

```
In [70]: import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns

In [71]: data = pd.read_csv("main_data.csv")

In [72]: pre_event_data = data.iloc[:, 2:9]
post_event_data = pd.concat([data.iloc[:, 2:5], data.iloc[:, 9:]], axis = 1)

In [73]: post_event_data = post_event_data.rename(columns = {"Physical Health.1": "Physical Health",
                                                            "Social Relationships.1": "Social Relationships"})

In [74]: Gender = {0: "Male", 1: "Female"}
Marital_status = {0: "Unmarried", 1: "Married", 2: "Separated", 3: "Widow", 4: "Hermit"}

In [75]: plt.figure(figsize=(12, 8))

plt.subplot(2, 2, 1)
Marital_status_data = pre_event_data["Marital Status"].value_counts()
Labels = [Marital_status[x] for x in Marital_status_data.keys()]
plt.pie(Marital_status_data.values, labels=Labels, autopct='%1.1f%%')
plt.title("Distribution of data on the basis of marital status")
plt.legend(loc='upper right')

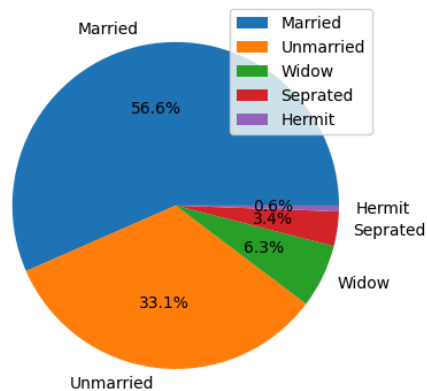
plt.subplot(2, 2, 2)
gender_data = pre_event_data["Gender"].value_counts()
Labels = [Gender[x] for x in gender_data.keys()]
plt.pie(gender_data.values, labels=Labels, autopct='%1.1f%%')
plt.title("Distribution of data on the basis of Gender")
plt.legend(loc='upper right')

plt.subplot(2, 1, 2)
age_data = {"Young": 0, "Middle_age": 0, "Senior_citizen": 0}
for age in pre_event_data["Age"]:
    if age >= 18 and age < 30: age_data["Young"] += 1
    elif age >= 30 and age < 50: age_data["Middle_age"] += 1
    else: age_data["Senior_citizen"] += 1

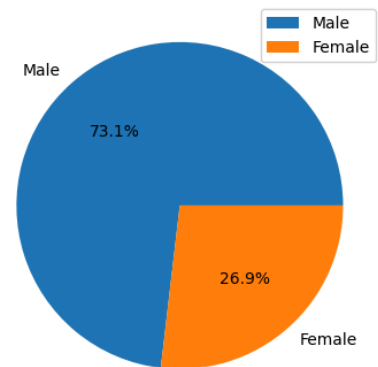
plt.pie(age_data.values(), labels=age_data.keys(), autopct='%1.1f%%')
plt.title("Distribution of data on the basis of Gender")
plt.legend(loc='upper right')

plt.tight_layout()
plt.show()
```

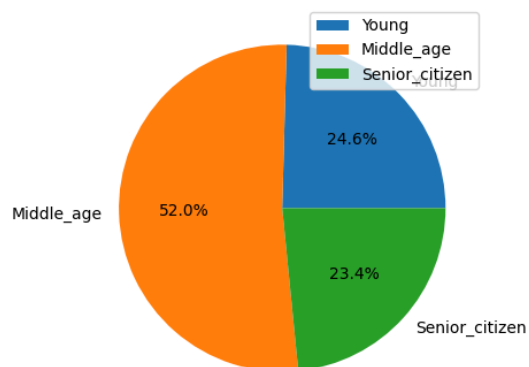
Distribution of data on the basis of marital status



Distribution of data on the basis of Gender



Distribution of data on the basis of Gender



In []:

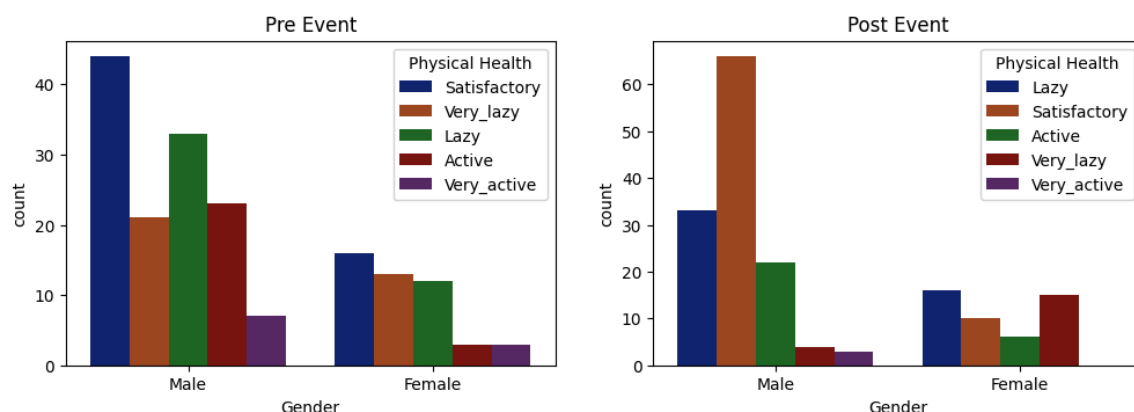
In []:

```
In [76]: def physical_health_div(mode,column):
physical_health = []
if mode == "pre": iterator = pre_event_data[column]
if mode == "post": iterator = post_event_data[column]
for points in iterator:
    if points>=0 and points<25: physical_health.append("Very_lazy")
    elif points>=25 and points<30: physical_health.append('Lazy')
    elif points>=30 and points<35: physical_health.append('Satisfactory')
    elif points>=35 and points<40: physical_health.append('Active')
    else: physical_health.append('Very_active')
return physical_health
```

```
In [77]: plt.figure(figsize=(12, 8))
plt.subplot(2, 2, 1)
physical_health_data_pre = {"Gender": pre_event_data["Gender"],"Physical Health"
physical_health_data_pre = pd.DataFrame(physical_health_data_pre)
fig = sns.countplot(x="Gender", data=physical_health_data_pre, palette='dark',hu
fig.set_title('Pre Event')
fig.set_xticklabels(["Male","Female"])

plt.subplot(2, 2, 2)
physical_health_data_post = {"Gender": post_event_data["Gender"],"Physical Healt
physical_health_data_post = pd.DataFrame(physical_health_data_post)
fig = sns.countplot(x="Gender", data=physical_health_data_post, palette='dark',h
fig.set_title('Post Event')
fig.set_xticklabels(["Male","Female"])
```

Out[77]: [Text(0, 0, 'Male'), Text(1, 0, 'Female')]



