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

Loaded the CSV file into a Pandas DataFrame.

```
df = pd.read_csv("Indian_Kids_Screen_Time.csv")
```

Used .head(), .info(), and .describe() to inspect data structure, data types, and summary statistics.

```
df.head(10)

   Age  Gender  Avg_Daily_Screen_Time_hr Primary_Device  \
0   14    Male                      3.99     Smartphone   
1   11  Female                      4.61         Laptop   
2   18  Female                      3.73             TV   
3   15  Female                      1.21         Laptop   
4   12  Female                      5.89     Smartphone   
5   14  Female                      4.88     Smartphone   
6   17    Male                      2.97             TV   
7   10    Male                      2.74             TV   
8   14    Male                      4.61         Laptop   
9   18    Male                      3.24         Tablet   

   Exceeded_Recommended_Limit  Educational_to_Recreational_Ratio  \
0                        True                               0.42   
1                        True                               0.30   
2                        True                               0.32   
3                       False                               0.39   
4                        True                               0.49   
5                        True                               0.44   
6                       False                               0.48   
7                        True                               0.54   
8                        True                               0.36   
9                        True                               0.48   

             Health_Impacts Urban_or_Rural  
0    Poor Sleep, Eye Strain          Urban  
1                Poor Sleep          Urban  
2                Poor Sleep          Urban  
3                       NaN          Urban  
4      Poor Sleep, Anxiety          Urban  
5                Poor Sleep          Urban  
6                       NaN          Rural  
7                       NaN          Urban  
8      Poor Sleep, Anxiety          Rural  
9  Poor Sleep, Obesity Risk          Urban  
```

```
df.describe()
```

```
               Age  Avg_Daily_Screen_Time_hr  \
count  9712.000000               9712.000000
mean     12.979201                  4.352837
std       3.162437                  1.718232
min       8.000000                  0.000000
25%      10.000000                  3.410000
50%      13.000000                  4.440000
75%      16.000000                  5.380000
max      18.000000                 13.890000

       Educational_to_Recreational_Ratio
count                        9712.000000
mean                            0.427226
std                             0.073221
min                             0.300000
25%                             0.370000
50%                             0.430000
75%                             0.480000
max                             0.600000
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9712 entries, 0 to 9711
Data columns (total 8 columns):
 #   Column                             Non-Null Count  Dtype
---  ------                             --------------  -----
 0   Age                                9712 non-null   int64
 1   Gender                             9712 non-null   object
 2   Avg_Daily_Screen_Time_hr           9712 non-null   float64
 3   Primary_Device                     9712 non-null   object
 4   Exceeded_Recommended_Limit         9712 non-null   bool
 5   Educational_to_Recreational_Ratio  9712 non-null   float64
 6   Health_Impacts                     6494 non-null   object
 7   Urban_or_Rural                     9712 non-null   object
dtypes: bool(1), float64(2), int64(1), object(4)
memory usage: 540.7+ KB
```

```
df.columns
```

```
Index(['Age', 'Gender', 'Avg_Daily_Screen_Time_hr', 'Primary_Device',
       'Exceeded_Recommended_Limit',
'Educational_to_Recreational_Ratio',
       'Health_Impacts', 'Urban_or_Rural'],
      dtype='object')
```

Identified and removed duplicate rows to ensure data quality. Checked for missing values to assess completeness.

```
df.duplicated().sum()

44

df[df.duplicated()] #to view the duplicated rows in the dataset

df.drop_duplicates(inplace=True) #dropping the duplicated from the
dataset and modifying the original dataset
```

Checking for the null values if any in the dataset.

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

Age                                     0
Gender                                  0
Avg_Daily_Screen_Time_hr                0
Primary_Device                          0
Exceeded_Recommended_Limit              0
Educational_to_Recreational_Ratio       0
Health_Impacts                       3180
Urban_or_Rural                          0
dtype: int64

df.fillna(0,inplace=True) #filling the NaN values in Health_Impacts to
0

df.head(10)

   Age  Gender  Avg_Daily_Screen_Time_hr Primary_Device  \
0   14    Male                      3.99     Smartphone   
1   11  Female                      4.61         Laptop   
2   18  Female                      3.73             TV   
3   15  Female                      1.21         Laptop   
4   12  Female                      5.89     Smartphone   
5   14  Female                      4.88     Smartphone   
6   17    Male                      2.97             TV   
7   10    Male                      2.74             TV   
8   14    Male                      4.61         Laptop   
9   18    Male                      3.24         Tablet   

   Exceeded_Recommended_Limit  Educational_to_Recreational_Ratio  \
0                        True                               0.42   
1                        True                               0.30   
2                        True                               0.32   
3                       False                               0.39   
4                        True                               0.49   
5                        True                               0.44   
6                       False                               0.48   
7                        True                               0.54   
8                        True                               0.36   
9                        True                               0.48   
```

```
             Health_Impacts Urban_or_Rural
0    Poor Sleep, Eye Strain          Urban
1                Poor Sleep          Urban
2                Poor Sleep          Urban
3                       NaN          Urban
4       Poor Sleep, Anxiety          Urban
5                Poor Sleep          Urban
6                       NaN          Rural
7                       NaN          Urban
8       Poor Sleep, Anxiety          Rural
9  Poor Sleep, Obesity Risk          Urban
```

```
df = df.sort_values(by='Age') #sorting the Age column in increasing order
df
```

```
      Age  Gender  Avg_Daily_Screen_Time_hr Primary_Device  \
0       8  Female                      3.34     Smartphone
602     8    Male                      7.90     Smartphone
603     8    Male                      1.85         Tablet
604     8    Male                      3.80             TV
605     8    Male                      6.27             TV
...   ...     ...                       ...            ...
9289   18    Male                      4.81     Smartphone
9148   18  Female                      5.23             TV
9135   18    Male                      4.81             TV
9162   18  Female                      4.26     Smartphone
9711   18  Female                      5.47         Tablet

      Exceeded_Recommended_Limit  Educational_to_Recreational_Ratio  \
0                           True                               0.49
602                         True                               0.55
603                        False                               0.50
604                         True                               0.57
605                         True                               0.60
...                          ...                                ...
9289                        True                               0.38
9148                        True                               0.47
9135                        True                               0.35
9162                        True                               0.40
9711                        True                               0.40

              Health_Impacts Urban_or_Rural
0     Poor Sleep, Eye Strain          Urban
602               Poor Sleep          Urban
603                      NaN          Rural
604               Poor Sleep          Rural
605               Poor Sleep          Rural
...                      ...            ...
```

```
9289  Poor Sleep, Eye Strain          Urban
9148                 Anxiety          Rural
9135  Poor Sleep, Eye Strain          Rural
9162  Poor Sleep, Eye Strain          Urban
9711     Eye Strain, Anxiety          Urban

[9712 rows x 8 columns]

df.reset_index(drop=True, inplace=True) #resetting the index after
arrange 'AGE' column in increasing order
df

      Age  Gender  Avg_Daily_Screen_Time_hr Primary_Device  \
0       8  Female                      3.34     Smartphone
1       8    Male                      7.90     Smartphone
2       8    Male                      1.85         Tablet
3       8    Male                      3.80             TV
4       8    Male                      6.27             TV
...   ...     ...                       ...            ...
9707   18    Male                      4.81     Smartphone
9708   18  Female                      5.23             TV
9709   18    Male                      4.81             TV
9710   18  Female                      4.26     Smartphone
9711   18  Female                      5.47         Tablet

      Exceeded_Recommended_Limit  Educational_to_Recreational_Ratio  \
0                           True                               0.49
1                           True                               0.55
2                          False                               0.50
3                           True                               0.57
4                           True                               0.60
...                          ...                                ...
9707                        True                               0.38
9708                        True                               0.47
9709                        True                               0.35
9710                        True                               0.40
9711                        True                               0.40

