Out[6]:

	User_ID	Date	Mood_Score	Sleep_Hours	Sleep_Quality	Screen_Time_Hours	Physical_Activity_Min	Social_Interactic
0	U038	2023-06-07	7	10.0	Excellent	5.1	41	
1	U046	2023-06-19	2	5.7	Good	5.5	36	
2	U015	2023-06-03	3	7.5	Good	1.0	7	
3	U014	2023-06-24	4	5.6	Good	3.7	35	
4	U016	2023-06-02	10	7.3	Excellent	5.4	29	

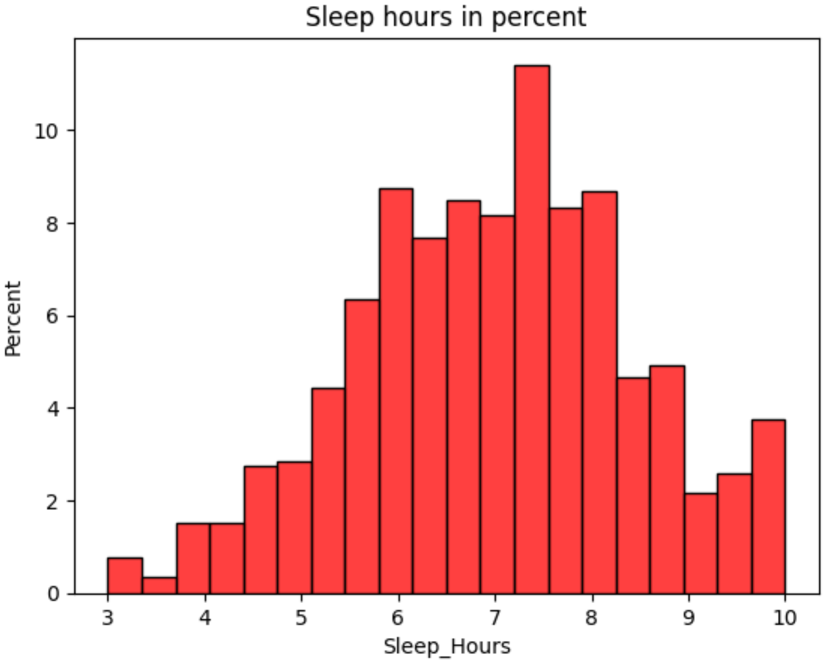
Graphical representation with Python libraries

Distribution plots and categorical plots

1a.Histplot with percent

In [7]:

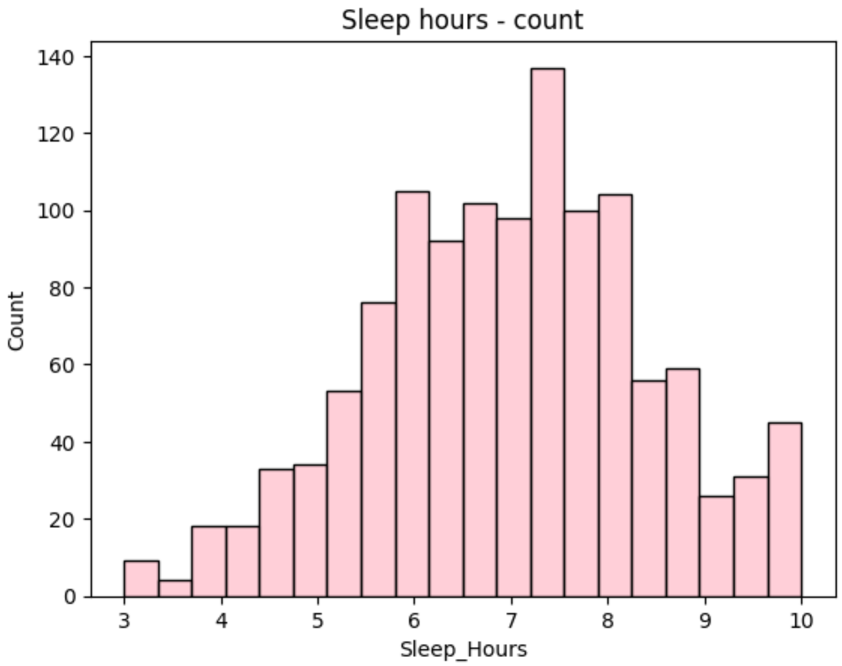
```
#1a.Histplot with percent
sns.histplot(data=df_final_kodiert,x="Sleep_Hours",stat="percent", color="red")
plt.title("Sleep hours in percent")
plt.show()
```



