MENTAL HEALTH AND SOCIALMEDIA DATA ANALYSIS

Social media is commonly defined as a collective term for websites and applications that focus on communication, interaction, content-sharing, and collaboration (Rouse, 2020). In recent years there has been a rapid rise in the way communication, connection and sharing information happens on social media. Various platforms have been created and are being used for various purposes, the primary ones being Facebook, Twitter, and Instagram. This digital transformation has also led to concerns about its potential impact on mental health. There is a relationship between social media and the well-being of an individual. There are both positive and negative effects of social media on well-being.

DATASET https://www.kaggle.com/code/syedanwarafridi/mental-health-trends-in-the-age-of-social-media/input

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
df=pd.read_csv("/content/socialmedia&mentalhelthdataset.csv")
```

```
df.head()
```

	Timestamp	1. What is your age?	2. Gender	3. Relationship Status	4. Occupation Status	5. What type of organizations are you affiliated with?	6. Do you use social media?
0	4/18/22 19:18	21.0	Male	In a relationship	University Student	University	Yes
1	4/18/22 19:19	21.0	Female	Single	University Student	University	Yes
2	4/18/22 19:25	21.0	Female	Single	University Student	University	Yes
3	4/18/22 19:29	21.0	Female	Single	University Student	University	Yes
4	4/18/22 19:33	21.0	Female	Single	University Student	University	Yes

5 rows × 21 columns

```
df.shape
```

```
(481, 21)
```

```
df.columns
```

```
Index(['Timestamp', '1. What is your age?', '2. Gender',
       '3. Relationship Status', '4. Occupation Status',
       '5. What type of organizations are you affiliated with?',
       '6. Do you use social media?',
       '7. What social media platforms do you commonly use?',
       '8. What is the average time you spend on social media every day?',
       '9. How often do you find yourself using Social media without a specific purpose?',
       '10. How often do you get distracted by Social media when you are busy doing something?',
       '11. Do you feel restless if you haven't used Social media in a while?',
       '12. On a scale of 1 to 5, how easily distracted are you?',
       '13. On a scale of 1 to 5, how much are you bothered by worries?',
       '14. Do you find it difficult to concentrate on things?',
       '15. On a scale of 1-5, how often do you compare yourself to other successful people through the
use of social media?',
       '16. Following the previous question, how do you feel about these comparisons, generally
speaking?',
       '17. How often do you look to seek validation from features of social media?',
       '18. How often do you feel depressed or down?',
       '19. On a scale of 1 to 5, how frequently does your interest in daily activities fluctuate?',
       '20. On a scale of 1 to 5, how often do you face issues regarding sleep?'],
      dtype='object')
```

Renaming Columns

```
new_column_names = {
    'Timestamp': 'timestamp',
    '1. What is your age?': 'age',
    '2. Gender': 'gender',
    '3. Relationship Status': 'relationship_status',
    '4. Occupation Status': 'occupation_status',
    '5. What type of organizations are you affiliated with?': 'affiliated_organizations',
    '6. Do you use social media?': 'use_social_media',
    '7. What social media platforms do you commonly use?': 'social_media_platforms',
    '8. What is the average time you spend on social media every day?': 'daily_social_media_time',
    '9. How often do you find yourself using Social media without a specific purpose?': 'frequency_social_me
    '10. How often do you get distracted by Social media when you are busy doing something?': 'frequency_soc
    "11. Do you feel restless if you haven't used Social media in a while?": 'restless_without_social_media'
    '12. On a scale of 1 to 5, how easily distracted are you?': 'distractibility_scale',
    '13. On a scale of 1 to 5, how much are you bothered by worries?': 'worry_level_scale',
    '14. Do you find it difficult to concentrate on things?': 'difficulty_concentrating',
    '15. On a scale of 1-5, how often do you compare yourself to other successful people through the use of
    '16. Following the previous question, how do you feel about these comparisons, generally speaking?': 'fe
    '17. How often do you look to seek validation from features of social media?': 'frequency_seeking_valida
    '18. How often do you feel depressed or down?': 'frequency_feeling_depressed',
    '19. On a scale of 1 to 5, how frequently does your interest in daily activities fluctuate?': 'interest_
    '20. On a scale of 1 to 5, how often do you face issues regarding sleep?': 'sleep_issues_scale',
}
df=df.rename(columns=new_column_names)
```

```
df
```

	timestamp	age	gender	relationship_status	occupation_status	affiliated_organizations	use_social_
0	4/18/22 19:18	21.0	Male	In a relationship	University Student	University	
1	4/18/22 19:19	21.0	Female	Single	University Student	University	
2	4/18/22 19:25	21.0	Female	Single	University Student	University	
3	4/18/22 19:29	21.0	Female	Single	University Student	University	
4	4/18/22 19:33	21.0	Female	Single	University Student	University	
...	...	...	...	...	...	...	
476	5/21/22 23:38	24.0	Male	Single	Salaried Worker	University, Private	
477	5/22/22 0:01	26.0	Female	Married	Salaried Worker	University	
478	5/22/22 10:29	29.0	Female	Married	Salaried Worker	University	
479	7/14/22 19:33	21.0	Male	Single	University Student	University	
480	11/12/22 13:16	53.0	Male	Married	Salaried Worker	Private	

481 rows × 21 columns

```
df.tail()
```

	timestamp	age	gender	relationship_status	occupation_status	affiliated_organizations	use_social_
476	5/21/22 23:38	24.0	Male	Single	Salaried Worker	University, Private	
477	5/22/22 0:01	26.0	Female	Married	Salaried Worker	University	
478	5/22/22 10:29	29.0	Female	Married	Salaried Worker	University	
479	7/14/22 19:33	21.0	Male	Single	University Student	University	
480	11/12/22 13:16	53.0	Male	Married	Salaried Worker	Private	

5 rows × 21 columns

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 481 entries, 0 to 480
Data columns (total 21 columns):
 #   Column                                Non-Null Count  Dtype
---  ------                                --------------  -----
 0   timestamp                             481 non-null    object
 1   age                                   481 non-null    float64
 2   gender                                481 non-null    object
 3   relationship_status                   481 non-null    object
 4   occupation_status                     481 non-null    object
 5   affiliated_organizations              451 non-null    object
 6   use_social_media                      481 non-null    object
 7   social_media_platforms                481 non-null    object
 8   daily_social_media_time               481 non-null    object
 9   frequency_social_media_no_purpose     481 non-null    int64
 10  frequency_social_media_distracted     481 non-null    int64
 11  restless_without_social_media         481 non-null    int64
 12  distractibility_scale              481 non-null    int64
 13  worry_level_scale                  481 non-null    int64
 14  difficulty_concentrating           481 non-null    int64
 15  compare_to_successful_people_scale 481 non-null    int64
 16  feelings_about_comparisons         481 non-null    int64
 17  frequency_seeking_validation       481 non-null    int64
 18  frequency_feeling_depressed        481 non-null    int64
 19  interest_fluctuation_scale         481 non-null    int64
 20  sleep_issues_scale                 481 non-null    int64
dtypes: float64(1), int64(12), object(8)
memory usage: 79.0+ KB
```

```
df.isnull().sum()
```