              Health_Impacts Urban_or_Rural
0     Poor Sleep, Eye Strain          Urban
1                 Poor Sleep          Urban
2                        NaN          Rural
3                 Poor Sleep          Rural
4                 Poor Sleep          Rural
...                      ...            ...
9707  Poor Sleep, Eye Strain          Urban
9708                 Anxiety          Rural
9709  Poor Sleep, Eye Strain          Rural
9710  Poor Sleep, Eye Strain          Urban
9711     Eye Strain, Anxiety          Urban
```

```
[9712 rows x 8 columns]
```

```
df['Gender'].value_counts() #to check the ratio of male and female
```

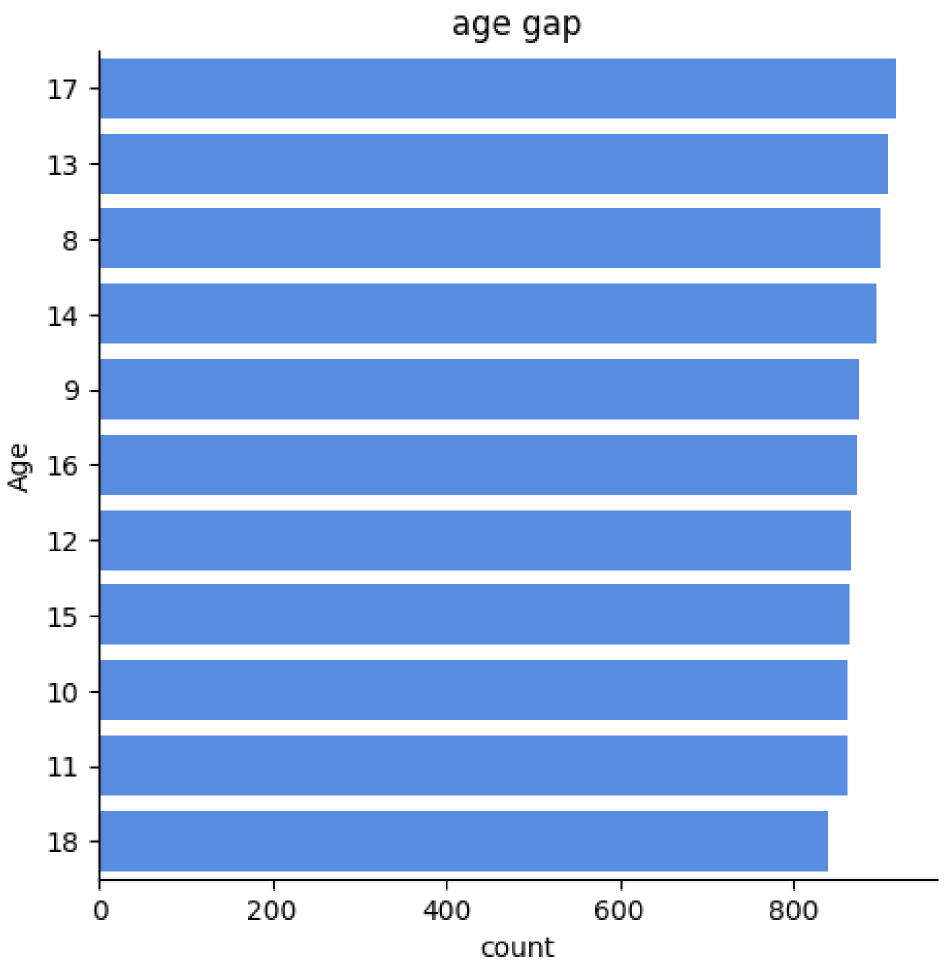
```
Gender
Male      4928
Female    4740
Name: count, dtype: int64
```

```
df['Age'].value_counts() # to check the different age groups in the
dataset
```

```
Age
17    918
13    910
8     900
14    895
9     875
16    874
12    867
15    864
10    863
11    863
18    839
Name: count, dtype: int64
```

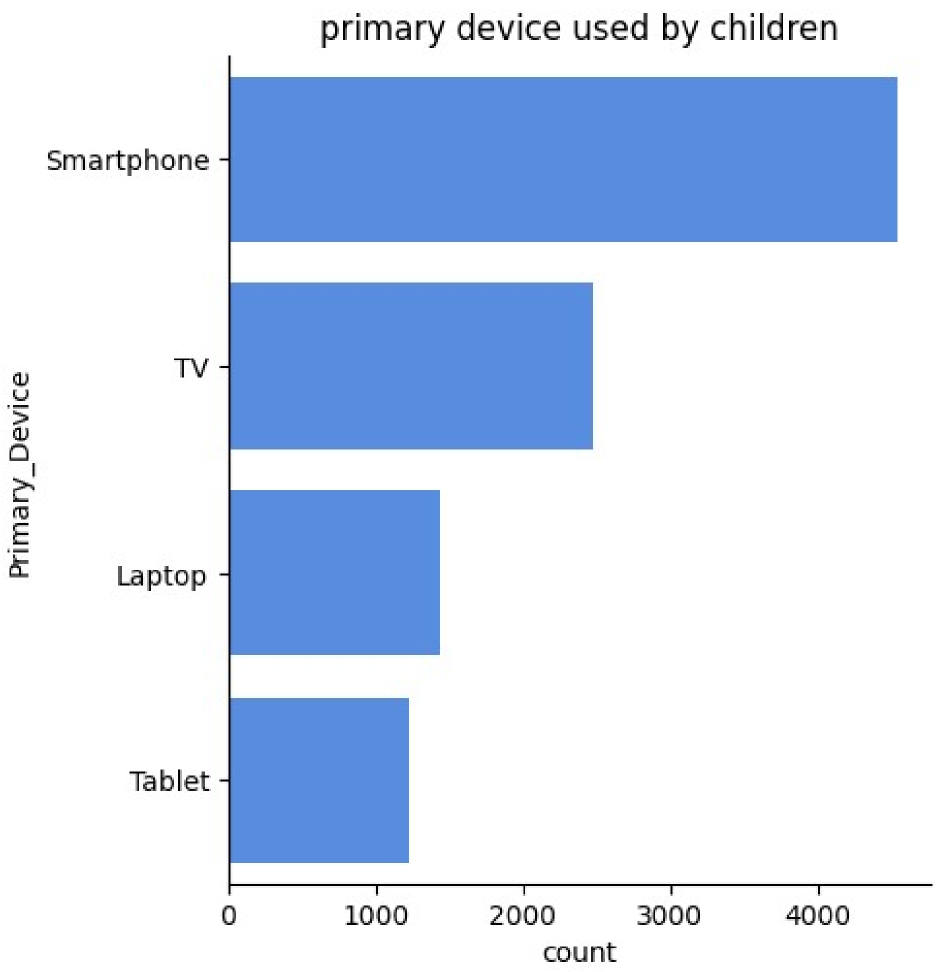
Which ages appear most frequently in the dataset?

```
sns.catplot(y="Age", data=df,kind = 'count',
            order=df['Age'].value_counts().index,
            color='#4287f5')
plt.title("age gap")
plt.show()
```



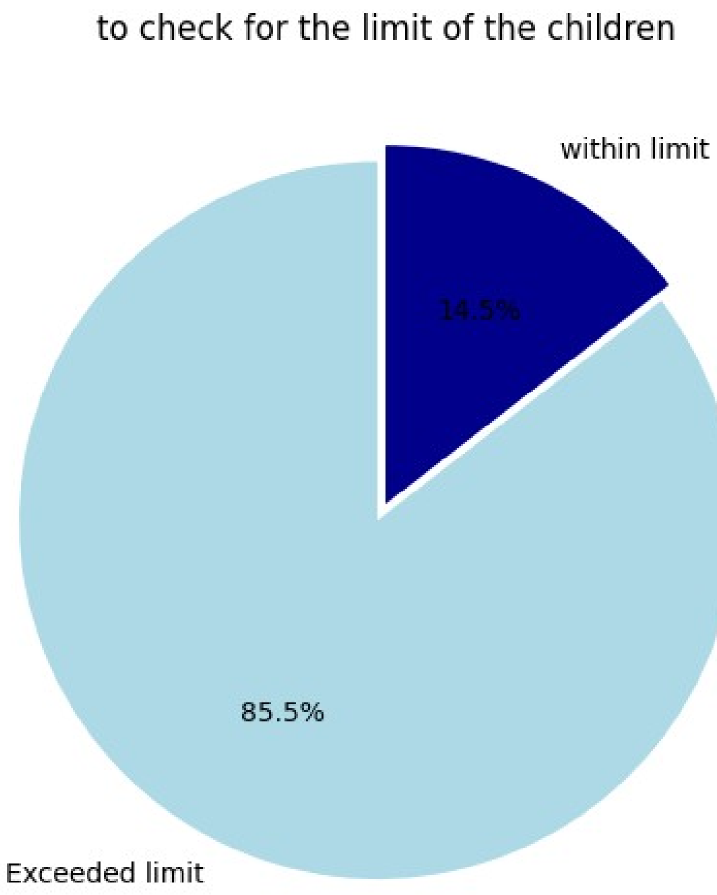
“What is the most commonly used digital device among children?”