1b.Histplot with count

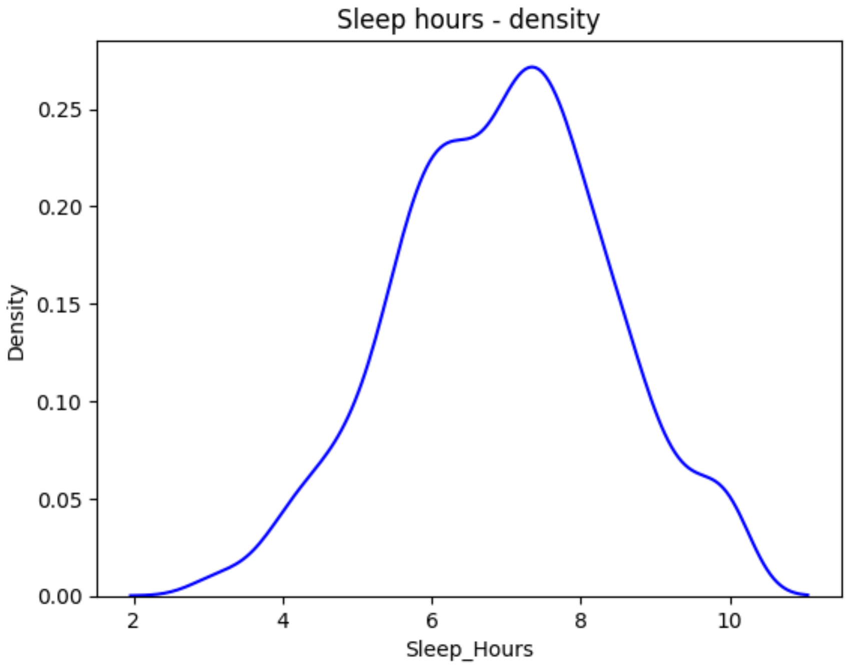
In [8]:

```
#1b.Histplot with count
sns.histplot(data=df_final_kodiert,x="Sleep_Hours", color="pink")
plt.title("Sleep hours - count")
plt.show()
```



2. Kdeplot

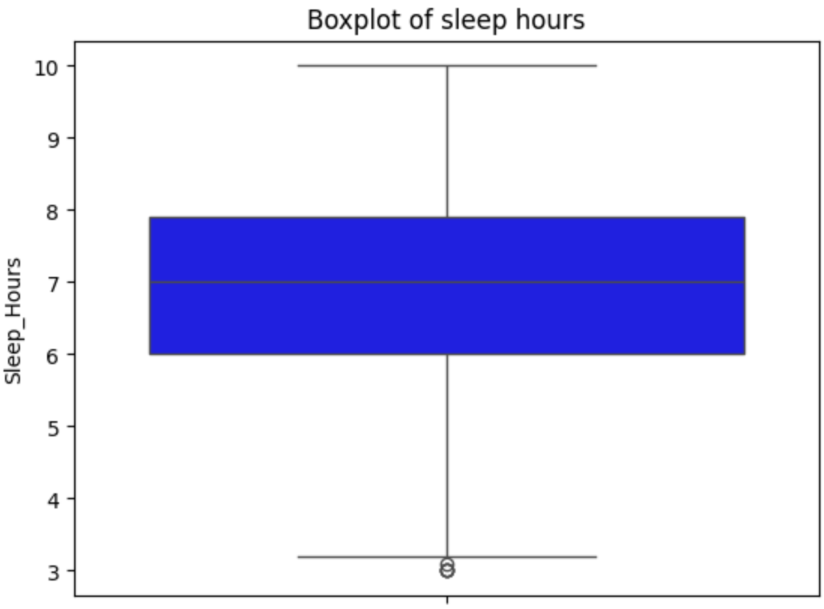
In [9]:
```
sns.kdeplot(data=df_final_kodiert,x="Sleep_Hours", color="blue")
plt.title("Sleep hours - density")
plt.show()
```