	0
timestamp	0
age	0
gender	0
relationship_status	0
occupation_status	0
affiliated_organizations	30
use_social_media	0
social_media_platforms	0
daily_social_media_time	0
frequency_social_media_no_purpose	0
frequency_social_media_distracted	0
restless_without_social_media	0
distractibility_scale	0
worry_level_scale	0
difficulty_concentrating	0
compare_to_successful_people_scale	0
feelings_about_comparisons	0
frequency_seeking_validation	0
frequency_feeling_depressed	0
interest_fluctuation_scale	0
sleep_issues_scale	0

dtype: int64

Imputting The Null Values

```
import pandas as pd
from scipy.stats import mode
```

```
import pandas as pd
from scipy.stats import mode

# Access the 'affiliated_organizations' column within the DataFrame
mode_value = df['affiliated_organizations'].mode()[0]

# Fill NaN values in the column with the mode value
df['affiliated_organizations'].fillna(mode_value, inplace=True)
```

```
<ipython-input-172-0176a3b033a8>:8: FutureWarning: A value is trying to be set on a copy of a DataFrame
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate obj
```

```
    For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplac

      df['affiliated_organizations'].fillna(mode_value, inplace=True)
```

```
df.isnull().sum()
```

```
                                       0
                        timestamp      0
                              age      0
                           gender      0
              relationship_status      0
                occupation_status      0
         affiliated_organizations      0
                 use_social_media      0
           social_media_platforms      0
          daily_social_media_time      0
frequency_social_media_no_purpose      0
 frequency_social_media_distracted     0
     restless_without_social_media     0
             distractibility_scale     0
                worry_level_scale      0
          difficulty_concentrating     0
compare_to_successful_people_scale     0
          feelings_about_comparisons   0
        frequency_seeking_validation   0
         frequency_feeling_depressed   0
           interest_fluctuation_scale  0
                 sleep_issues_scale    0

dtype: int64
```

so now our data have no null values

```
duplicate_rows=df[df.duplicated()]
```

```
duplicate_rows.sum()
```

```
                                              0
                    timestamp                 0
                       age                  0.0
                     gender                   0
               relationship_status            0
                occupation_status             0
             affiliated_organizations         0
                 use_social_media             0
              social_media_platforms          0
             daily_social_media_time          0
      frequency_social_media_no_purpose       0
       frequency_social_media_distracted      0
         restless_without_social_media        0
              distractibility_scale           0
                worry_level_scale             0
             difficulty_concentrating         0
      compare_to_successful_people_scale      0
           feelings_about_comparisons         0
          frequency_seeking_validation        0
           frequency_feeling_depressed        0
            interest_fluctuation_scale        0
               sleep_issues_scale             0

dtype: object
```

```
df.columns
```

```
Index(['timestamp', 'age', 'gender', 'relationship_status',
       'occupation_status', 'affiliated_organizations', 'use_social_media',
       'social_media_platforms', 'daily_social_media_time',
       'frequency_social_media_no_purpose',
       'frequency_social_media_distracted', 'restless_without_social_media',
       'distractibility_scale', 'worry_level_scale',
       'difficulty_concentrating', 'compare_to_successful_people_scale',
       'feelings_about_comparisons', 'frequency_seeking_validation',
       'frequency_feeling_depressed', 'interest_fluctuation_scale',
       'sleep_issues_scale'],
      dtype='object')
```

Descriptive Analysis

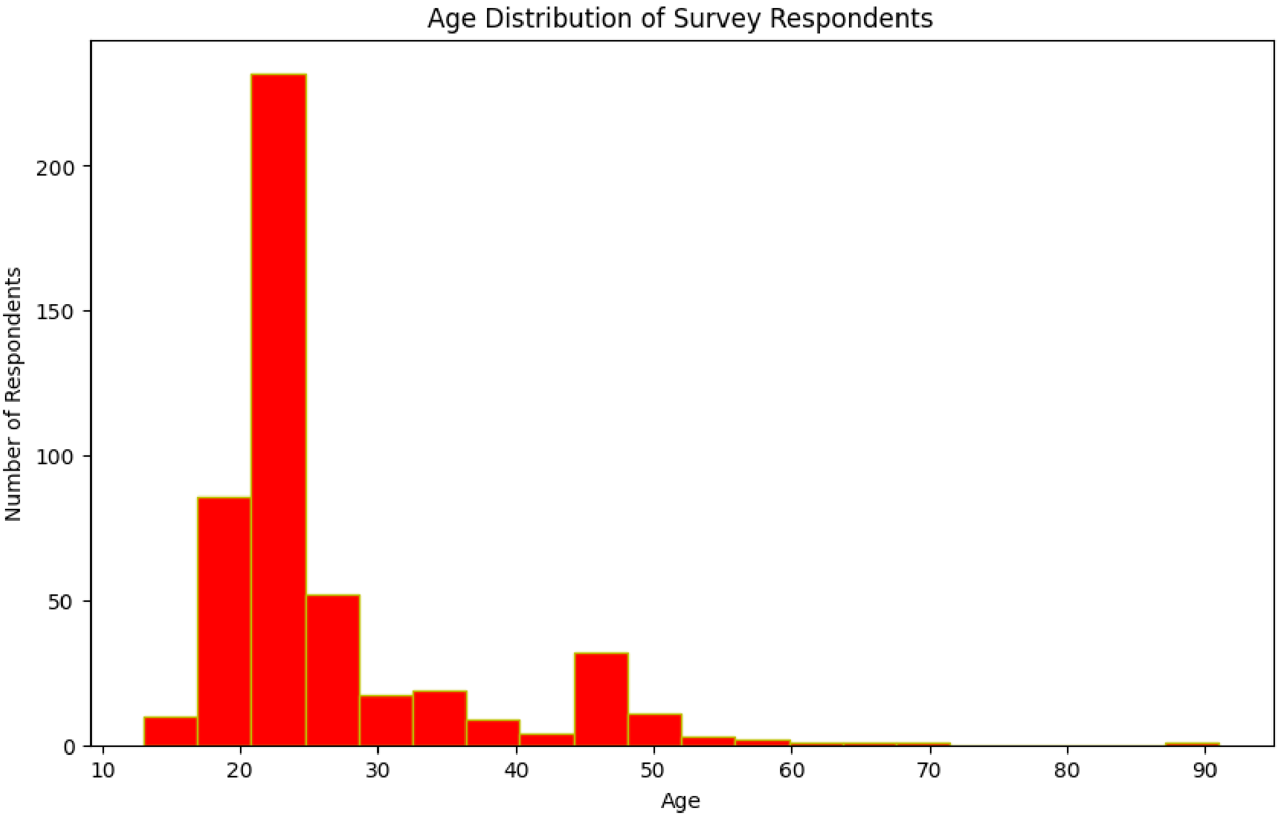
1. What is the average age of the survey respondents?
2. How is the gender distribution among the respondents?
3. What are the most common relationship statuses and occupation statuses among the respondents?
4. Which social media platforms are the most commonly used among the survey participants?
5. What is the distribution of daily social media usage times among the respondents?
6. How often do respondents find themselves using social media without a specific purpose, and how does this vary by age and gender?

what is the average age of the survey respondents and what is the distribution of age in the dataset

```
average_age=df['age'].mean()
print(f"The average age of survey respondent is {average_age}")
```

```
The average age of survey respondent is 26.13659043659044
```

```
plt.figure(figsize=(10, 6))
plt.hist(df['age'], bins=20,color='red',edgecolor='y')
plt.title('Age Distribution of Survey Respondents')
plt.xlabel('Age')
plt.ylabel('Number of Respondents')
plt.show()
```



INTERPRETATION

Mainly people belongs to 20-26 Age spent lots of time on social media platforms.

How is the gender distribution among the respondents

```
gender_distribution=df['gender'].value_counts()
print(gender_distribution)
```