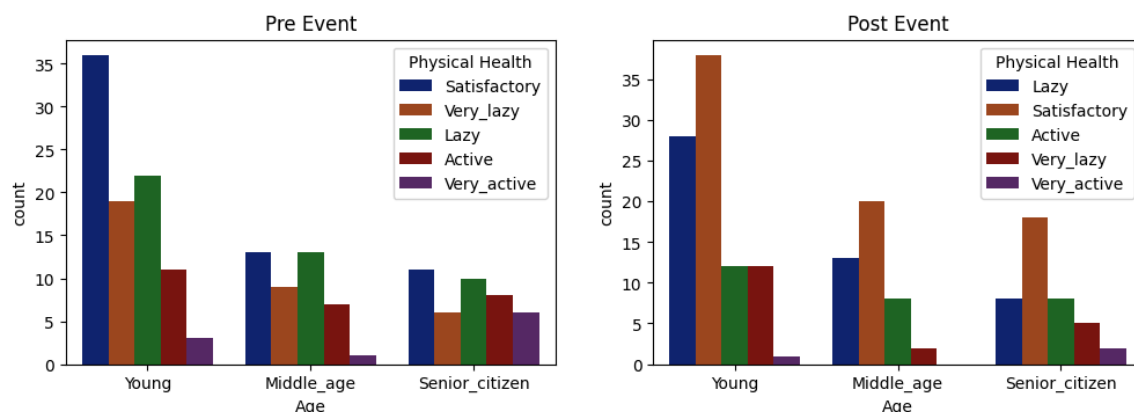
```
In [78]: age_div = []
for age in pre_event_data["Age"]:
    if age>=18 and age<30: age_div.append("Young_age")
    elif age>=30 and age<50: age_div.append("Middle_age")
    else: age_div.append("Senior_citizen")

plt.figure(figsize=(12, 8))

plt.subplot(2, 2, 1)
physical_health_data_pre = {"Age":age_div,"Physical Health":physical_health_div(
physical_health_data_pre = pd.DataFrame(physical_health_data_pre)
fig = sns.countplot(x="Age", data=physical_health_data_pre, palette='dark',hue='
fig.set_title('Pre Event')
fig.set_xticklabels(["Young","Middle_age","Senior_citizen"])

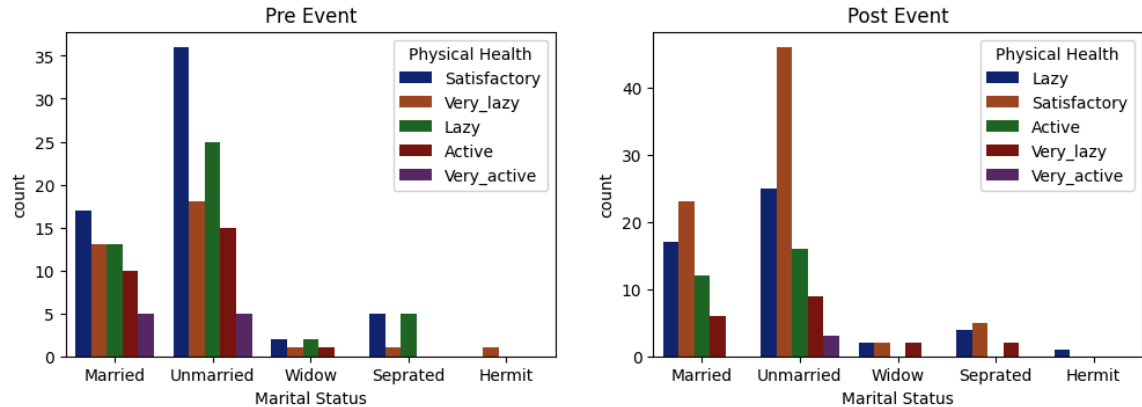
plt.subplot(2, 2, 2)
physical_health_data_post = {"Age": age_div,"Physical Health":physical_health_di
physical_health_data_post = pd.DataFrame(physical_health_data_post)
fig = sns.countplot(x="Age", data=physical_health_data_post, palette='dark',hue=
fig.set_title('Post Event')
fig.set_xticklabels(["Young","Middle_age","Senior_citizen"])
```

Out[78]: [Text(0, 0, 'Young'), Text(1, 0, 'Middle_age'), Text(2, 0, 'Senior_citizen')]



```
In [79]: plt.figure(figsize=(12, 8))
plt.subplot(2, 2, 1)
physical_health_data_pre = {"Marital Status":pre_event_data["Marital Status"],"P
physical_health_data_pre = pd.DataFrame(physical_health_data_pre)
fig = sns.countplot(x="Marital Status", data=physical_health_data_pre, palette='
fig.set_title('Pre Event')
fig.set_xticklabels(["Married","Unmarried","Widow","Seprated","Hermit"])
```

```
plt.subplot(2, 2, 2)
physical_health_data_post = {"Marital Status":post_event_data["Marital Status"],
physical_health_data_post = pd.DataFrame(physical_health_data_post)
fig = sns.countplot(x="Marital Status", data=physical_health_data_post, palette=
fig.set_title('Post Event')
fig.set_xticklabels(["Married","Unmarried","Widow","Seprated","Hermit"])
plt.show()
```



In []:

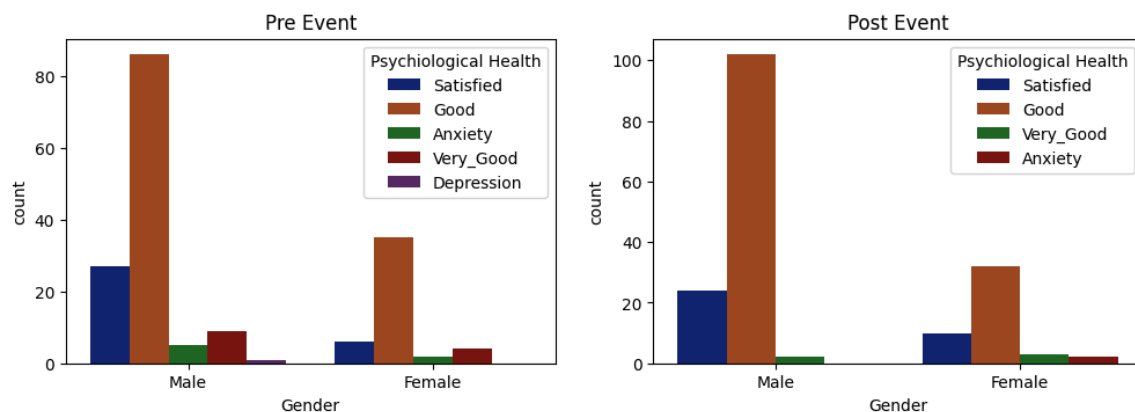
In []:

```
In [80]: def psychological_health_div(mode,column):
psychological_health = []
if mode == "pre": iterator = pre_event_data[column]
if mode == "post": iterator = post_event_data[column]
for points in iterator:
    if points>=0 and points<15: psychological_health.append("Depression")
    elif points>=15 and points<20: psychological_health.append('Anxiety')
    elif points>=20 and points<25: psychological_health.append('Satisfied')
    elif points>=25 and points<35: psychological_health.append('Good')
    else: psychological_health.append('Very_Good')
return psychological_health
```

```
In [81]: plt.figure(figsize=(12, 8))
plt.subplot(2, 2, 1)
psychological_health_data_pre = {"Gender": pre_event_data["Gender"],"Psychiolog
psychological_health_data_pre = pd.DataFrame(psychological_health_data_pre)
fig = sns.countplot(x="Gender", data=psychological_health_data_pre, palette='da
fig.set_title('Pre Event')
fig.set_xticklabels(["Male","Female"])

plt.subplot(2, 2, 2)
psychological_health_data_post = {"Gender": post_event_data["Gender"],"Psychiol
psychological_health_data_post = pd.DataFrame(psychological_health_data_post)
fig = sns.countplot(x="Gender", data=psychological_health_data_post, palette='d
fig.set_title('Post Event')
fig.set_xticklabels(["Male","Female"])
```

Out[81]: [Text(0, 0, 'Male'), Text(1, 0, 'Female')]