3a. Boxplot

In [12]:
```
sns.boxplot(data=df_final_kodiert,y="Sleep_Hours", color="blue")
plt.title("Boxplot of sleep hours")
plt.show()
```

```
c:\Users\goex1\AppData\Local\Programs\Python\Python313\Lib\site-packages\seaborn\categorical.py:640: FutureWarnin
g: SeriesGroupBy.grouper is deprecated and will be removed in a future version of pandas.
  positions = grouped.grouper.result_index.to_numpy(dtype=float)
```

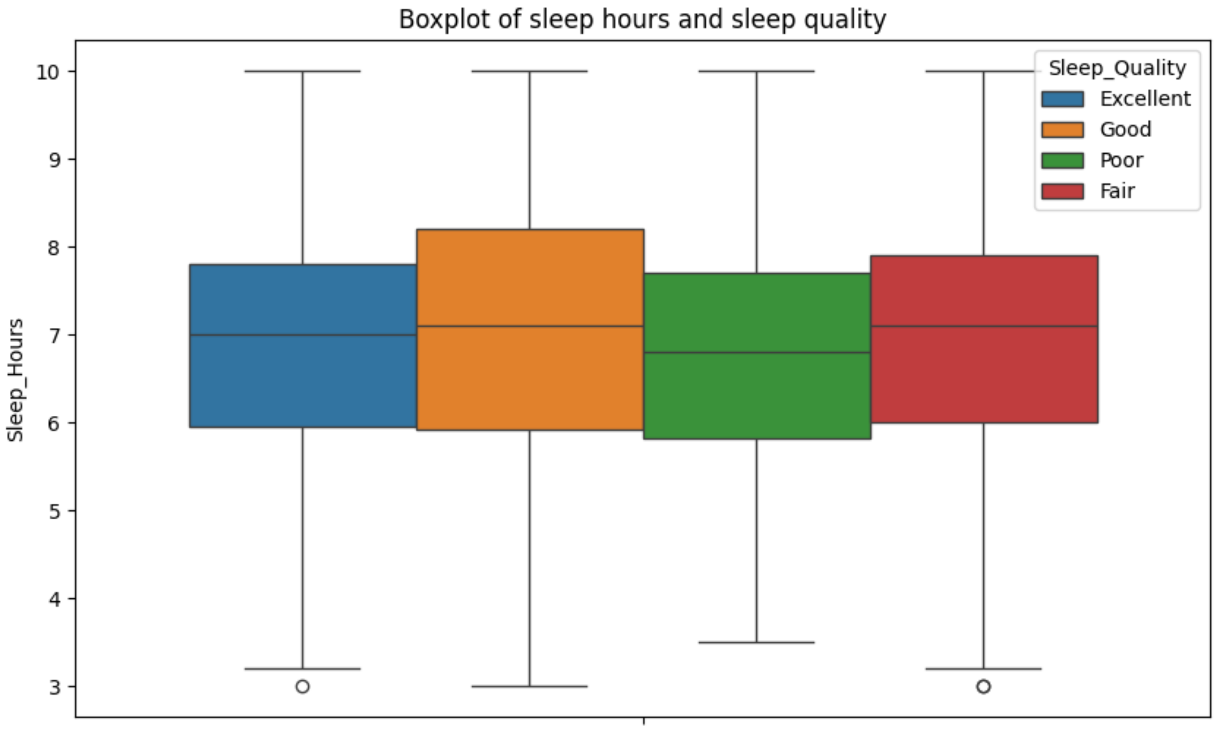


3b. Boxplot with hue

In [19]:

```
plt.figure(figsize=(10,6))
sns.boxplot(data=df_train,y="Sleep_Hours", hue="Sleep_Quality")
plt.title("Boxplot of sleep hours and sleep quality")
plt.show()
```

```
c:\Users\goex1\AppData\Local\Programs\Python\Python313\Lib\site-packages\seaborn\_base.py:949: FutureWarning: Whe
n grouping with a length-1 list-like, you will need to pass a length-1 tuple to get_group in a future version of
pandas. Pass `(name,)` instead of `name` to silence this warning.
  data_subset = grouped_data.get_group(pd_key)
c:\Users\goex1\AppData\Local\Programs\Python\Python313\Lib\site-packages\seaborn\categorical.py:640: FutureWarnin
g: SeriesGroupBy.grouper is deprecated and will be removed in a future version of pandas.
  positions = grouped.grouper.result_index.to_numpy(dtype=float)
c:\Users\goex1\AppData\Local\Programs\Python\Python313\Lib\site-packages\seaborn\_base.py:949: FutureWarning: Whe
n grouping with a length-1 list-like, you will need to pass a length-1 tuple to get_group in a future version of
pandas. Pass `(name,)` instead of `name` to silence this warning.
  data_subset = grouped_data.get_group(pd_key)