```
sns.catplot(y="Primary_Device" , data=df, kind='count',
            order=df['Primary_Device'].value_counts().index,
            color='#4287f5')
plt.title('primary device used by children')
plt.show()
```



What proportion of children are exceeding the recommended screen time limit?

```
count_values = df['Exceeded_Recommended_Limit'].value_counts()
count_values

labels= ['Exceeded limit','within limit']
plt.figure(figsize=(6,6))
plt.pie(count_values,labels=labels,autopct='%1.1f%%',startangle=90,
colors =['lightblue','darkblue'],explode=(0.05, 0))
plt.title("to check for the limit of the children")
plt.show()
```



to check for the limit of the children

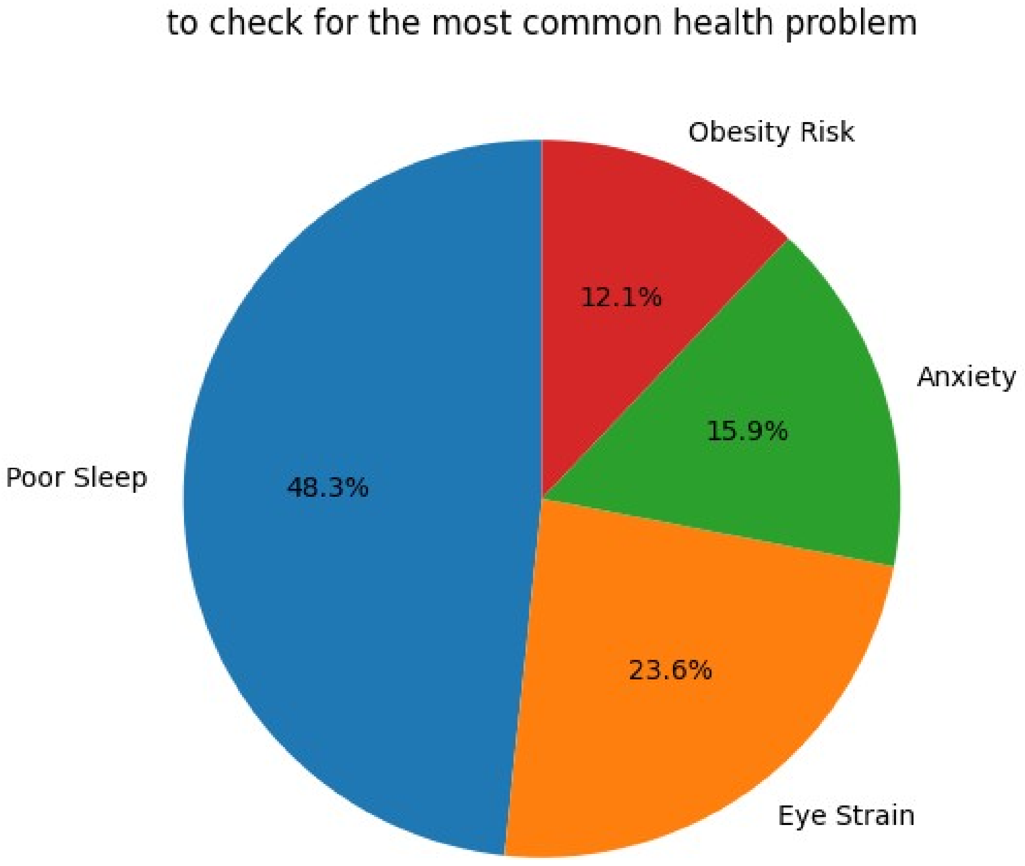
The Health_Impacts column was split and exploded to separate multiple health issues into individual rows. This made it easier to analyze and visualize each health impact independently.

```
df['Health_Impacts']= df['Health_Impacts'].str.split(', ')
df=df.explode('Health_Impacts').reset_index(drop=True)
df.head()

   Age  Gender  Avg_Daily_Screen_Time_hr Primary_Device  \
0    8  Female                      3.34     Smartphone
1    8  Female                      3.34     Smartphone
2    8    Male                      7.90     Smartphone
3    8    Male                      1.85         Tablet
4    8    Male                      3.80             TV

   Exceeded_Recommended_Limit  Educational_to_Recreational_Ratio  \
0                        True                               0.49
1                        True                               0.49
2                        True                               0.55
3                       False                               0.50
4                        True                               0.57
```

```
  Health_Impacts Urban_or_Rural
0     Poor Sleep          Urban
1     Eye Strain          Urban
2     Poor Sleep          Urban
3            NaN          Rural
4     Poor Sleep          Rural
```

The code counts how often each health issue appears in the dataset. A pie chart is then used to visualize the most common health problems among children.

```
# Count occurrences of each health problem
Health_Problem=df['Health_Impacts'].value_counts()
Health_Problem

health_counts = df['Health_Impacts'].value_counts()
plt.figure(figsize=(6,6))
plt.pie(Health_Problem,autopct='%1.1f%
%',startangle=90,labels=health_counts.index,)
plt.title("to check for the most common health problem")
plt.show()
```



```
df.columns
```

```
Index(['Age', 'Gender', 'Avg_Daily_Screen_Time_hr', 'Primary_Device',
       'Exceeded_Recommended_Limit',
'Educational_to_Recreational_Ratio',
       'Health_Impacts', 'Urban_or_Rural'],
      dtype='object')
```

```
#Now we have started working on the column 'Avg_Daily_Screen_Time_hr'
```

```
df['Avg_Daily_Screen_Time_hr'].describe()
```

```
count    13290.000000
mean         4.501646
std          1.613662
min          0.000000
25%          3.620000
50%          4.550000
75%          5.450000
max         13.890000
Name: Avg_Daily_Screen_Time_hr, dtype: float64
```

```
df.groupby('Gender')['Avg_Daily_Screen_Time_hr'].mean() #to compare
the average time between female and male
```

```
Gender
Female    4.475504
Male      4.526634
Name: Avg_Daily_Screen_Time_hr, dtype: float64
```

```
df.loc[df['Avg_Daily_Screen_Time_hr'].idxmax()] #to check for the
maximum in the column
```