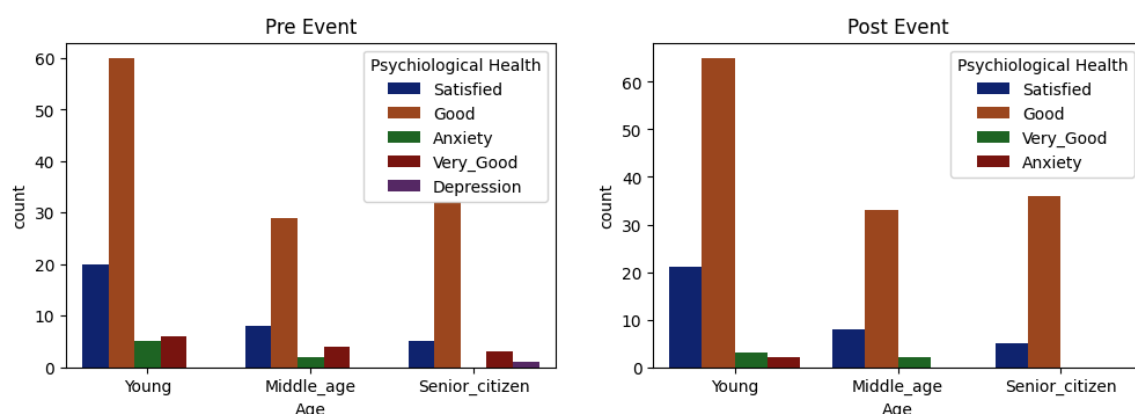
```
In [82]: age_div = []
for age in pre_event_data["Age"]:
    if age>=18 and age<30: age_div.append("Young_age")
    elif age>=30 and age<50: age_div.append("Middle_age")
    else: age_div.append("Senior_citizen")

plt.figure(figsize=(12, 8))

plt.subplot(2, 2, 1)
psychological_health_data_pre = {"Age":age_div,"Psychological Health":psychiol
psychological_health_data_pre = pd.DataFrame(psychological_health_data_pre)
fig = sns.countplot(x="Age", data=psychological_health_data_pre, palette='dark')
fig.set_title('Pre Event')
fig.set_xticklabels(["Young","Middle_age","Senior_citizen"])

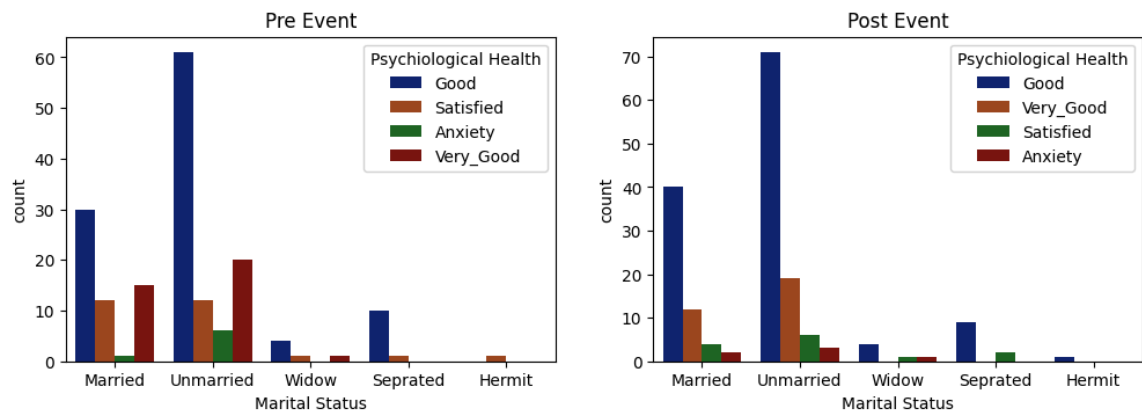
plt.subplot(2, 2, 2)
psychological_health_data_post = {"Age": age_div,"Psychological Health":psychi
psychological_health_data_post = pd.DataFrame(psychological_health_data_post)
fig = sns.countplot(x="Age", data=psychological_health_data_post, palette='dark')
fig.set_title('Post Event')
fig.set_xticklabels(["Young","Middle_age","Senior_citizen"])
```

```
Out[82]: [Text(0, 0, 'Young'), Text(1, 0, 'Middle_age'), Text(2, 0, 'Senior_citizen')]
```



```
In [83]: plt.figure(figsize=(12, 8))
plt.subplot(2, 2, 1)
psychological_health_data_pre = {"Marital Status":pre_event_data["Marital Statu
psychological_health_data_pre = pd.DataFrame(psychological_health_data_pre)
fig = sns.countplot(x="Marital Status", data=psychological_health_data_pre, pal
fig.set_title('Pre Event')
fig.set_xticklabels(["Married","Unmarried","Widow","Seprated","Hermit"])
```

```
plt.subplot(2, 2, 2)
psychological_health_data_post = {"Marital Status": post_event_data["Marital Sta
psychological_health_data_post = pd.DataFrame(psychological_health_data_post)
fig = sns.countplot(x="Marital Status", data=psychological_health_data_post, pa
fig.set_title('Post Event')
fig.set_xticklabels(["Married", "Unmarried", "Widow", "Seprated", "Hermit"])
plt.show()
```



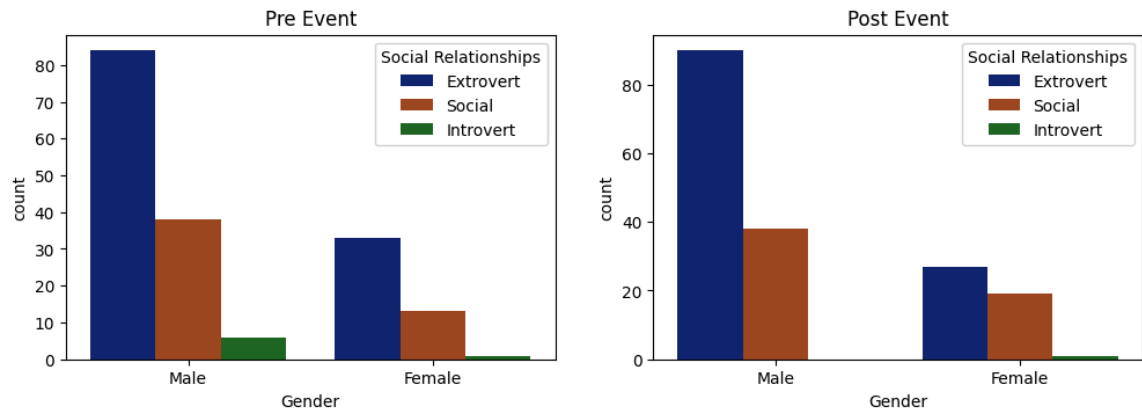
In []:

In []:

```
In [84]: def social_relationships_div(mode, column):
    social_relationships = []
    if mode == "pre": iterator = pre_event_data[column]
    if mode == "post": iterator = post_event_data[column]
    for points in iterator:
        if points >= 0 and points < 5: social_relationships.append("Introvert")
        elif points >= 5 and points < 10: social_relationships.append('Social')
        else: social_relationships.append('Extrovert')
    return social_relationships
```

```
In [85]: plt.figure(figsize=(12, 8))
plt.subplot(2, 2, 1)
social_relationships_data_pre = {"Gender": pre_event_data["Gender"], "Social Rela
social_relationships_data_pre = pd.DataFrame(social_relationships_data_pre)
fig = sns.countplot(x="Gender", data=social_relationships_data_pre, palette='dar
fig.set_title('Pre Event')
fig.set_xticklabels(["Male", "Female"])

plt.subplot(2, 2, 2)
social_relationships_data_post = {"Gender": post_event_data["Gender"], "Social Re
social_relationships_data_post = pd.DataFrame(social_relationships_data_post)
fig = sns.countplot(x="Gender", data=social_relationships_data_post, palette='da
fig.set_title('Post Event')
fig.set_xticklabels(["Male", "Female"])
plt.show()
```

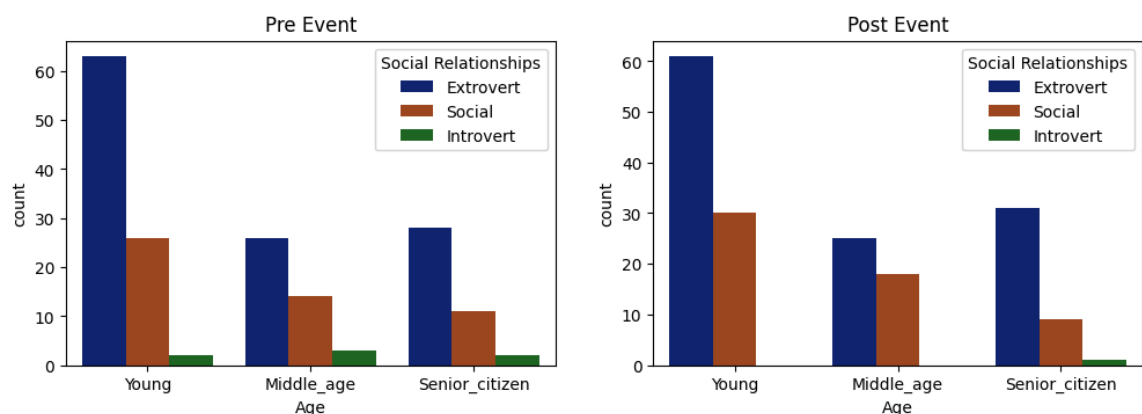


```
In [86]: age_div = []
for age in pre_event_data["Age"]:
    if age>=18 and age<30: age_div.append("Young_age")
    elif age>=30 and age<50: age_div.append("Middle_age")
    else: age_div.append("Senior_citizen")

plt.figure(figsize=(12, 8))

plt.subplot(2, 2, 1)
social_relationships_data_pre = {"Age":age_div,"Social Relationships":social_rel
social_relationships_data_pre = pd.DataFrame(social_relationships_data_pre)
fig = sns.countplot(x="Age", data=social_relationships_data_pre, palette='dark',
fig.set_title('Pre Event')
fig.set_xticklabels(["Young","Middle_age","Senior_citizen"])

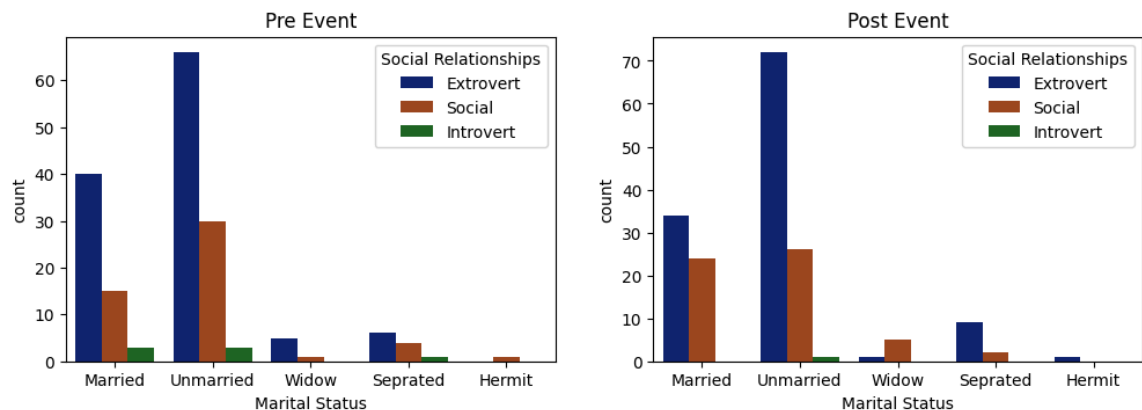
plt.subplot(2, 2, 2)
social_relationships_data_post = {"Age": age_div,"Social Relationships":social_r
social_relationships_data_post = pd.DataFrame(social_relationships_data_post)
fig = sns.countplot(x="Age", data=social_relationships_data_post, palette='dark'
fig.set_title('Post Event')
fig.set_xticklabels(["Young","Middle_age","Senior_citizen"])
plt.show()
```



```
In [87]: plt.figure(figsize=(12, 8))
plt.subplot(2, 2, 1)
social_relationships_data_pre = {"Marital Status":pre_event_data["Marital Status
social_relationships_data_pre = pd.DataFrame(social_relationships_data_pre)
fig = sns.countplot(x="Marital Status", data=social_relationships_data_pre, pale
fig.set_title('Pre Event')
fig.set_xticklabels(["Married","Unmarried","Widow","Seprated","Hermit"])

plt.subplot(2, 2, 2)
```

```
social_relationships_data_post = {"Marital Status":post_event_data["Marital Stat
social_relationships_data_post = pd.DataFrame(social_relationships_data_post)
fig = sns.countplot(x="Marital Status", data=social_relationships_data_post, pal
fig.set_title('Post Event')
fig.set_xticklabels(["Married","Unmarried","Widow","Seprated","Hermit"])
plt.show()
```