c:\Users\goex1\AppData\Local\Programs\Python\Python313\Lib\site-packages\seaborn\categorical.py:640: FutureWarnin
g: SeriesGroupBy.grouper is deprecated and will be removed in a future version of pandas.
  positions = grouped.grouper.result_index.to_numpy(dtype=float)
c:\Users\goex1\AppData\Local\Programs\Python\Python313\Lib\site-packages\seaborn\_base.py:949: FutureWarning: Whe
n grouping with a length-1 list-like, you will need to pass a length-1 tuple to get_group in a future version of
pandas. Pass `(name,)` instead of `name` to silence this warning.
  data_subset = grouped_data.get_group(pd_key)
c:\Users\goex1\AppData\Local\Programs\Python\Python313\Lib\site-packages\seaborn\categorical.py:640: FutureWarnin
g: SeriesGroupBy.grouper is deprecated and will be removed in a future version of pandas.
  positions = grouped.grouper.result_index.to_numpy(dtype=float)
c:\Users\goex1\AppData\Local\Programs\Python\Python313\Lib\site-packages\seaborn\_base.py:949: FutureWarning: Whe
n grouping with a length-1 list-like, you will need to pass a length-1 tuple to get_group in a future version of
pandas. Pass `(name,)` instead of `name` to silence this warning.
  data_subset = grouped_data.get_group(pd_key)
c:\Users\goex1\AppData\Local\Programs\Python\Python313\Lib\site-packages\seaborn\categorical.py:640: FutureWarnin
g: SeriesGroupBy.grouper is deprecated and will be removed in a future version of pandas.
  positions = grouped.grouper.result_index.to_numpy(dtype=float)
```

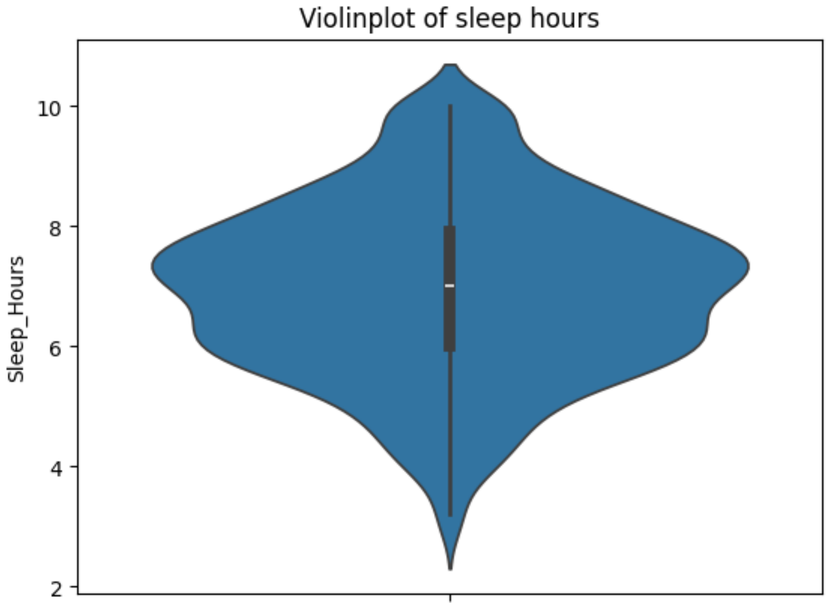


4a. Violinplot

In [ ]:

```
sns.violinplot(data=df_train,y="Sleep_Hours")
plt.title("Violinplot of sleep hours")
plt.show()
```