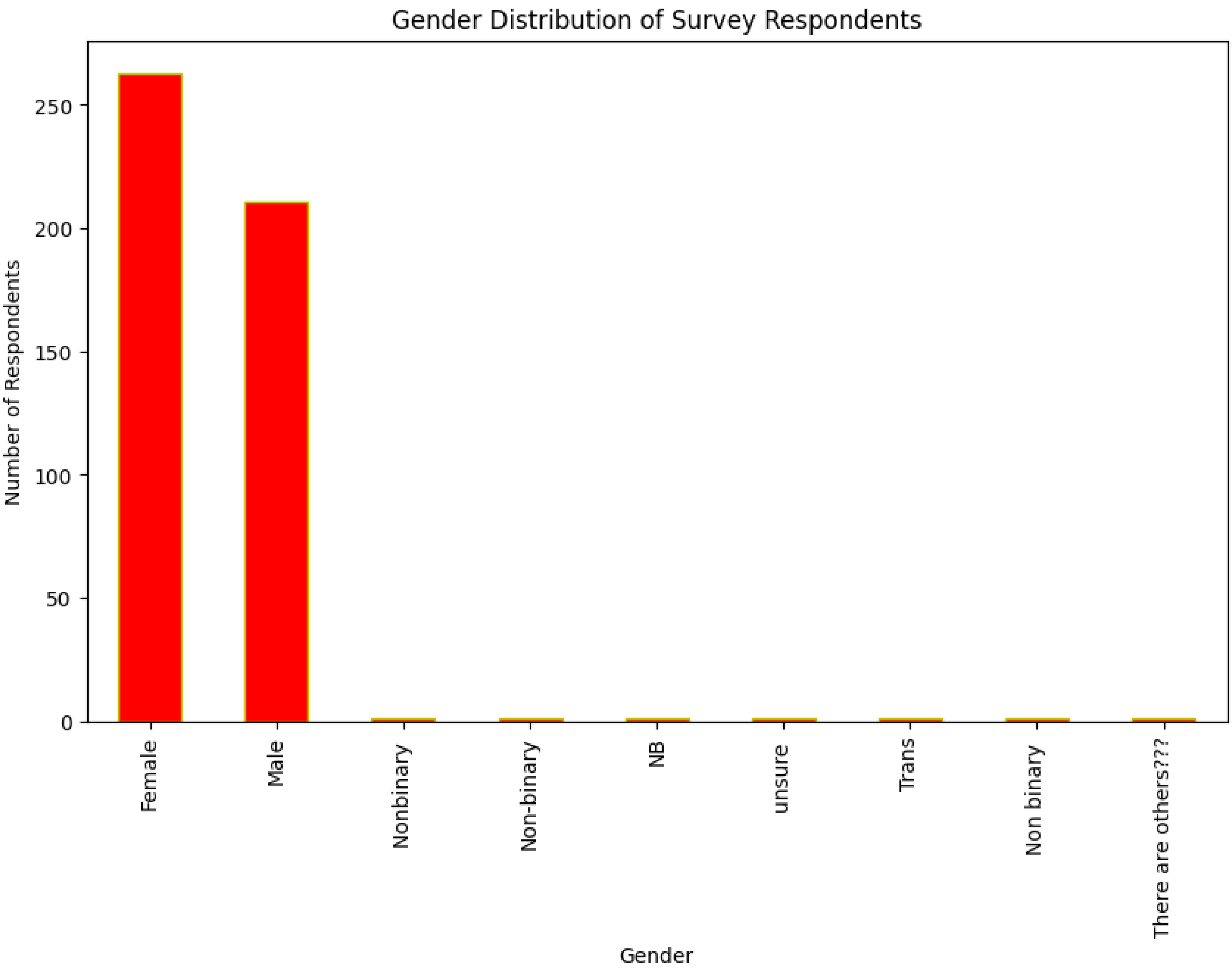
```
gender
Female                   263
Male                     211
Nonbinary                  1
Non-binary                 1
NB                         1
unsure                     1
Trans                      1
Non binary                 1
There are others???        1
Name: count, dtype: int64
```

INTERPRETATION

female category use social media more time compared to others.

```
plt.figure(figsize=(10,6))
gender_distribution.plot(kind='bar',color='red',edgecolor='y')
```

```
plt.title('Gender Distribution of Survey Respondents')
plt.xlabel('Gender')
plt.ylabel('Number of Respondents')
plt.show()
```

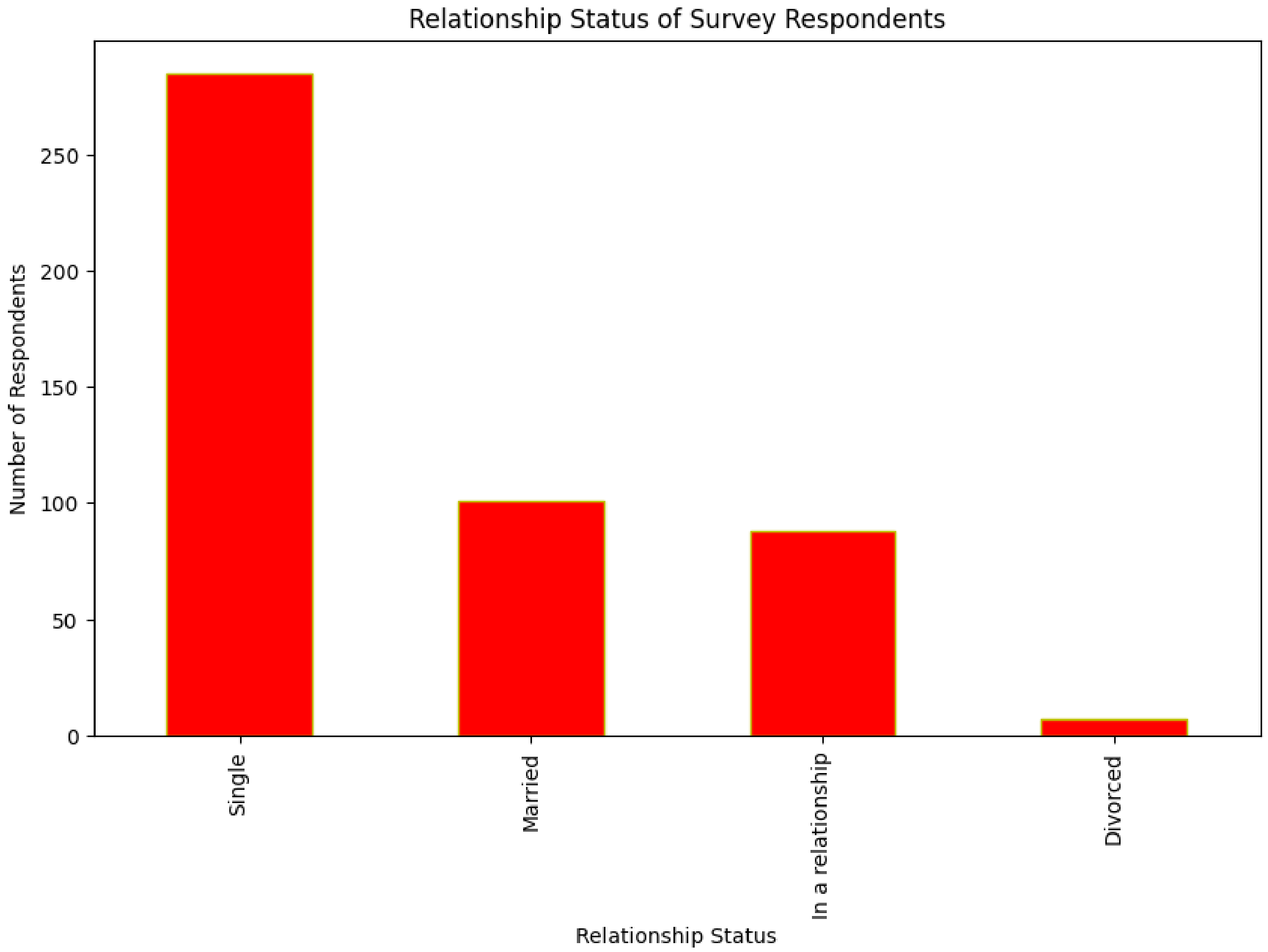


What are the common relationship statuses and occupation statuses among the respondents

```
relationship_status_counts=df['relationship_status'].value_counts()
print(relationship_status_counts)
```