```
Age                                          10
Gender                                   Female
Avg_Daily_Screen_Time_hr                  13.89
Primary_Device                       Smartphone
Exceeded_Recommended_Limit                 True
Educational_to_Recreational_Ratio          0.58
Health_Impacts                       Eye Strain
Urban_or_Rural                            Urban
Name: 3218, dtype: object
```

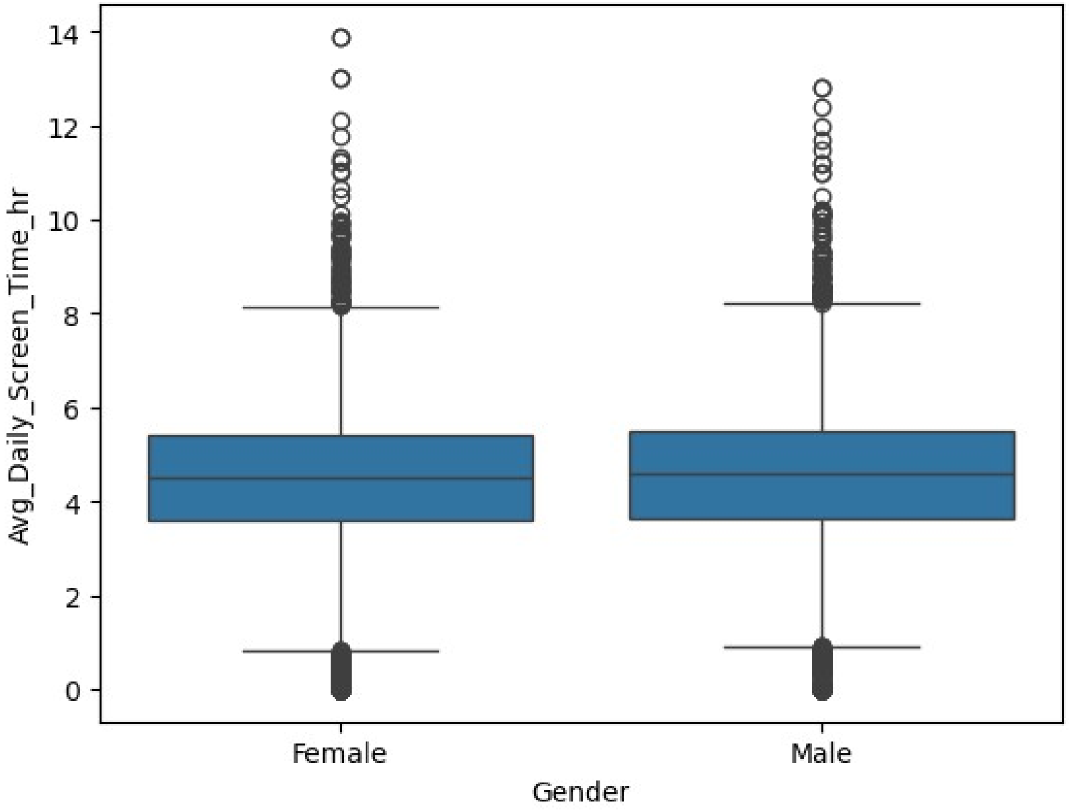
```
df.loc[df['Avg_Daily_Screen_Time_hr'].idxmin()] # to check for the
minimum index in the column
```

```
Age                                           8
Gender                                   Female
Avg_Daily_Screen_Time_hr                    0.0
Primary_Device                       Smartphone
Exceeded_Recommended_Limit                False
```

```
Educational_to_Recreational_Ratio          0.47
Health_Impacts                              NaN
Urban_or_Rural                            Urban
Name: 18, dtype: object
```

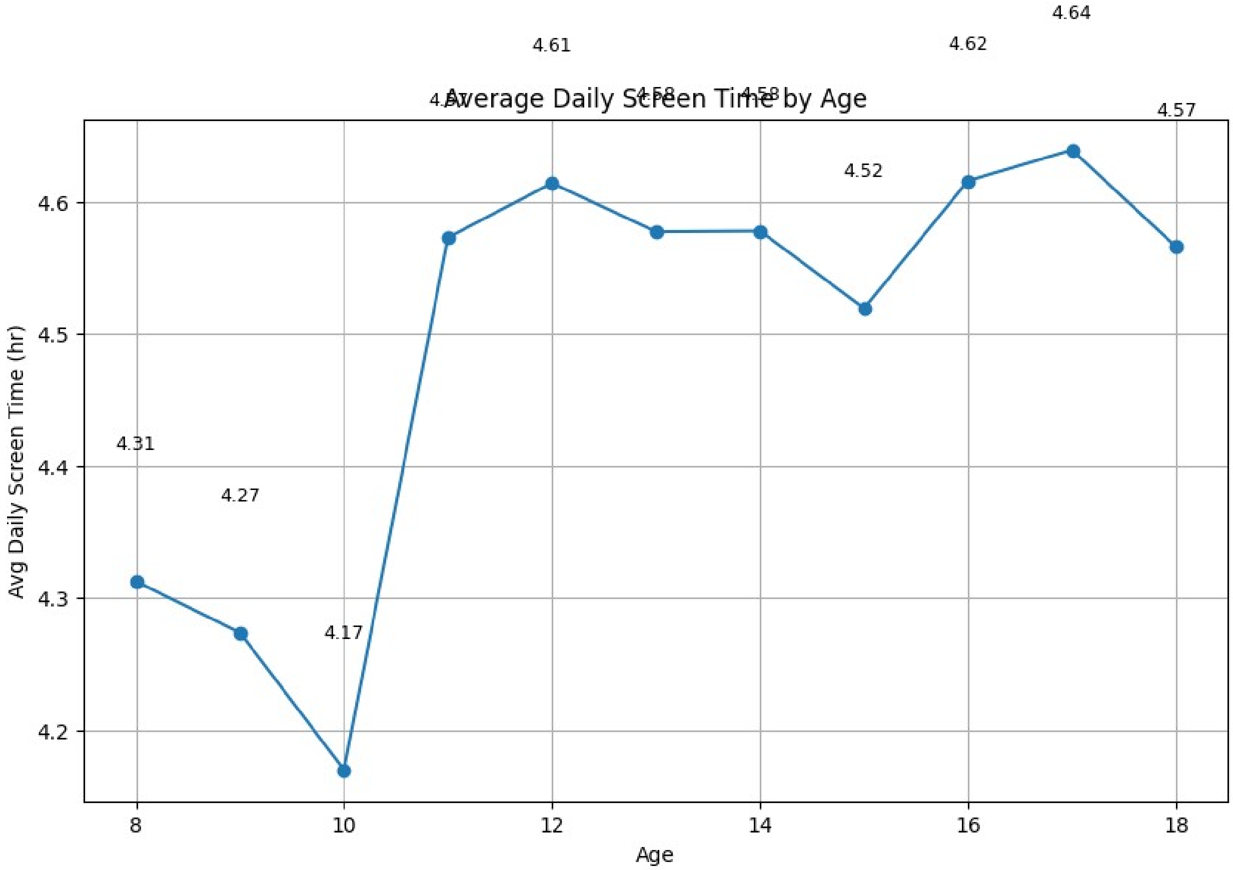
```
sns.boxplot(x='Gender', y='Avg_Daily_Screen_Time_hr', data=df)
#This code creates a boxplot to compare average daily screen time
between genders. It helps visualize the distribution, median, and
outliers for each gender group.
```

```
<Axes: xlabel='Gender', ylabel='Avg_Daily_Screen_Time_hr'>
```



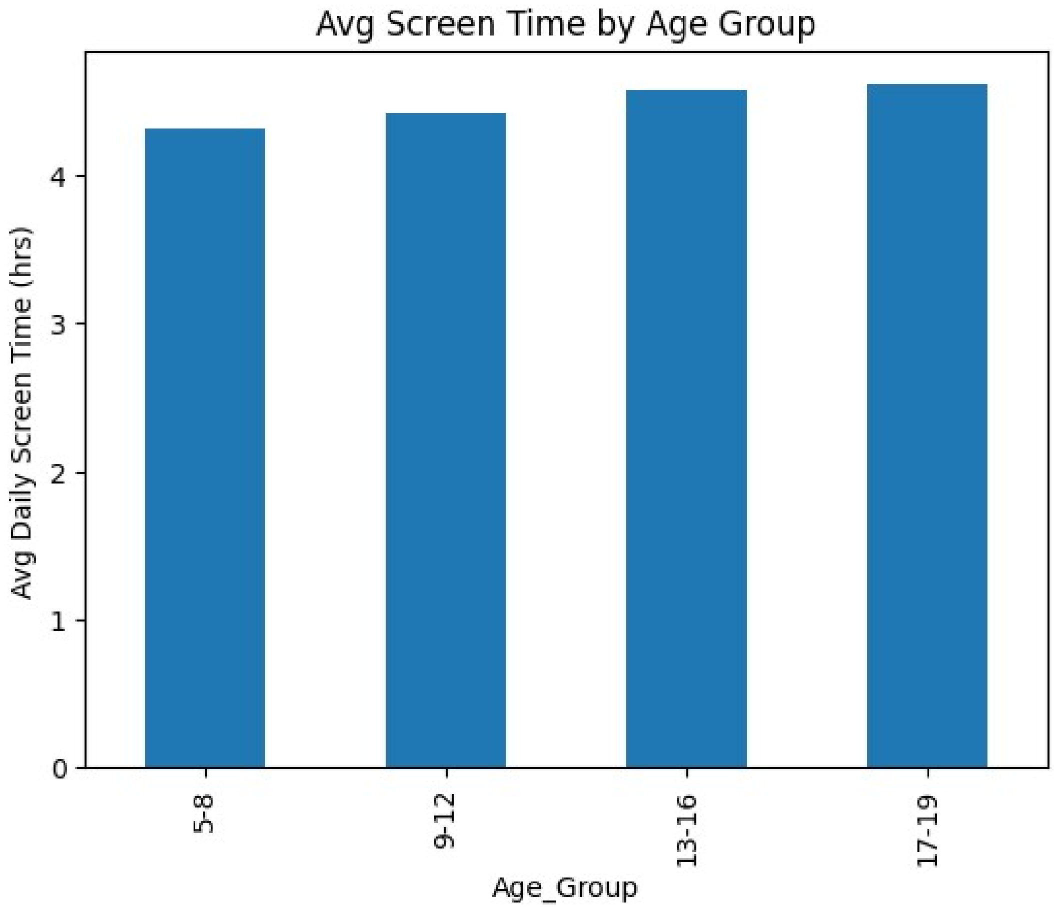
The code groups data by age and calculates the average daily screen time for each group. It then plots a line chart with point labels to visualize how screen time changes with age.

