, 3, 4, 2, 3, 3, 3, 3, 4, 3, 3, 2,
        5, 3, 3, 3, 1, 3, 3, 4, 3, 3, 5, 3, 3, 4, 1, 3, 3, 3, 2, 2, 1, 2, 1, 1, 3, 3, 2, 2, 4, 4, 3,
        5, 3, 3, 3, 4, 4, 3, 3, 3, 4, 3, 2, 4, 4, 3, 2, 4, 3, 4, 2, 1, 4, 5, 5, 2, 4, 1, 4, 3, 4, 2,
```

```
    3, 4, 3, 5, 4, 4, 3, 3, 2, 2, 4, 3, 4, 3, 4, 2, 2, 2, 4, 4, 4, 3, 1, 2, 1, 2, 2, 4, 3, 4, 4,
    3
]
data = np.array(data)

#size and sample variance
n = len(data)
s2 = np.var(data, ddof=1)

alpha = 0.05
df = n - 1

chi2_lower = chi2.ppf(alpha / 2, df)     # lower critical value
chi2_upper = chi2.ppf(1 - alpha / 2, df) # upper critical value

#confidence interval for Variance
lower_bound = (df * s2) / chi2_upper
upper_bound = (df * s2) / chi2_lower

#output
print("95% Confidence Interval for Variance:->")
print(f"Sample Variance(s²): {s2:.4f}")
print(f"Confidence Interval: ({lower_bound:.4f}, {upper_bound:.4f})")
```

Output:

95% Confidence Interval for Variance:->
Sample Variance(s^2):1.1925
Confidence Interval:(1.0320,1.3939)

Conclusion:

This range tells us that the variability in sleep satisfaction is moderate. Since the entire interval is above 1, it suggests that students' satisfaction ratings are not tightly clustered around the mean — there's noticeable spread, possibly due to differences in lifestyle, stress, or sleep duration.

3. 95% confidence interval for difference in means:

Code:

```
 import numpy as np
from scipy.stats import t

data1 = [4,5,3,3,3,4,5,3,2,3,4,3,4,4,3,3,1,1,5,2,1,2,3,2,1,2,4,1,3,
        3,4,4,5,3,3,3,4,3,4,3,4,2,2,1,3,2,2,1,3,4,2,4,3,1,4,3,4,4,
        2,4,3,4,3,1,2,3,3,2,5,4,3,4,4,3,4,3,3,4,4,4,3,2,5,3,1,4,3,
        3,4,4,3,4,4,3,5,2,4,3,3,3,4,3,3,2,3,1,3,3,4,3,3,5,3,3,2,2,
        2,3,2,2,3,5,4,3,3,1,1,3,2,3,4,3,2,3,4,2,1,3] #female
```

```
data2 = [3,3,3,4,1,4,1,3,3,4,4,4,3,2,2,3,3,4,2,2,5,3,3,5,1,1,4,3,4,
      4,4,4,3,3,3,1,1,4,3,2,3,4,3,3,3,2,4,5,3,4,3,4,3,3,3,4,3,4,
      3,1,4,3,2,3,5,2,3,3,3,5,3,2,1,4,1,3,3,2,5,1,3,5,2,2,1,3,3,
      4,4,3,1,2,4,5,1,2,2,3,5,1,4,3,5,1,4,4,2,1,4,3,3,2,4,3,2,5,
      2,3,4,3,1,1,3,2,2,5,4,3,5,3,4,2,2,4,3,4,3,1,3,4,2,3,3,3,5,
      3,3,3,4,1,3,3,1,1,1,3,4,4,3,3,3,4,4,3,3,3,2,4,4,3,2,4,4,2,
      4,5,5,2,4,4,4,3,4,5,4,3,2,4,3,4,2,2,4,4,4,3,1,2,2,2,4,4,4,
      3] #male

data1 = np.array(data1)
data2 = np.array(data2)

#Calculate sample means and variances
mean1 = np.mean(data1)
mean2 = np.mean(data2)
var1 = np.var(data1, ddof=1)
var2 = np.var(data2, ddof=1)
n1 = len(data1)
n2 = len(data2)

#standard error
se = np.sqrt(var1 / n1 + var2 / n2)

#degrees of freedom
df = (var1 / n1 + var2 / n2)*2 / ((var12) / (n12 * (n1 - 1)) + (var22) / (n2*2 * (n2 - 1)))


t_crit = t.ppf(0.975, df)  # 95% CI


#Compute confidence interval
diff = mean1 - mean2
lower = diff - t_crit * se
upper = diff + t_crit * se

#output
print("95% Confidence Interval for Difference in Means")
print(f"Mean of data1: {mean1:.4f}")
print(f"Mean of data2: {mean2:.4f}")
print(f"Difference(data1-data2): {diff:.4f}")
print(f"Confidence Interval: ({lower:.4f}, {upper:.4f})")
```

Output:

```
95% Confidence Interval for Difference in Means
Mean of data1:3.0145
```

Mean of data2: 3.0441
Difference(data1-data2): -0.0296
Confidence Interval: (-0.2640,0.2047)

Conclusion:
The small difference in mean sleep satisfaction between male and female students (−0.0296) is not statistically significant, as the 95% confidence interval includes zero.

Have you ever missed a class / assignment / test due to being tired?

4. Comprehensively test the hypothesis at a level of significance of 0.05.

Code-

```
 import math
from scipy.stats import norm


# Survey data
x = 222          # Number of "Yes" responses (successes)
n = 342        # Total number of responses
p0 = 0.5        # Null hypothesis proportion


# Sample proportion
p_hat = x / n


# Standard error under H0
se = math.sqrt(p0 * (1 - p0) / n)


# Z-statistic
z = (p_hat - p0) / se


# p-value for one-tailed test (H1: p > 0.5)
p_value = 1 - norm.cdf(z)


# Output results
print(f"Sample Proportion (p̂): {p_hat:.4f}")
print(f"Z-statistic: {z:.4f}")
```

```
print(f"P-value: {p_value:.4f}")


# Conclusion
alpha = 0.05
if p_value < alpha:
    print("Reject the null hypothesis (H0). There is significant evidence that p > 0.5.")
else:
    print("Fail to reject the null hypothesis (H0). Not enough evidence that p > 0.5.")
```

Output:

Z-statistic: 5.5155

P-value: 0.0000

Reject the null hypothesis (H0). There is significant evidence that p > 0.5.

Conclusion:

There is strong statistical evidence that more than half of the students have missed a class/test due to being tired.This highlights the potential academic consequences of poor sleep patterns among college students, emphasizing the need for better sleep management.