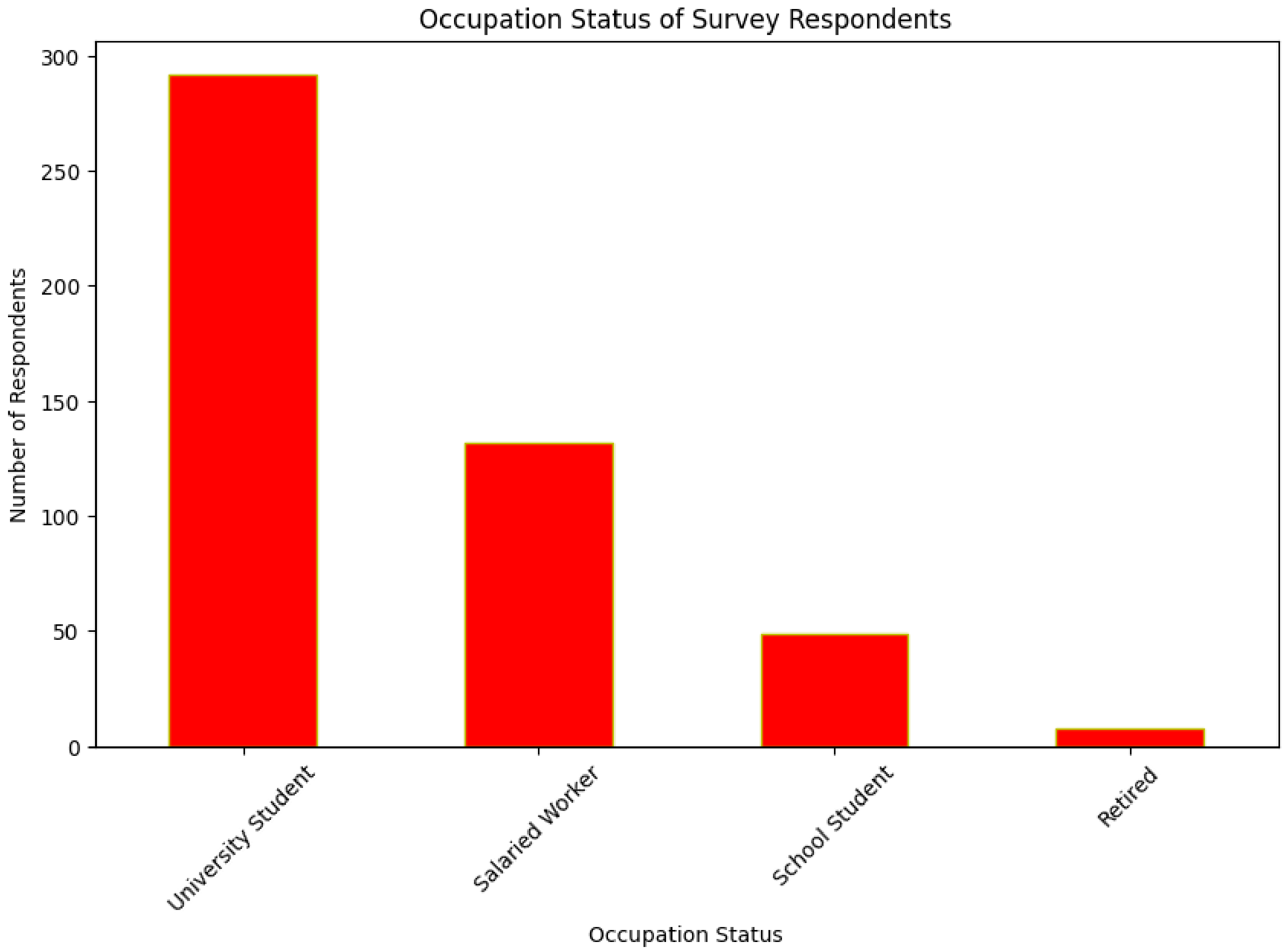
```
relationship_status
Single               285
Married              101
In a relationship     88
Divorced               7
Name: count, dtype: int64
```

```
plt.figure(figsize=(10,6))
relationship_status_counts.plot(kind='bar',color='red',edgecolor='y')
plt.title('Relationship Status of Survey Respondents')
plt.xlabel('Relationship Status')
plt.ylabel('Number of Respondents')
plt.show()
```



```
occupation_statuses = df['occupation_status'].value_counts()

plt.figure(figsize=(10, 6))
occupation_statuses.plot(kind='bar',color='red',edgecolor='y')
plt.title('Occupation Status of Survey Respondents')
plt.xlabel('Occupation Status')
plt.ylabel('Number of Respondents')
plt.xticks(rotation=45)
plt.show()
```



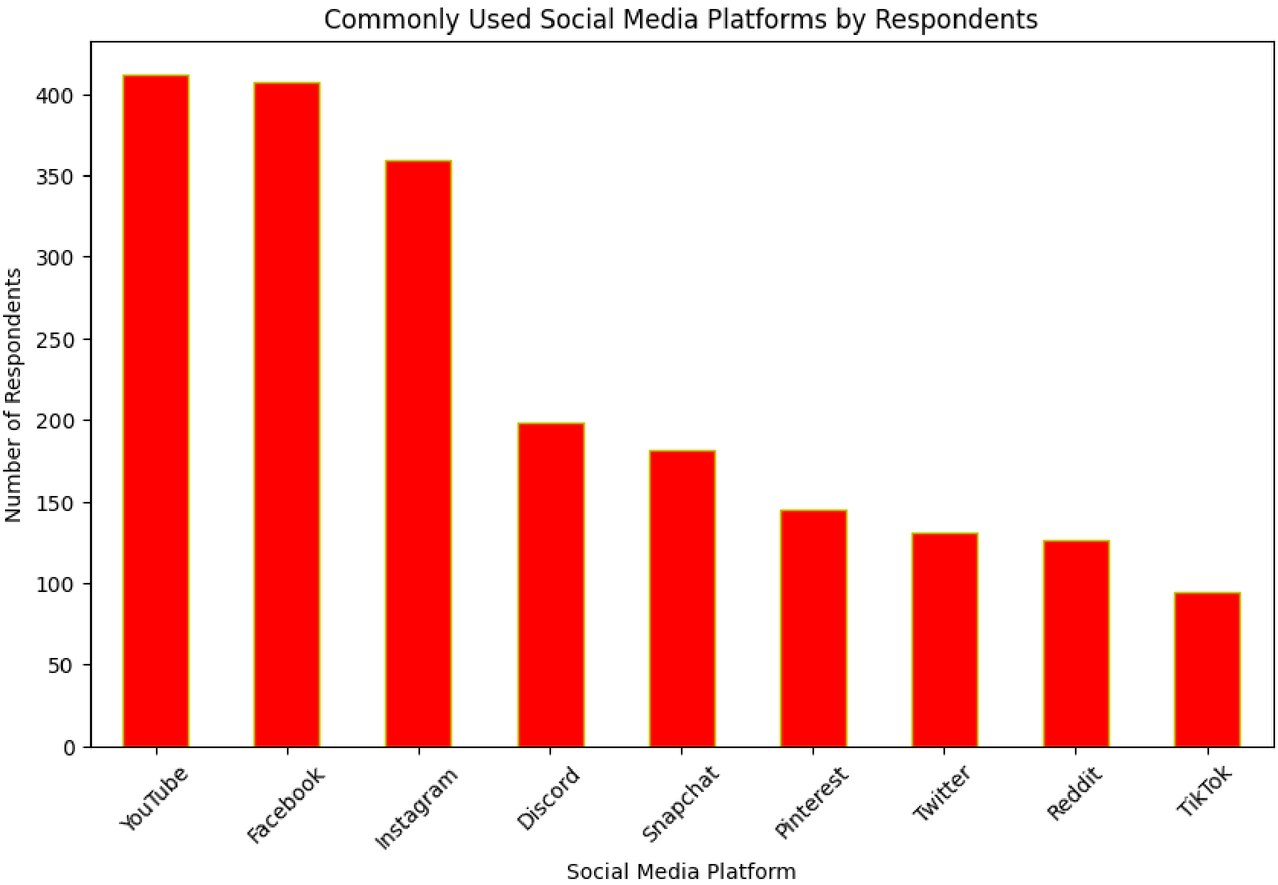
INTERPRETATION

University student use social media compared to salaried worker and others.Retired persons spent less time on social media platforms.