```
# Group and get average screen time by age
avg_screen_time_by_age = df.groupby('Age')
['Avg_Daily_Screen_Time_hr'].mean()

# Plot line
plt.figure(figsize=(10, 6))
plt.plot(avg_screen_time_by_age.index, avg_screen_time_by_age.values,
marker='o', linestyle='-')
plt.title("Average Daily Screen Time by Age")
plt.xlabel("Age")
plt.ylabel("Avg Daily Screen Time (hr)")
plt.grid(True)

# Annotate each point with its value
for x, y in zip(avg_screen_time_by_age.index,
avg_screen_time_by_age.values):
    plt.text(x, y + 0.1, f"{y:.2f}", ha='center', fontsize=9,
color='black')

plt.show()
```



```


# Create age groups
df['Age_Group'] = pd.cut(df['Age'], bins=[4, 8, 12, 16, 19],
labels=['5-8', '9-12', '13-16', '17-19'])

# Average screen time per age group
df.groupby('Age_Group')
['Avg_Daily_Screen_Time_hr'].mean().plot(kind='bar', title='Avg Screen
Time by Age Group')
plt.ylabel("Avg Daily Screen Time (hrs)")
plt.show()

#The code categorizes ages into groups (e.g., 5–8, 9–12) and calculates the average screen time for each group. A bar chart is then plotted to compare screen time across these age ranges.
```

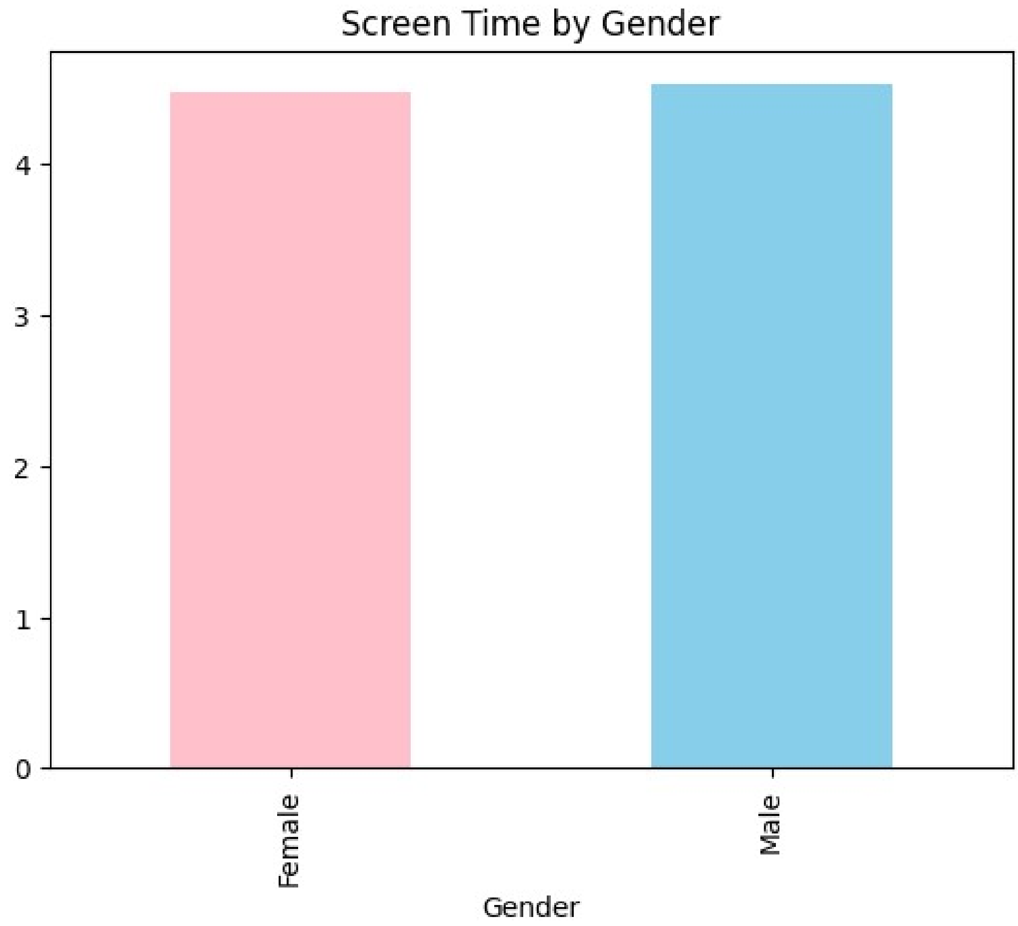
```
C:\Users\Admin\AppData\Local\Temp\ipykernel_9036\2035169107.py:5: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.
  df.groupby('Age_Group')['Avg_Daily_Screen_Time_hr'].mean().plot(kind='bar', title='Avg Screen Time by Age Group')
```



```
df.groupby('Gender')
['Avg_Daily_Screen_Time_hr'].mean().plot(kind='bar', color=['pink', 'skyblue'], title='Screen Time by Gender')
```