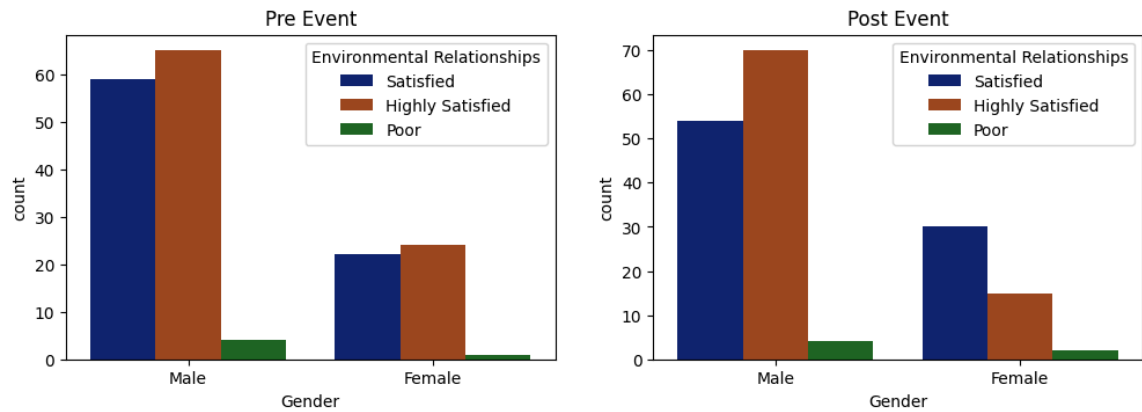
In []:

In []:

```
In [88]: def environmental_relationships_div(mode,column):
environmental_relationships = []
if mode == "pre": iterator = pre_event_data[column]
if mode == "post": iterator = post_event_data[column]
for points in iterator:
    if points>=0 and points<10: environmental_relationships.append("Poor")
    elif points>=10 and points<20: environmental_relationships.append('Satis
    else: environmental_relationships.append('Highly Satisfied')
return environmental_relationships
```

```
In [89]: plt.figure(figsize=(12, 8))
plt.subplot(2, 2, 1)
environmental_relationships_data_pre = {"Gender": pre_event_data["Gender"],"Envi
environmental_relationships_data_pre = pd.DataFrame(environmental_relationships_
fig = sns.countplot(x="Gender", data=environmental_relationships_data_pre, palet
fig.set_title('Pre Event')
fig.set_xticklabels(["Male","Female"])

plt.subplot(2, 2, 2)
environmental_relationships_data_post = {"Gender": post_event_data["Gender"],"En
environmental_relationships_data_post = pd.DataFrame(environmental_relationships_
fig = sns.countplot(x="Gender", data=environmental_relationships_data_post, pale
fig.set_title('Post Event')
fig.set_xticklabels(["Male","Female"])
plt.show()
```