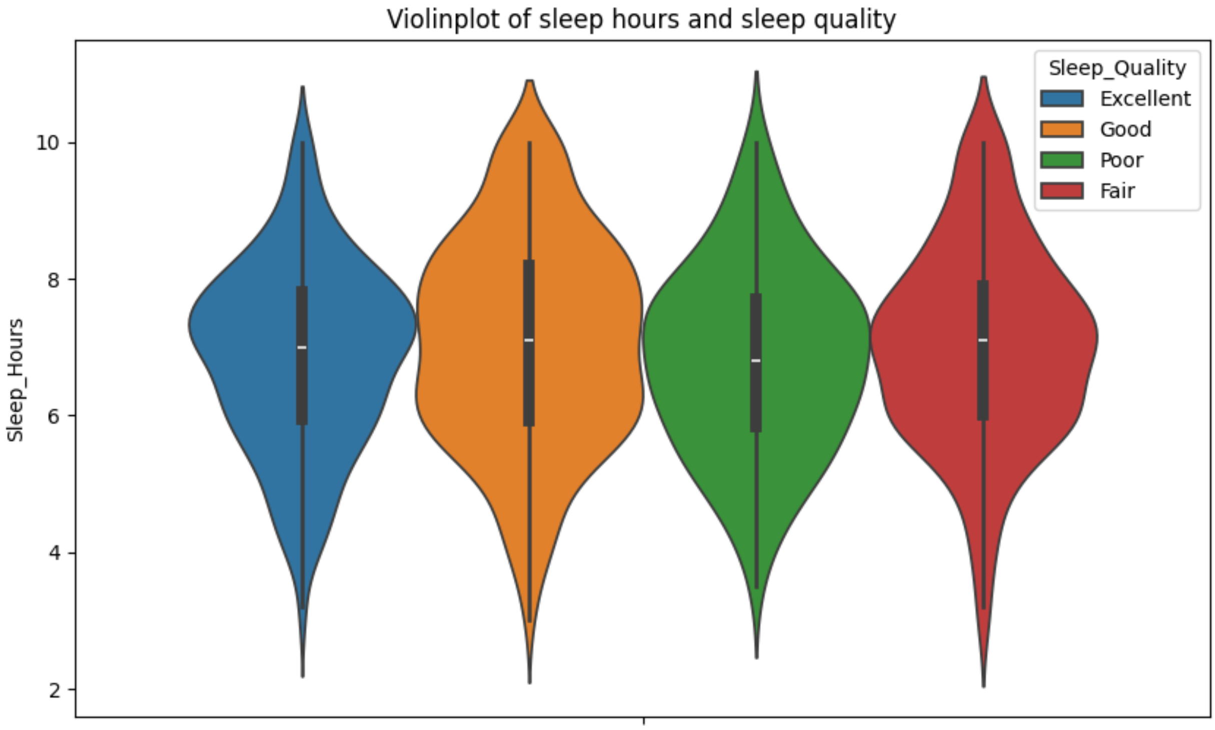
```
c:\Users\goex1\AppData\Local\Programs\Python\Python313\Lib\site-packages\seaborn\_base.py:949: FutureWarning: Whe
n grouping with a length-1 list-like, you will need to pass a length-1 tuple to get_group in a future version of
pandas. Pass `(name,)` instead of `name` to silence this warning.
  data_subset = grouped_data.get_group(pd_key)
```



4b. Violinplot with hue

In [17]:

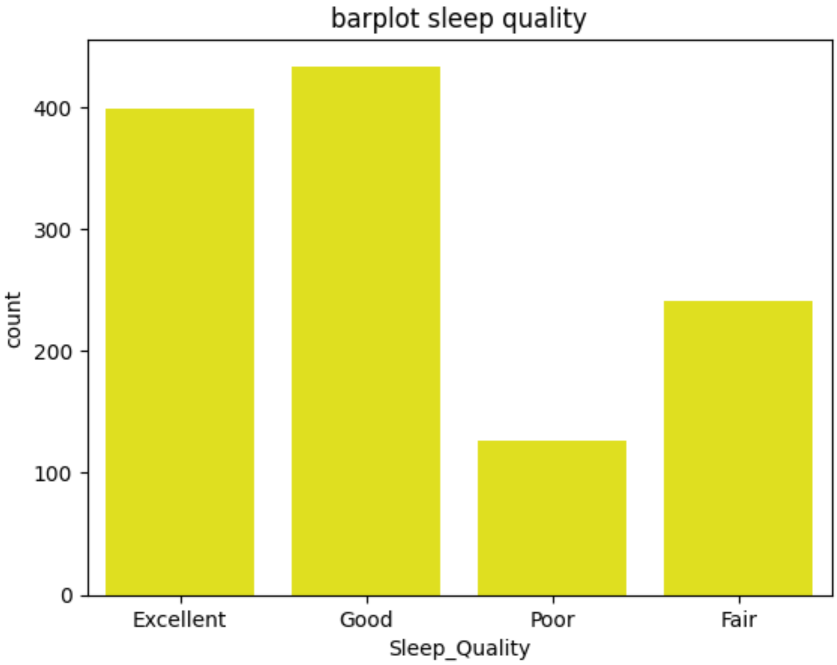
```
plt.figure(figsize=(10,6))
sns.violinplot(data=df_train,y="Sleep_Hours", hue="Sleep_Quality")
plt.title("Violinplot of sleep hours and sleep quality")
plt.show()
```



5a. Barplot

In [ ]:

```
sns.countplot(data=df_train,x="Sleep_Quality",color="yellow")
plt.title("barplot sleep quality")
plt.show()
```



5b. Barplot and hue

In [ ]:

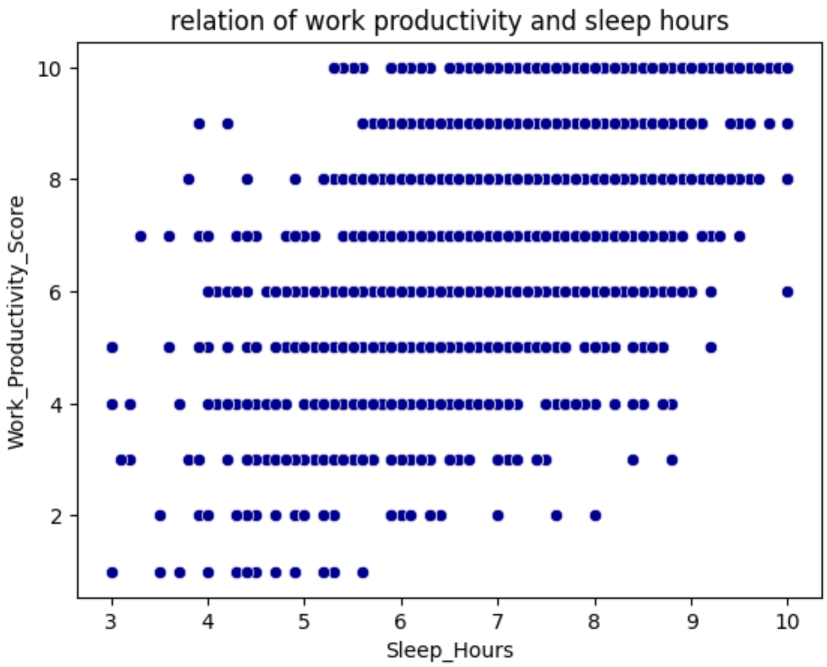
```
plt.figure(figsize=(10,6))
plt.title("barplot mean sleep hours and sleep quality")
sns.barplot(data=df_train,y="Sleep_Hours", hue="Sleep_Quality")
plt.ylabel("mean sleep hours")
plt.show()
```

Graphics for bivariate relationships

1. Scatterplot

In [18]:

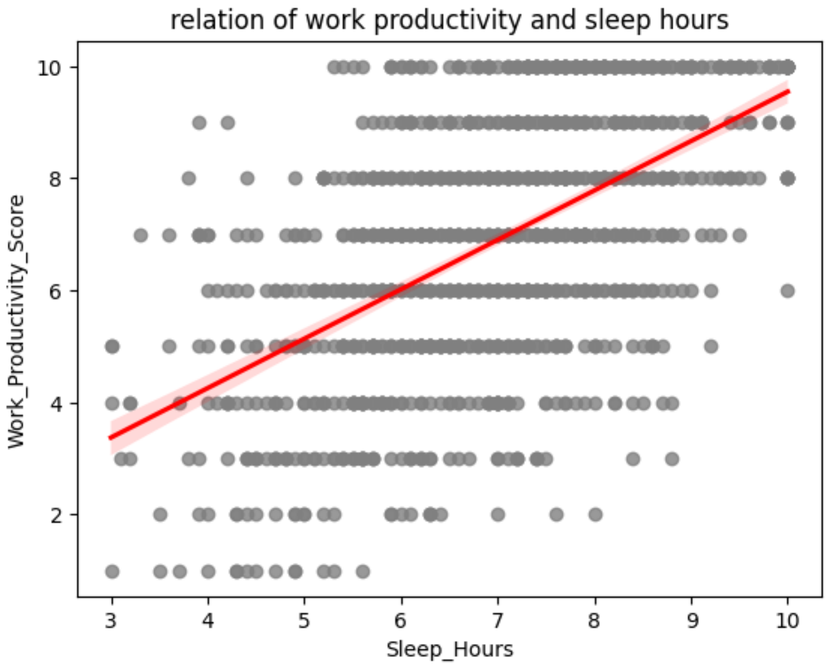
```
sns.scatterplot(data=df_train,x="Sleep_Hours",y="Work_Productivity_Score",color="darkblue")
plt.title("relation of work productivity and sleep hours")
plt.show()
```



2. Regplot

In [19]:

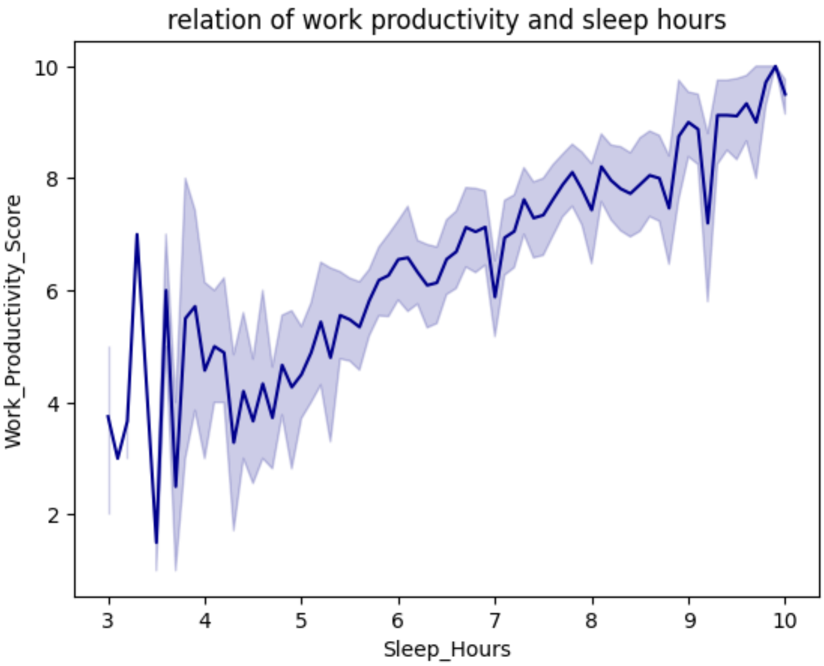
```
sns.regplot(data=df_train,
            x="Sleep_Hours",
            y="Work_Productivity_Score",
            scatter_kws={"color": "grey"},
            line_kws={"color": "red"})
plt.title("relation of work productivity and sleep hours")
plt.show()
```



3. Lineplot

In [20]:

```
sns.lineplot(data=df_train,x="Sleep_Hours",y="Work_Productivity_Score",color="darkblue")
plt.title("relation of work productivity and sleep hours")
plt.show()
```

