Import Libraries

- sea born

What is sea born used for? Seaborn is an open-source Python library built on top of matplotlib. It is used for data visualization and exploratory data analysis.

- pandas

pandas is a software library written for the Python programming language for data manipulation and analysis

- matplot lib

Matplotlib is a cross-platform, data visualization and graphical plotting library for Python and its numerical extension NumPy.

In [1]:
```
import pandas as pd
```

In [2]:
```
df = pd.read_csv(r"C:\Users\Asad\Desktop\Data_Assignment\assignment_data.csv")
```

In [3]:
```
print(df)
```

```
     Gender  Location    Age Qualification_completed    field_of_study  \
0      Male  Pakistan  36-40                Masters  Natural Sciences
1      Male  Pakistan  26-30              Bachelors             CS/IT
2      Male  Pakistan  31-35                Masters       Enginnering
3    Female  Pakistan  31-35                Masters             CS/IT
4    Female  Pakistan  26-30                Masters       Enginnering
..      ...       ...    ...                    ...               ...
370    Male  Pakistan  26-30                Masters       Enginnering
371    Male  Pakistan  31-35              Bachelors       Enginnering
372    Male  Pakistan  21-25              Bachelors             CS/IT
373    Male  Pakistan  26-30                Masters       Enginnering
374  Female  Pakistan  31-35                Masters       Mathematics

           Purpose_for_chilla What are you? Blood group   \
0       to boost my skill set     Unemplyed           B+
1       to boost my skill set       Student           B+
2    Switch my field of study      Employed           B+
3       to boost my skill set      Employed           O+
4       to boost my skill set       Student           A-
..                        ...           ...          ...
370     to boost my skill set      Employed           O+
371     to boost my skill set      Employed           A+
372     to boost my skill set      Employed           O+
373     to boost my skill set      Employed           B-
374  Switch my field of study     Unemplyed           B+

    Which mobile sim do you use Prepaid or Postpaid  ...  \
0                       U-fone             Prepaid  ...
1                       U-fone             Prepaid  ...
2                         Zong             Prepaid  ...
```

```
3                        U-fone            Postpaid  ...
4                      Mobilink             Prepaid  ...
..                          ...                 ...  ...
370                     Telenor             Prepaid  ...
371                        Zong            Postpaid  ...
372                    Mobilink             Prepaid  ...
373                    Mobilink             Prepaid  ...
374                     Telenor             Prepaid  ...

    Your favorite programming language? Marital Status? Are you Vaccinated?  \
0                                Python             Yes                  Yes
1                                Python              No                  Yes
2                                Python             Yes                  Yes
3                                Python             Yes                  Yes
4                            Javascript              No                  Yes
..                                  ...             ...                  ...
370                                   R             Yes                  Yes
371                              Python             Yes                  Yes
372                              Python              No                  Yes
373                              Python              No                   No
374                              Python             Yes                  Yes

    Where do you live? Research/Working experience (Float/Int) years  \
0               Urbun                                              5
1               Urbun                                              1
2               Urbun                                            5.5
3               Urbun                                              5
4               Rural                                            3.5
..                ...                                            ...
370             Rural                                              7
371             Urbun                                              5
372             Urbun                                              0
373             Urbun                                              2
374             Urbun                                              3

    Age (years)-Float/Int Your Weight in kg? (float)  \
0                   38.00                       77.0
1                   25.00                       53.6
2                   31.34                       93.0
3                   33.00                       60.0
4                   27.00                       59.9
..                    ...                        ...
370                 28.00                       70.5
371                 33.00                       83.4
372                 22.80                       60.0
373                 29.00                       86.0
374                 31.00                       54.5

    Height in cm? Freelancer- (Float)  \
0                            179.000
1                            178.000
2                            173.000
3                            157.000
4                            164.544
..                               ...
370                          178.500
371                          172.700
372                            1.680
373                          180.000
374                          161.544

     How many hours you code a day? (int) e.g: 5,4,3  \
0                                              3.0
1                                              2.0
2                                              2.0
```

```
3                                                   3.0
4                                                   6.0
..                                                  ...
370                                                 4.0
371                                                 1.0
372                                                 0.0
373                                                 2.0
374                                                 3.0

     Light kitni der band hti hy? int
0                                   2.0
1                                   6.0
2                                   0.0
3                                  24.0
4                                  12.0
..                                  ...
370                                 3.0
371                                 1.0
372                                 0.0
373                                 1.0
374                                 0.0
```