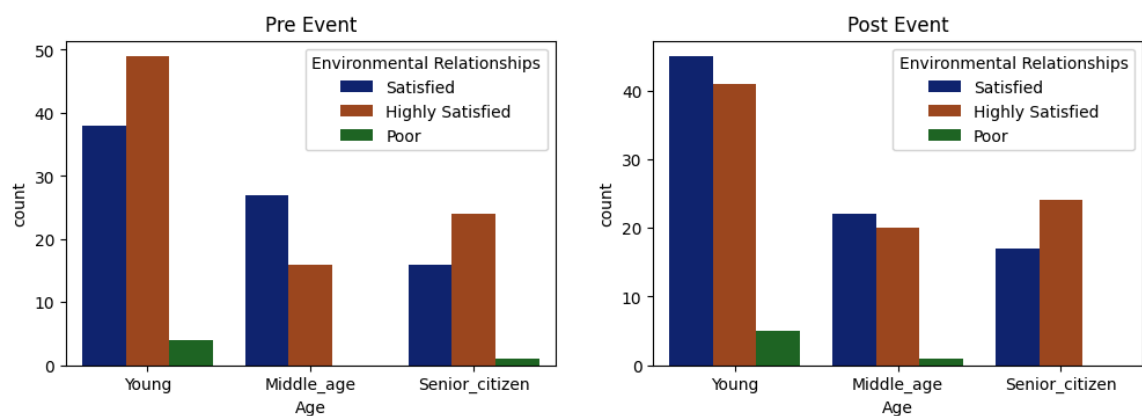



```
In [90]: age_div = []
for age in pre_event_data["Age"]:
    if age>=18 and age<30: age_div.append("Young_age")
    elif age>=30 and age<50: age_div.append("Middle_age")
    else: age_div.append("Senior_citizen")

plt.figure(figsize=(12, 8))

plt.subplot(2, 2, 1)
environmental_relationships_data_pre = {"Age":age_div,"Environmental Relationships":pre_event_data["Environmental Relationships"]}
environmental_relationships_data_pre = pd.DataFrame(environmental_relationships_data_pre)
fig = sns.countplot(x="Age", data=environmental_relationships_data_pre, palette="magma")
fig.set_title('Pre Event')
fig.set_xticklabels(["Young", "Middle_age", "Senior_citizen"])

plt.subplot(2, 2, 2)
environmental_relationships_data_post = {"Age": age_div,"Environmental Relationships":post_event_data["Environmental Relationships"]}
environmental_relationships_data_post = pd.DataFrame(environmental_relationships_data_post)
fig = sns.countplot(x="Age", data=environmental_relationships_data_post, palette="magma")
fig.set_title('Post Event')
fig.set_xticklabels(["Young", "Middle_age", "Senior_citizen"])
plt.show()
```



```
In [91]: plt.figure(figsize=(12, 8))
plt.subplot(2, 2, 1)
environmental_relationships_data_pre = {"Marital Status":pre_event_data["Marital Status"],"Environmental Relationships":pre_event_data["Environmental Relationships"]}
environmental_relationships_data_pre = pd.DataFrame(environmental_relationships_data_pre)
fig = sns.countplot(x="Marital Status", data=environmental_relationships_data_pre, palette="magma")
fig.set_title('Pre Event')
fig.set_xticklabels(["Married", "Unmarried", "Widow", "Seprated", "Hermit"])

plt.subplot(2, 2, 2)
```

```
environmental_relationships_data_post = {"Marital Status":post_event_data["Marital Status"], "Environmental Relationships":post_event_data["Environmental Relationships"]}
environmental_relationships_data_post = pd.DataFrame(environmental_relationships_data_post)
fig = sns.countplot(x="Marital Status", data=environmental_relationships_data_post)
fig.set_title('Post Event')
fig.set_xticklabels(["Married", "Unmarried", "Widow", "Seprated", "Hermit"])
plt.show()
```