```
#The code calculates the average daily screen time for each gender
using groupby. It then visualizes the comparison with a color-coded
bar chart.
```

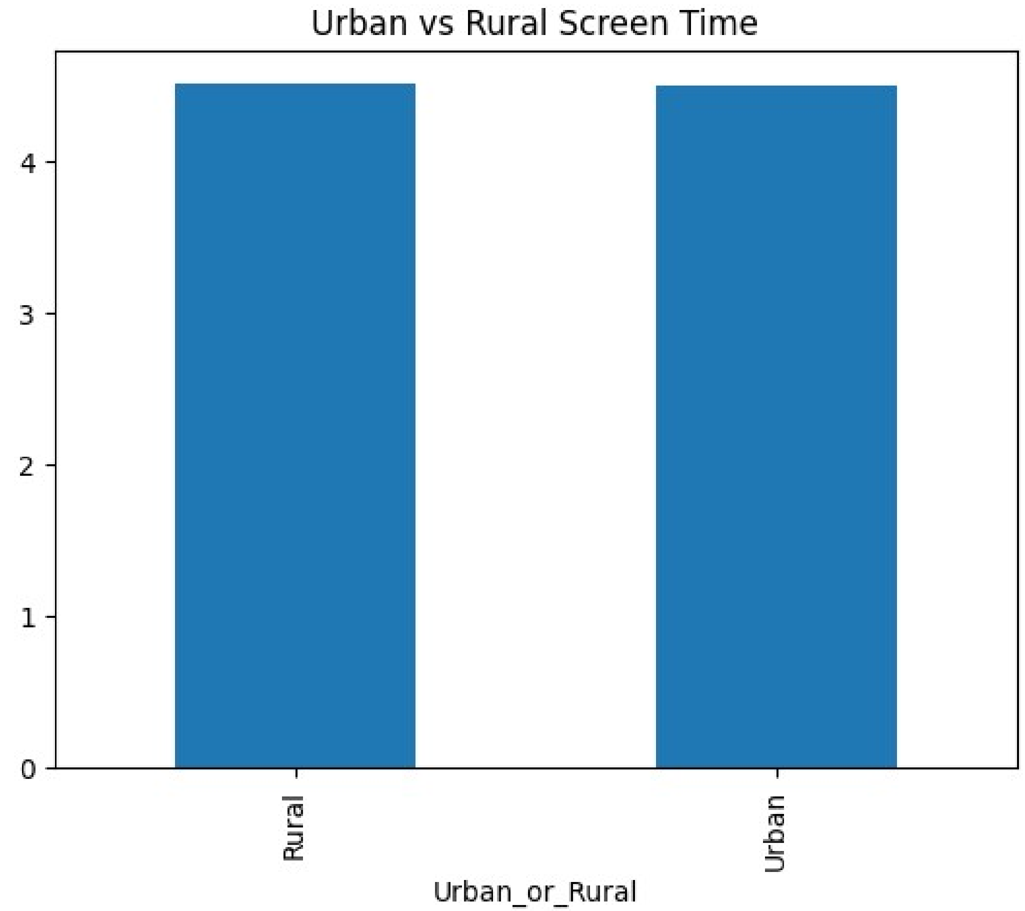
```
<Axes: title={'center': 'Screen Time by Gender'}, xlabel='Gender'>
```



The code groups the data by Urban or Rural areas and calculates the average screen time for each. A bar chart is then used to compare screen time between urban and rural children.

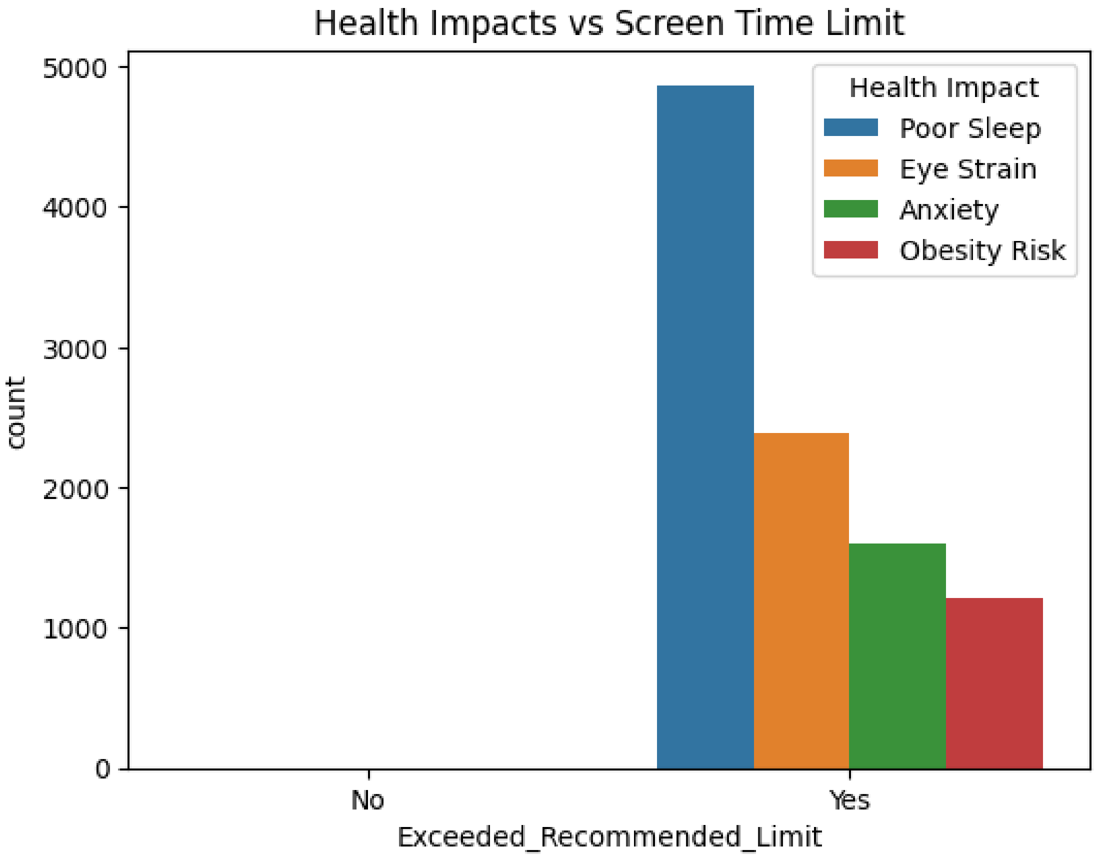
```
df.groupby('Urban_or_Rural')
['Avg_Daily_Screen_Time_hr'].mean().plot(kind='bar', title='Urban vs
Rural Screen Time')
```

```
<Axes: title={'center': 'Urban vs Rural Screen Time'},
xlabel='Urban_or_Rural'>
```



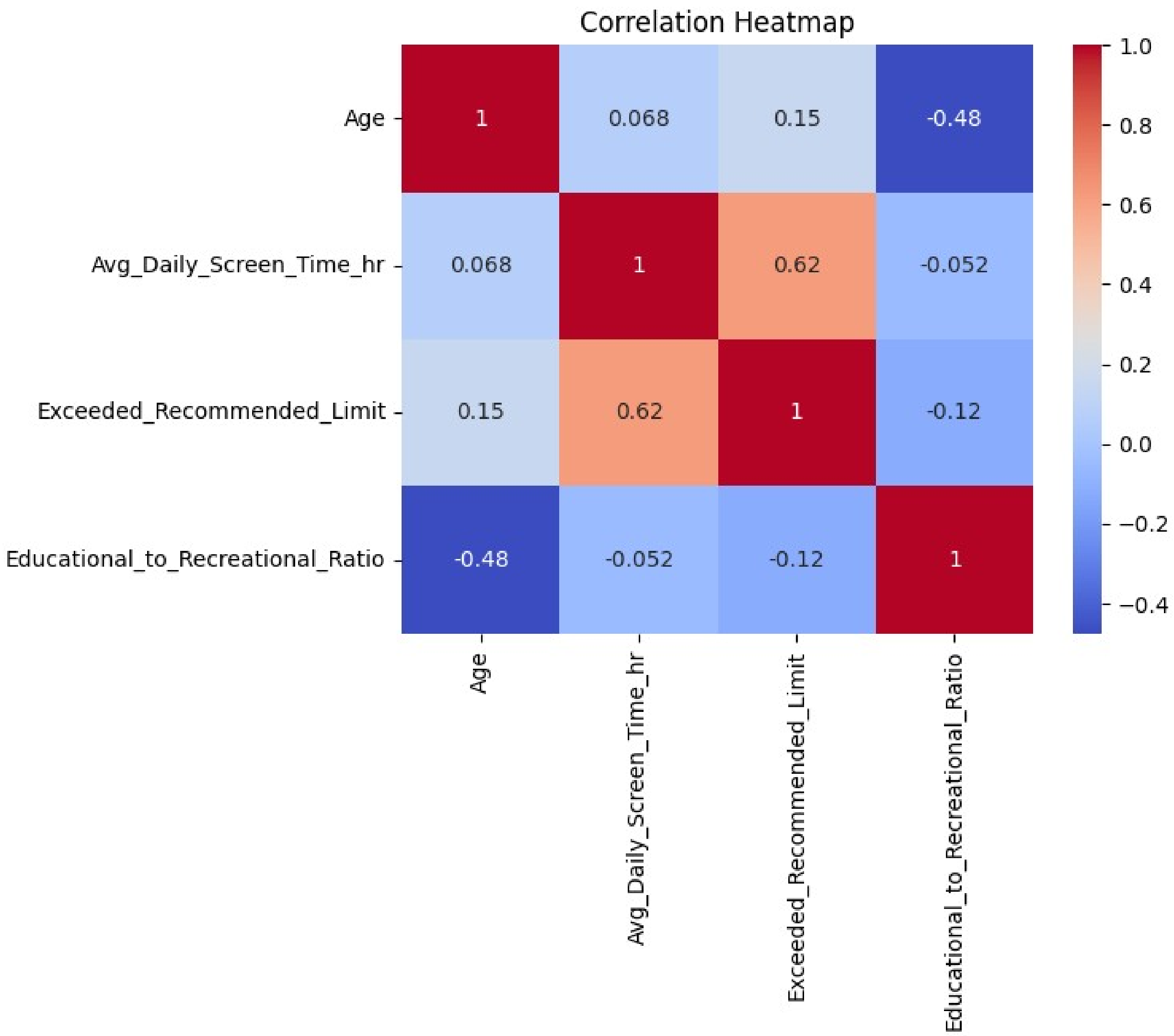
The code plots a grouped bar chart to compare health issues based on whether children exceeded recommended screen time. It visually highlights which health problems are more common among those with excessive screen time.