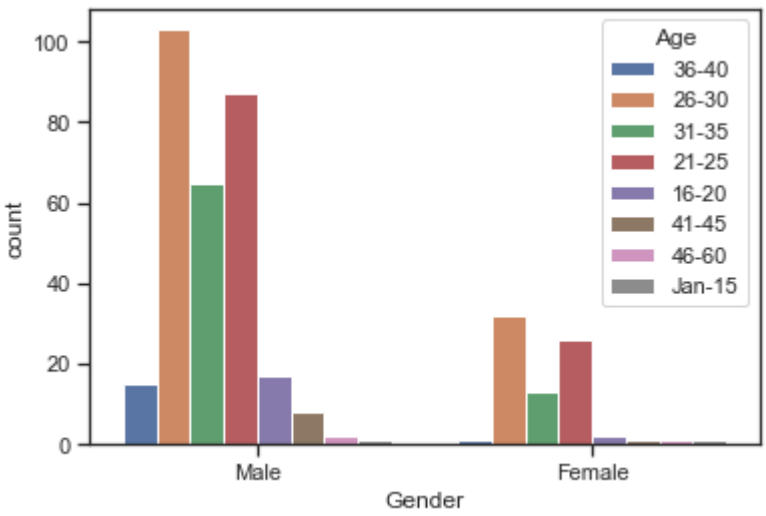
In [4]:
```
import seaborn as sns
```

In [5]:
```
import matplotlib.pyplot as plt
```

In [6]:
```
sns.set_theme(style="ticks" , color_codes=True)
```

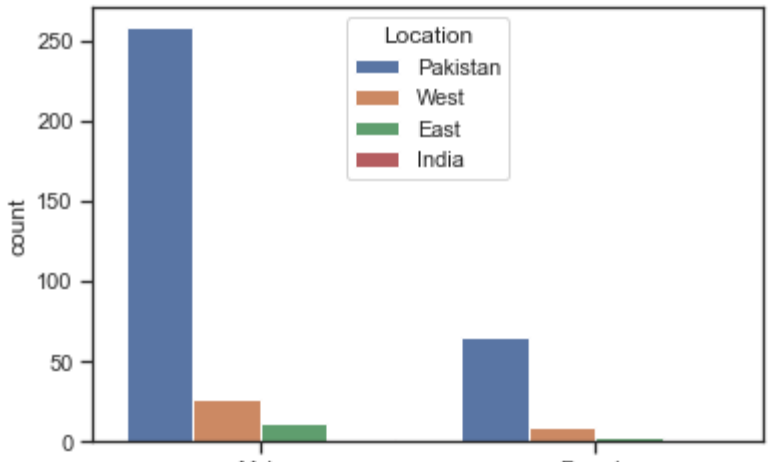
In [7]:
```
sns.countplot(x="Gender" , hue="Age" , data=df)
```