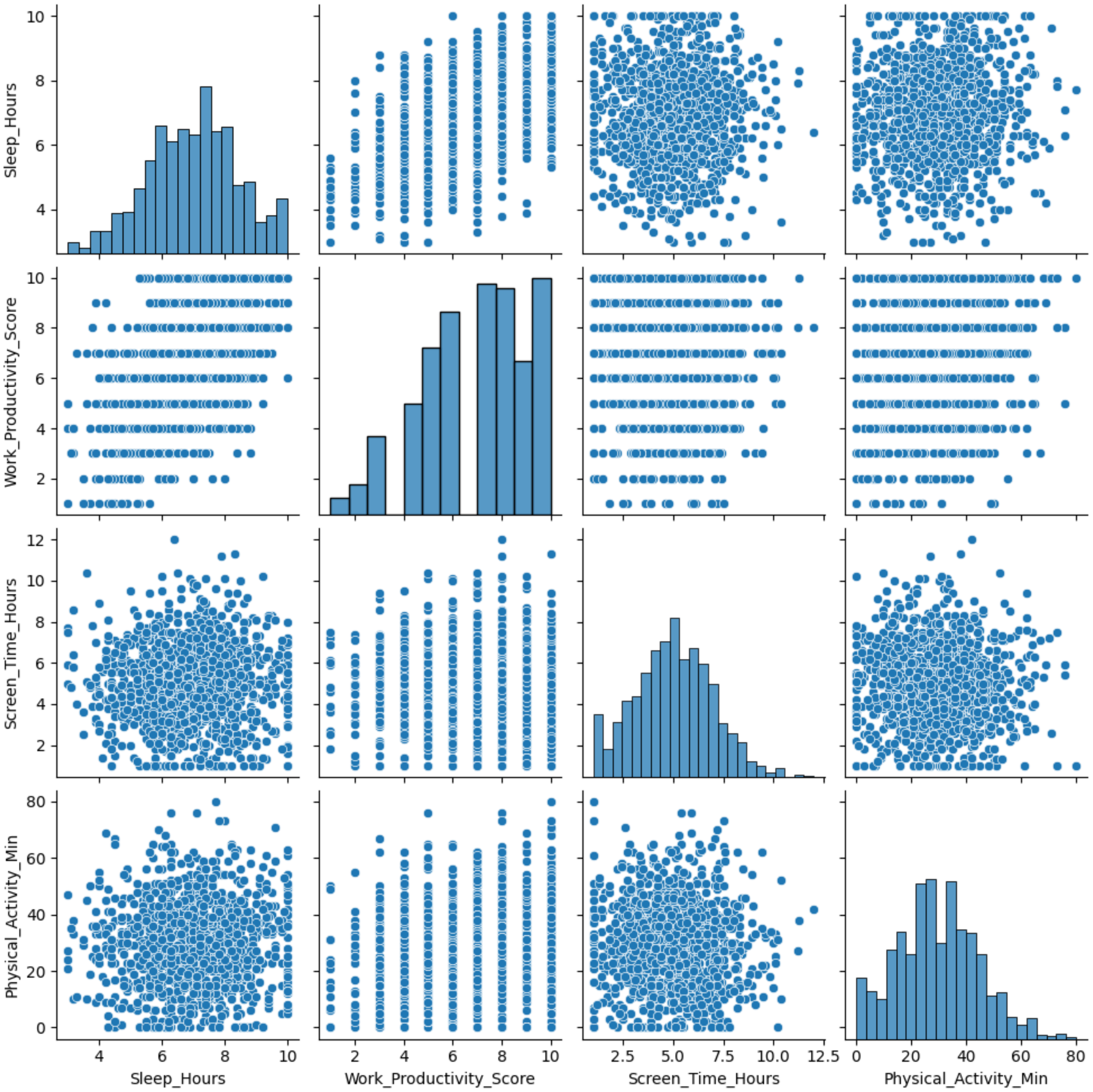


Multidimensional visualization

1. Pairplot

In [21]:

```
relevant_variables=["Sleep_Hours", "Work_Productivity_Score","Screen_Time_Hours","Physical_Activity_Min"]
sns.pairplot(data=df_train, vars=relevant_variables)
plt.show()
```


Sleep_Hours
Work_Productivity_Score
Screen_Time_Hours
Physical_Activity_Min
4
6
8
10
2
12
0
20
40
60
80
2.5
5.0
7.5
10.0
12.5
Sleep_Hours
Work_Productivity_Score
Screen_Time_Hours
Physical_Activity_Min