Which Socialmedia platforms are the most commonly used among the survey participents

```
social_media_platforms = df['social_media_platforms'].str.split(', ', expand=True).stack().value_counts()

plt.figure(figsize=(10, 6))
social_media_platforms.plot(kind='bar',color='red',edgecolor='y')
plt.title('Commonly Used Social Media Platforms by Respondents')
plt.xlabel('Social Media Platform')
plt.ylabel('Number of Respondents')
plt.xticks(rotation=45)
plt.show()
```



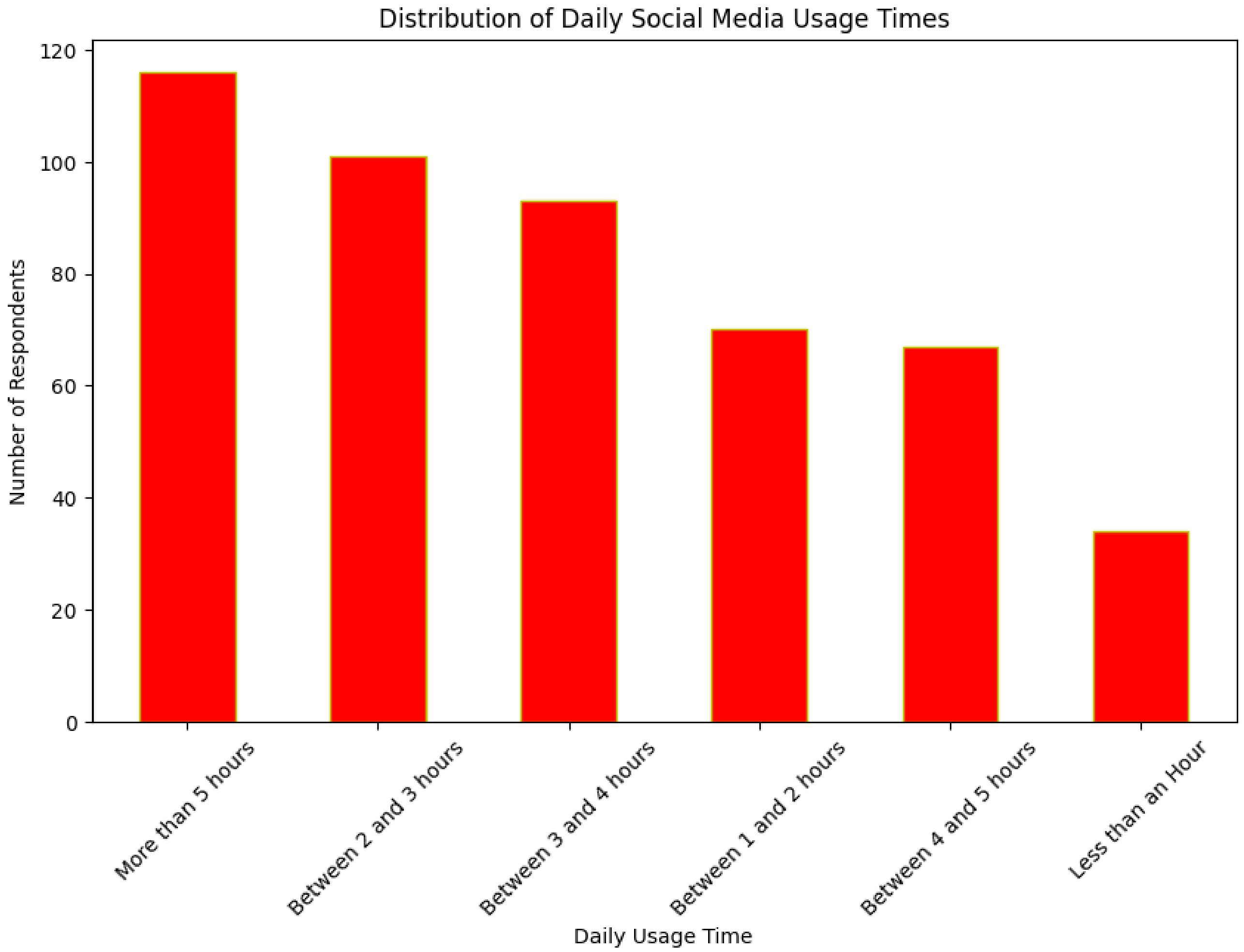
INTERPRETATION

people use youtube more compared to others.

What is the distribution of Daily socialmedia usage time among the respondents

```
daily_usage_distribution = df['daily_social_media_time'].value_counts()

plt.figure(figsize=(10, 6))
daily_usage_distribution.plot(kind='bar',color='red',edgecolor='y')
plt.title('Distribution of Daily Social Media Usage Times')
plt.xlabel('Daily Usage Time')
plt.ylabel('Number of Respondents')
plt.xticks(rotation=45)
plt.show()
```

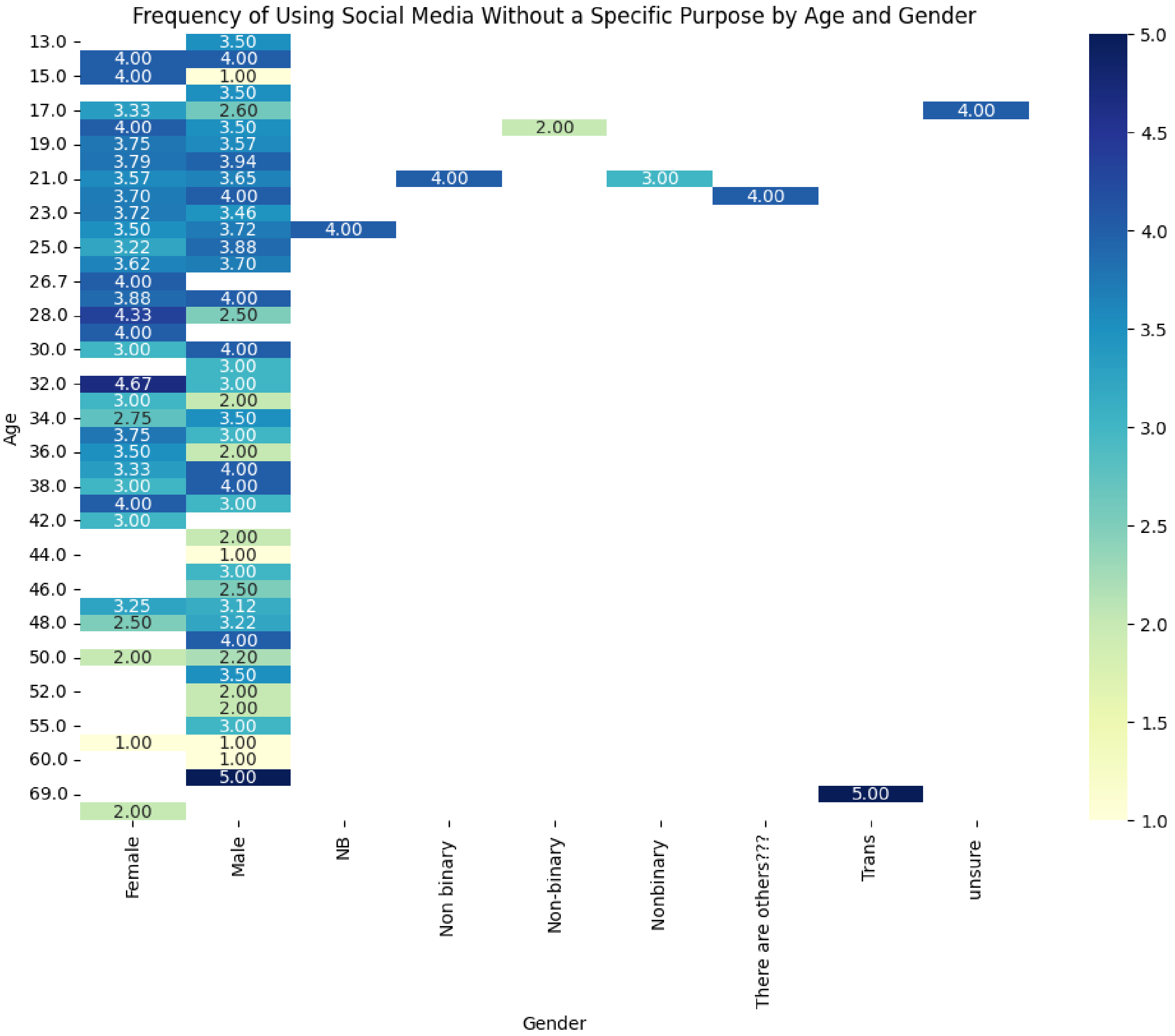


How often do respondents find themselves using social media without a specific purpose, and how does this vary by age and gender?

```
freq_by_age_gender = df.groupby(['age', 'gender'])['frequency_social_media_no_purpose'].mean().reset_index()

pivot_table = freq_by_age_gender.pivot_table(index='age', columns='gender', values='frequency_social_media_n

plt.figure(figsize=(12, 8))
sns.heatmap(pivot_table, cmap='YlGnBu', annot=True, fmt=".2f")
plt.title('Frequency of Using Social Media Without a Specific Purpose by Age and Gender')
plt.xlabel('Gender')
plt.ylabel('Age')
plt.show()
```



INTERPRETATION

Transgenders belongs to the age 69 use social media without any purpose maximum ,Female belongs to 32 Age and Male beongs to the Age 60-69,spent maximum time on social media without any specific purpose.