Out[7]: <AxesSubplot:xlabel='Gender', ylabel='count'>



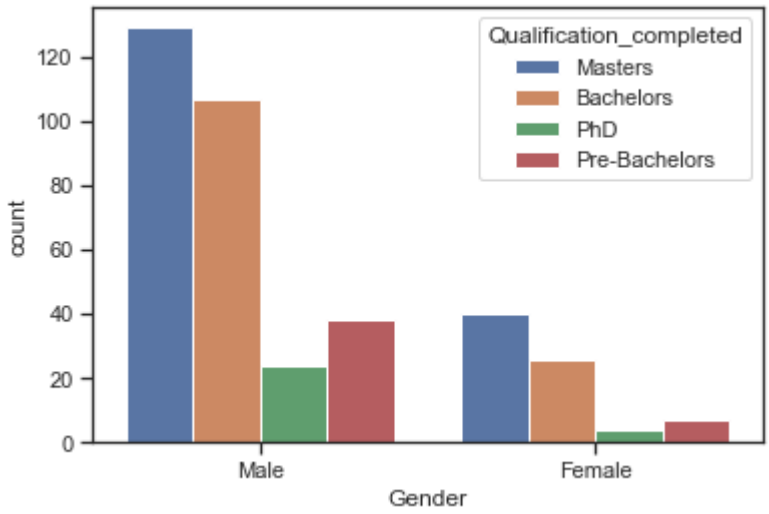
In [8]:
```
sns.countplot(x="Gender" , hue="Location" , data=df)
```

Out[8]: <AxesSubplot:xlabel='Gender', ylabel='count'>



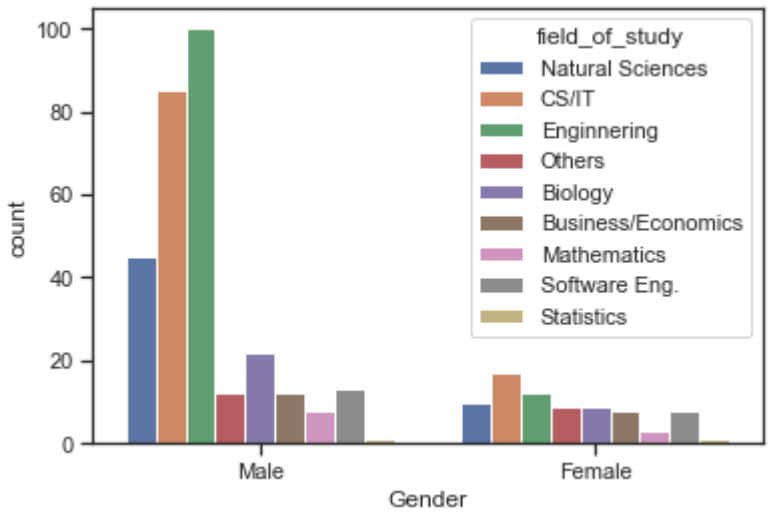
In [9]:
```
sns.countplot(x="Gender" , hue="Qualification_completed" , data=df)
```

Out[9]: <AxesSubplot:xlabel='Gender', ylabel='count'>



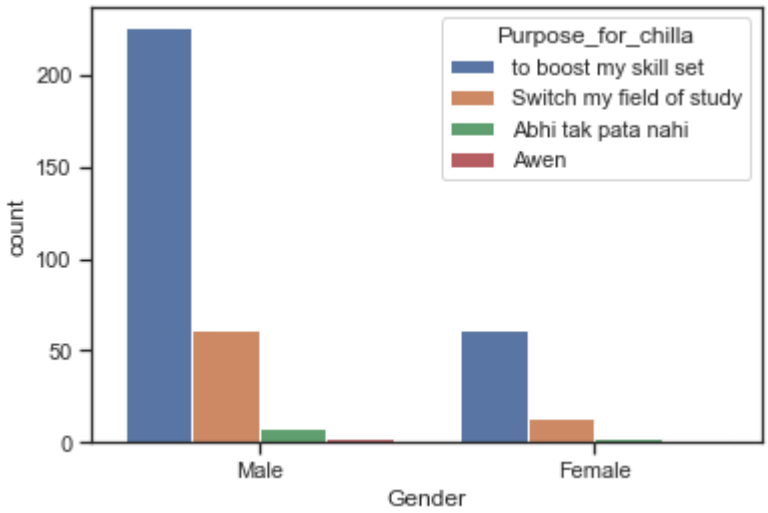
In [10]:
```
sns.countplot(x="Gender" , hue="field_of_study" , data=df)
```