```
sns.countplot(x='Exceeded_Recommended_Limit', hue='Health_Impacts',
data=df)
plt.title("Health Impacts vs Screen Time Limit")
plt.xticks([0, 1], ['No', 'Yes'])
plt.legend(title="Health Impact", loc='upper right')
plt.show()
```

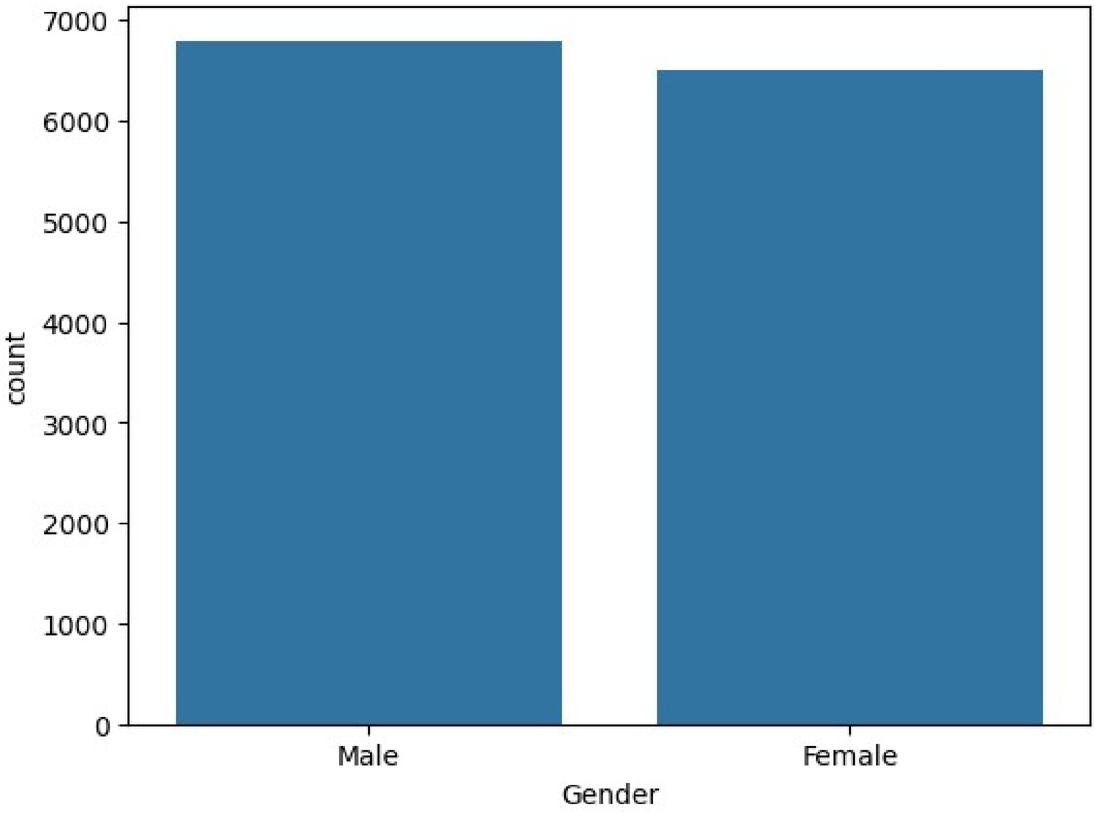


This code creates a heatmap to visualize correlations between all numeric variables in the dataset. It helps identify strong positive or negative relationships, aiding deeper insights into variable interactions.

```
sns.heatmap(df.corr(numeric_only=True), annot=True, cmap='coolwarm')
plt.title("Correlation Heatmap")
plt.show()
```

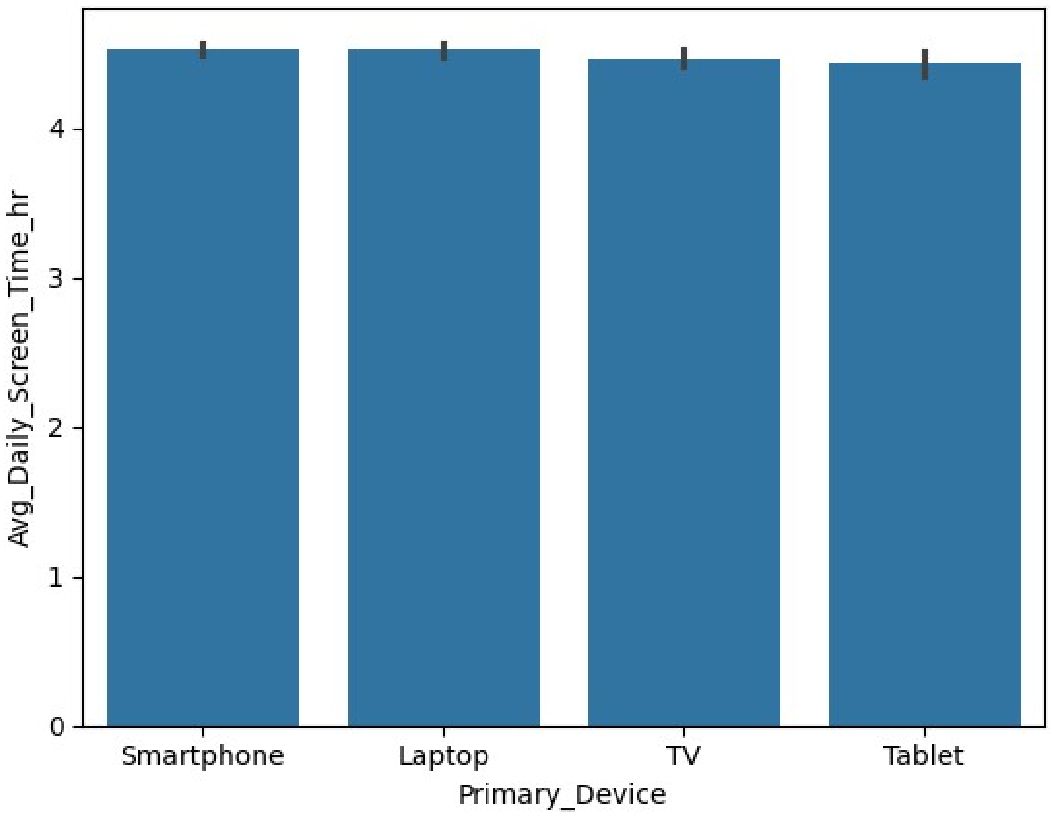


```
sns.countplot(data=df, x='Gender') #to visualise the ratio of gender
in the dataset