Exploratory Data Analysis

Is there a correlation between the time spent on social media and feelings of restlessness when not using it?

```
# Convert 'daily_social_media_time' to string and extract numeric values
df['daily_social_media_time'] = df['daily_social_media_time'].astype(str)
df['daily_social_media_time'] = df['daily_social_media_time'].str.extract('(\d+)')

# Convert extracted values to numeric, handling errors by coercing to NaN
df['daily_social_media_time'] = pd.to_numeric(df['daily_social_media_time'], errors='coerce')
 # Calculate correlation
correlation = df['daily_social_media_time'].corr(df['restless_without_social_media'])
print(f"Correlation between Time Spent on Social Media and Feelings of Restlessness: {correlation:.2f}")
```

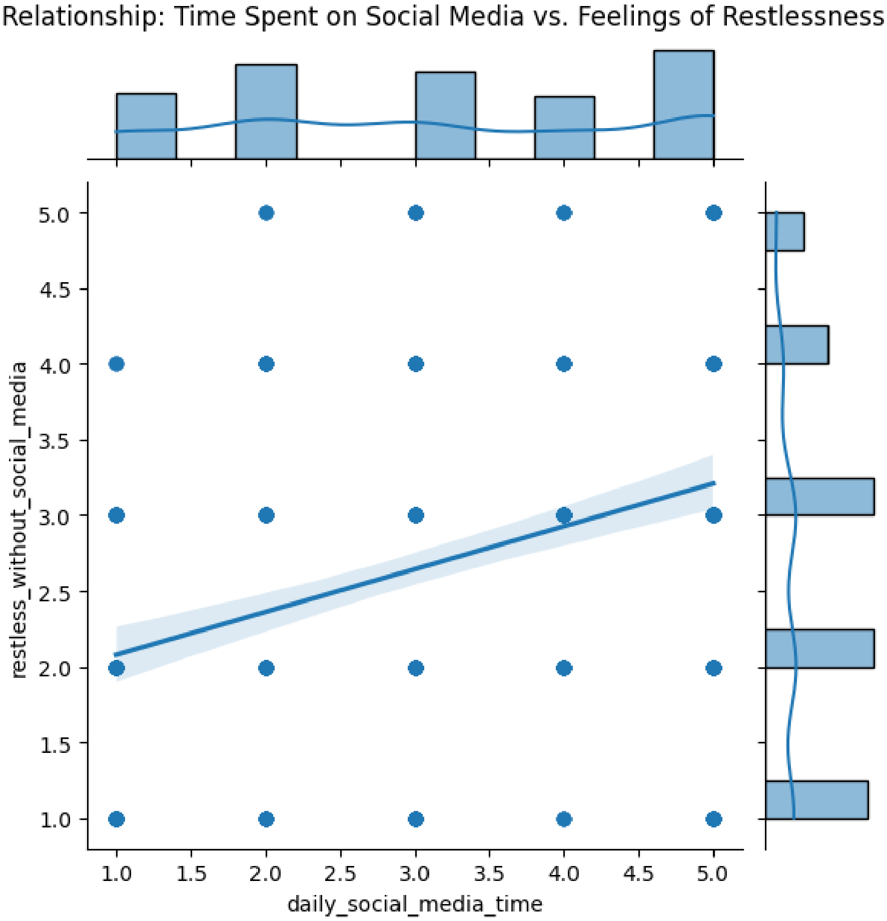
```
Correlation between Time Spent on Social Media and Feelings of Restlessness: 0.32
```

INTERPRETATION

Correlation between Daily use of social media and Feelings of Restlessness Without using is positive Correlation.That means when Daily usage time increases The Feelings of Restlessness without using it also increases and viceversa.

```
plt.figure(figsize=(10, 6))
sns.jointplot(x='daily_social_media_time', y='restless_without_social_media', data=df, kind='reg')
plt.suptitle('Relationship: Time Spent on Social Media vs. Feelings of Restlessness', y=1.02)
plt.show()
```

```
<Figure size 1000x600 with 0 Axes>
```



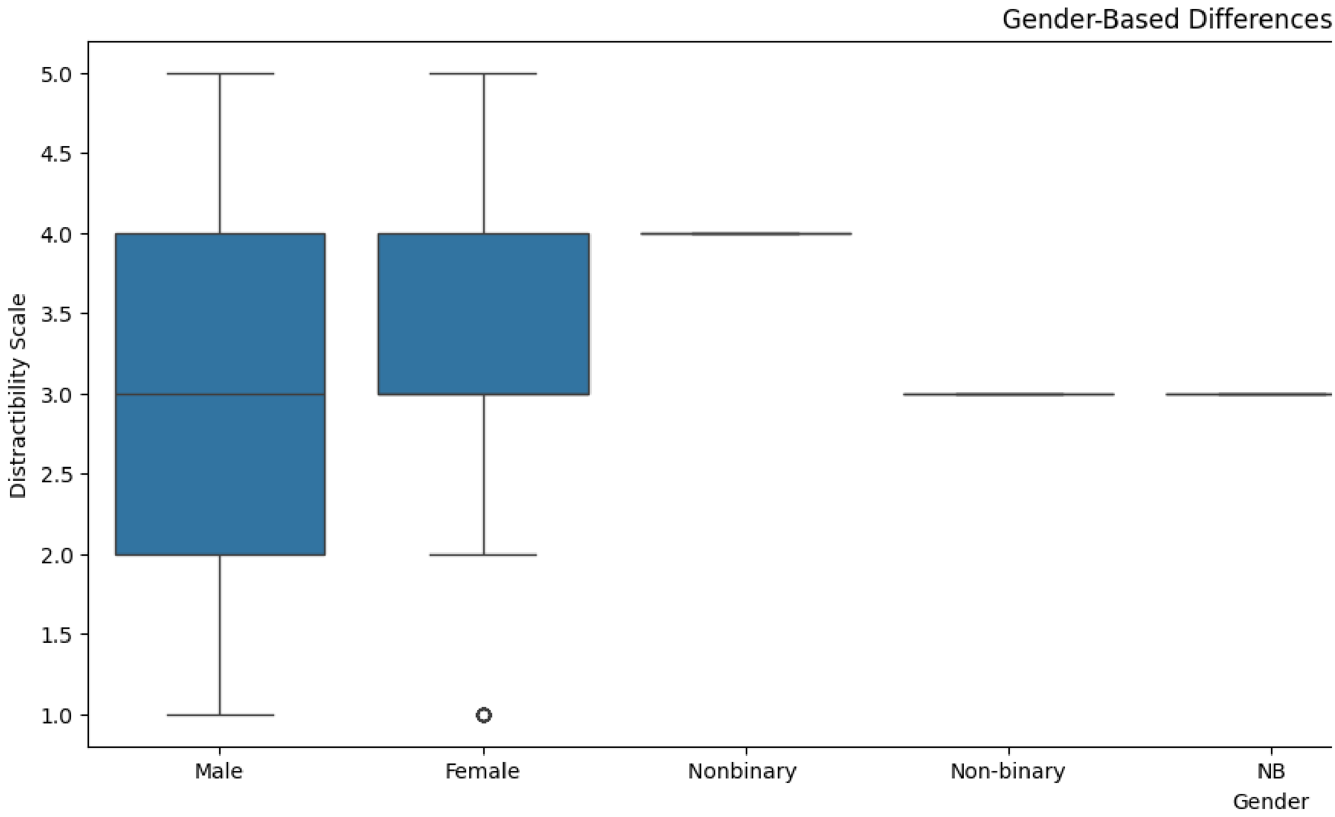
INTERPRETAION The Scatterplot shows positve Correlation

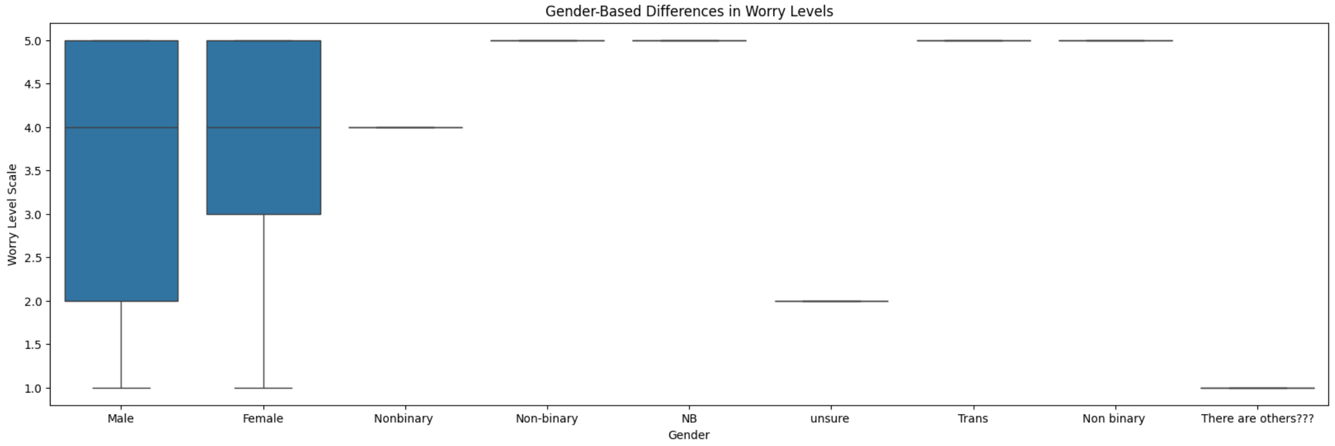
Are there any gender-based differences in social media usage patterns and their impact on mental health?

BOXPLOT

```
plt.figure(figsize=(20, 6))
sns.boxplot(x='gender', y='distractibility_scale', data=df)
plt.title('Gender-Based Differences in Distractibility')
plt.xlabel('Gender')
plt.ylabel('Distractibility Scale')
plt.show()

plt.figure(figsize=(20, 6))
sns.boxplot(x='gender', y='worry_level_scale', data=df)
plt.title('Gender-Based Differences in Worry Levels')
plt.xlabel('Gender')
plt.ylabel('Worry Level Scale')
plt.show()
```



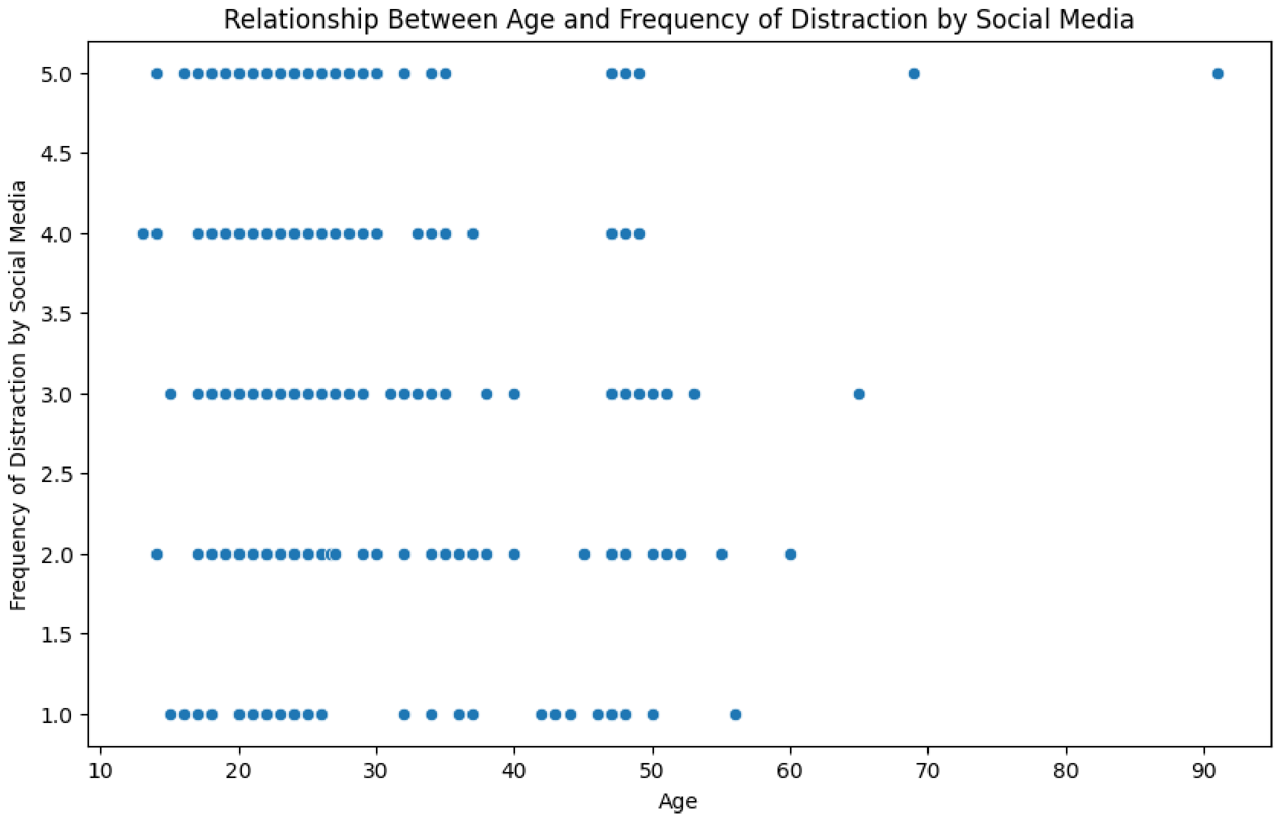


INTERPRETAION

Median of Male group is higher than Female in the boxplot of Distractability scale which means social media effects more on male catergory mental health,In case of Worrylevel don't have any gender differences.

Do younger age groups report more frequent distraction by social media during other activities?

```
plt.figure(figsize=(10, 6))
sns.scatterplot(x='age', y='frequency_social_media_distracted', data=df)
plt.title('Relationship Between Age and Frequency of Distraction by Social Media')
plt.xlabel('Age')
plt.ylabel('Frequency of Distraction by Social Media')
plt.show()
```