<Axes: xlabel='Gender', ylabel='count'>
```



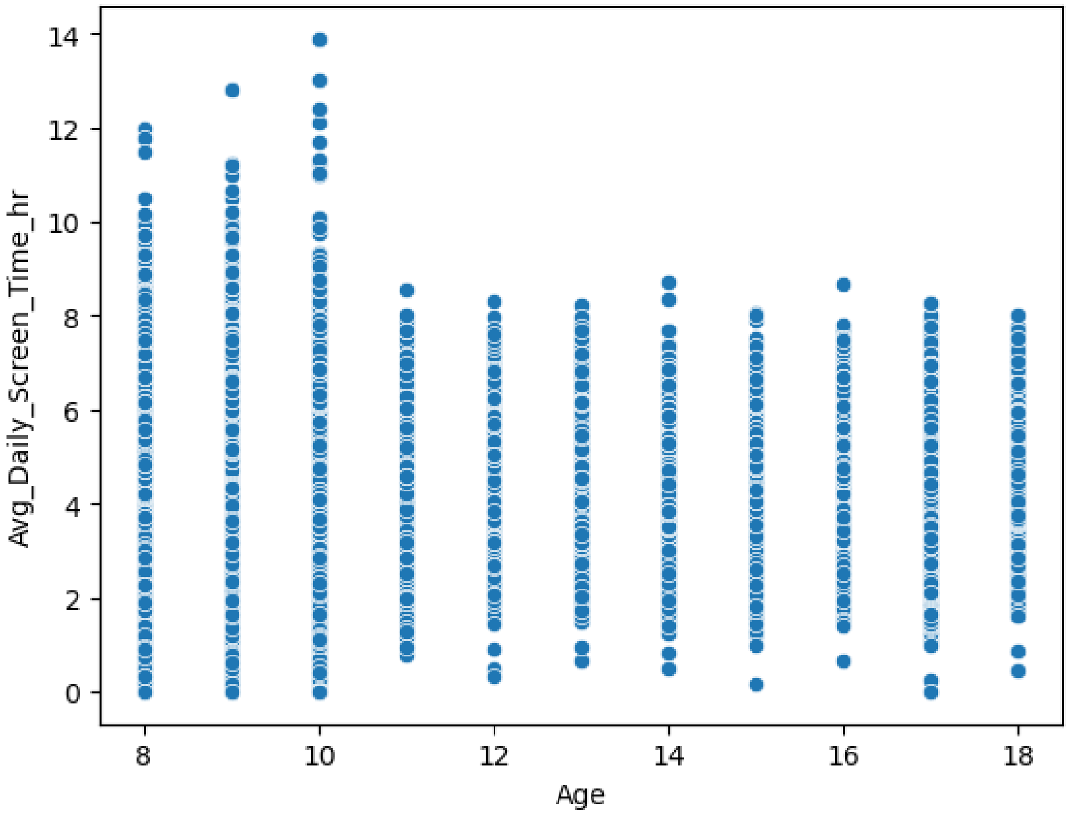
```
sns.barplot(data=df, x='Primary_Device', y='Avg_Daily_Screen_Time_hr')
```

```
<Axes: xlabel='Primary_Device', ylabel='Avg_Daily_Screen_Time_hr'>
```

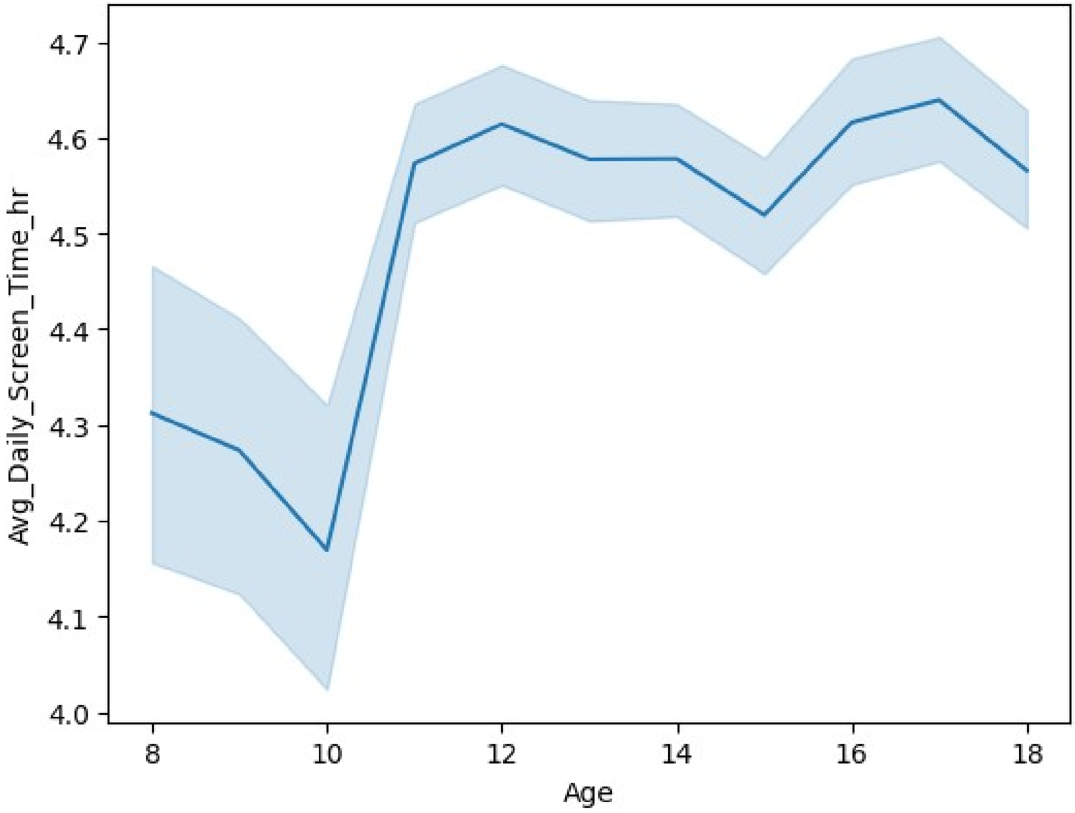


```
sns.scatterplot(data=df, x='Age', y='Avg_Daily_Screen_Time_hr')
#scatterplot to visualise the avg_daily_screen_time_hr for every age
group in the Age column
```

```
<Axes: xlabel='Age', ylabel='Avg_Daily_Screen_Time_hr'>
```



```
sns.lineplot(data=df, x='Age', y='Avg_Daily_Screen_Time_hr')
#lineplot to visualise the avg_daily_screen_time_hr for every age
group in the Age column

<Axes: xlabel='Age', ylabel='Avg_Daily_Screen_Time_hr'>
```



#SUMMARY OF THE DATASET Data Analysis Summary: Children’s Screen Time Study This analysis explores patterns, habits, and health impacts of daily screen time among children using various statistical and visualization techniques in Python (Pandas, Matplotlib, Seaborn).

1. Data Cleaning & Preparation Removed NaN values and checked for duplicate entries to ensure data quality.

Converted the Health_Impacts column from comma-separated strings into individual rows using str.split() and explode() for clearer analysis.

Created age groups (5–8, 9–12, 13–16, 17–19) using pd.cut() to simplify age-based comparisons.

1. Descriptive Statistics & Distributions Used histograms and KDE plots to analyze the distribution of the Educational to Recreational Screen Time Ratio.

Visualized average screen time by:

Age (line plot)

Age group (bar chart)

Gender (bar chart)

Urban vs Rural location (bar chart)

Boxplots were used to show screen time variation and outliers across genders.

1. Category & Count Analysis Used countplot, catplot, and value_counts() to:

Identify the most common primary devices used.

Understand the age distribution and gender balance.

Examine how many children exceeded the recommended screen time.

1. Health Impact Analysis Exploded multi-label health issues for better granularity.

Plotted a pie chart showing the most common health impacts of screen time.

Analyzed how exceeding screen time limits correlates with reported health issues using a grouped countplot.

1. Correlation Analysis A heatmap of numeric features revealed relationships between screen time, age, and other factors.

This helped highlight which factors might influence excessive screen use or health outcomes.

□ Conclusion The study shows a clear trend of increasing screen time with age.

Children who exceeded the recommended limits were more likely to report health issues such as eye strain and poor sleep.

Gender and urban/rural differences were also notable, and the use of educational vs. recreational screen time varied widely.