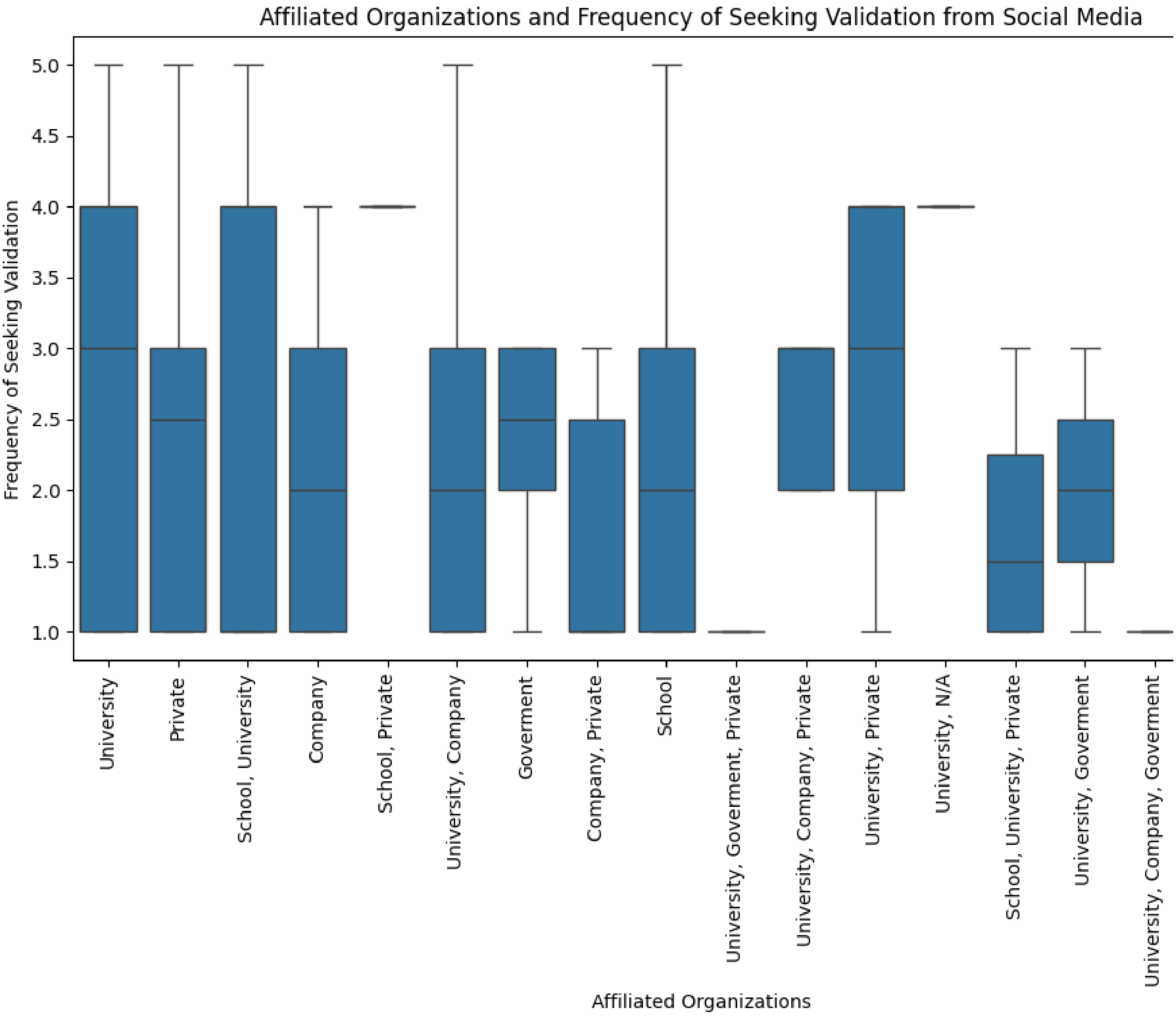


INTERPRETATION

yes younger age groups report more frequent distraction by social media during other activities

Are people who affiliate with different types of organizations more or less likely to seek validation from social media?

```
plt.figure(figsize=(12, 6))
sns.boxplot(x='affiliated_organizations', y='frequency_seeking_validation', data=df)
plt.title('Affiliated Organizations and Frequency of Seeking Validation from Social Media')
plt.xlabel('Affiliated Organizations')
plt.ylabel('Frequency of Seeking Validation')
plt.xticks(rotation=90)
plt.show()
```



INTERPRETATION

More is University students less is company,private

LOGISTIC REGRESSION MODEL

To create predictive model for sleep issues (sleep_issues_scale) based on various factors such as social media usage, mental health indicators, and demographic information.

Demographic Data: age, gender, relationship_status, occupation_status, affiliated_organizations

Social Media Usage: use_social_media, social_media_platforms, daily_social_media_time, frequency_social_media_no_purpose, frequency_social_media_distracted, restless_without_social_media

Mental Health Scales: distractibility_scale, worry_level_scale, difficulty_concentrating, compare_to_successful_people_scale, feelings_about_comparisons, frequency_seeking_validation, frequency_feeling_depressed, interest_fluctuation_scale Target Variable: sleep_issues_scale

Import Libraries

```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.model_selection import GridSearchCV
```

Convert Target Variable (sleep_issues_scale) into Binary Outcome: To predict the presence or absence of sleep issues, convert sleep_issues_scale into a binary variable. For example, classify sleep_issues_scale >= 3 as "1" (indicating sleep issues), and < 3 as "0" (indicating no sleep issues).

```
# Convert 'sleep_issues_scale' to binary classification (1 = sleep issues, 0 = no sleep issues)
df['sleep_issues'] = (df['sleep_issues_scale'] >= 3).astype(int)

# Drop the original 'sleep_issues_scale' column as we no longer need it
df.drop(columns=['sleep_issues_scale'], inplace=True)
```

```
# Fill missing values for numerical columns with the mean
numerical_cols = df.select_dtypes(include=['float64', 'int64']).columns
df[numerical_cols] = df[numerical_cols].fillna(df[numerical_cols].mean())

# Fill missing values for categorical columns with the mode
categorical_cols = df.select_dtypes(include=['object']).columns
for col in categorical_cols:
    df[col] = df[col].fillna(df[col].mode()[0])

# Check again if there are any missing values
print(df.isnull().sum())
```

```
timestamp                                                 0
age                                                       0
gender                                                    0
use_social_media                                          0
daily_social_media_time                                   0
                                                         ..
affiliated_organizations_University, Company, Private     0
affiliated_organizations_University, Goverment            0
affiliated_organizations_University, Goverment, Private   0
affiliated_organizations_University, N/A                  0
affiliated_organizations_University, Private              0
Length: 164, dtype: int64
```

Encode Categorical Variables: Use Label Encoding for binary features (gender, use_social_media). Use One-Hot Encoding for multi-category features (social_media_platforms, relationship_status, etc.).

```
# Label Encoding for binary categorical variables (e.g., gender, use_social_media)
label_encoder = LabelEncoder()
df['gender'] = label_encoder.fit_transform(df['gender'])
df['use_social_media'] = label_encoder.fit_transform(df['use_social_media'])
df['restless_without_social_media'] = label_encoder.fit_transform(df['restless_without_social_media'])

# One-Hot Encoding for multi-category variables (e.g., social_media_platforms, relationship_status, occupatior
df = pd.get_dummies(df, columns=['social_media_platforms', 'relationship_status', 'occupation_status', 'affili
```

Feature Scaling: Logistic regression benefits from scaled data, so we will standardize numerical columns.

```
# List of numerical features to scale
numerical_features = [
    'age', 'daily_social_media_time', 'frequency_social_media_no_purpose', 'frequency_social_media_distracted'
    'distractibility_scale', 'worry_level_scale', 'difficulty_concentrating', 'compare_to_successful_people_sc
    'feelings_about_comparisons', 'frequency_seeking_validation', 'frequency_feeling_depressed', 'interest_flu
]

# Initialize the scaler
scaler = StandardScaler()

# Scale the numerical features
df[numerical_features] = scaler.fit_transform(df[numerical_features])
```

```
# Define feature variables (all columns except the target)
X = df.drop(columns=['sleep_issues', 'timestamp']) # Changed 'sleep_issues_scale' to 'sleep_issues' and 'Tim

# Define the target variable (sleep_issues, binary: 1 = sleep issues, 0 = no sleep issues)
y = df['sleep_issues'] # Changed 'sleep_issues_scale' to 'sleep_issues'


# Now split the data into training and test sets
X = df.drop(columns=['sleep_issues', 'timestamp'])
y = df['sleep_issues']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Now train the model
model = LogisticRegression(solver='liblinear')
model.fit(X_train, y_train)
```

```
LogisticRegression
LogisticRegression(solver='liblinear')
```

```
# List of numerical features to scale
numerical_features = [
    'age', 'daily_social_media_time', 'frequency_social_media_no_purpose', 'frequency_social_media_distracte
    'distractibility_scale', 'worry_level_scale', 'difficulty_concentrating', 'compare_to_successful_people_
    'feelings_about_comparisons', 'frequency_seeking_validation', 'frequency_feeling_depressed', 'interest_f
]

# Initialize the scaler
scaler = StandardScaler()

# Scale the numerical features
df[numerical_features] = scaler.fit_transform(df[numerical_features])



# Define feature variables (all columns except the target)
X = df.drop(columns=['sleep_issues', 'timestamp']) # Changed 'sleep_issues_scale' to 'sleep_issues' and 'Tim

# Define the target variable (sleep_issues, binary: 1 = sleep issues, 0 = no sleep issues)
y = df['sleep_issues'] # Changed 'sleep_issues_scale' to 'sleep_issues'



# Split the data into training and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```