Out[10]: <AxesSubplot:xlabel='Gender', ylabel='count'>



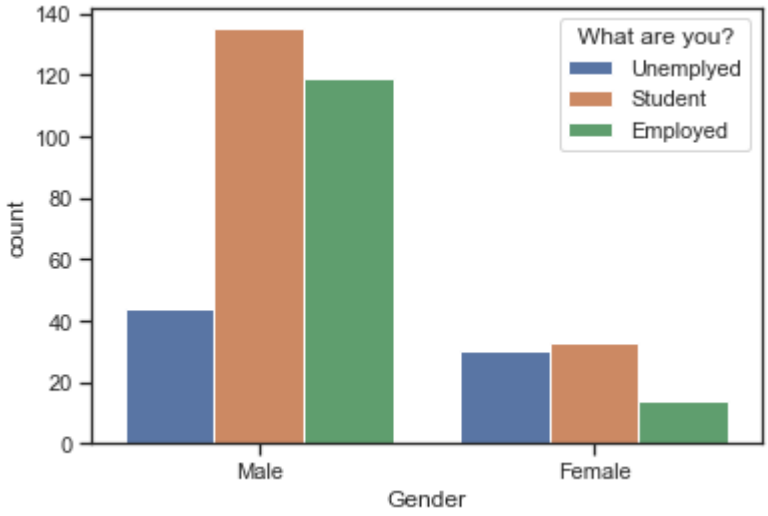
In [11]:
```
sns.countplot(x="Gender" , hue="Purpose_for_chilla" , data=df)
```

Out[11]: <AxesSubplot:xlabel='Gender', ylabel='count'>



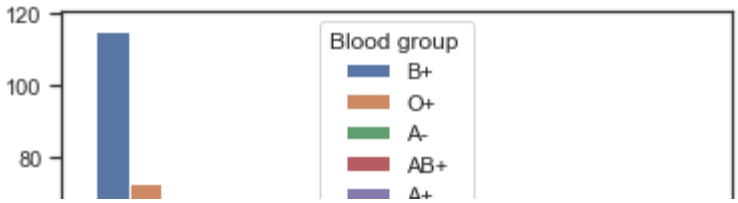
In [12]:
```
sns.countplot(x="Gender" , hue="What are you?" , data=df)
```

Out[12]: <AxesSubplot:xlabel='Gender', ylabel='count'>



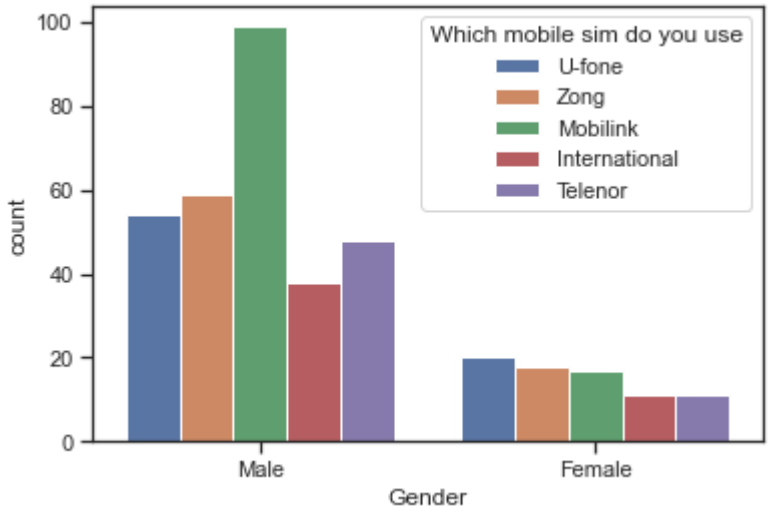
In [13]:
```
sns.countplot(x="Gender" , hue="Blood group " , data=df)
```

Out[13]: <AxesSubplot:xlabel='Gender', ylabel='count'>



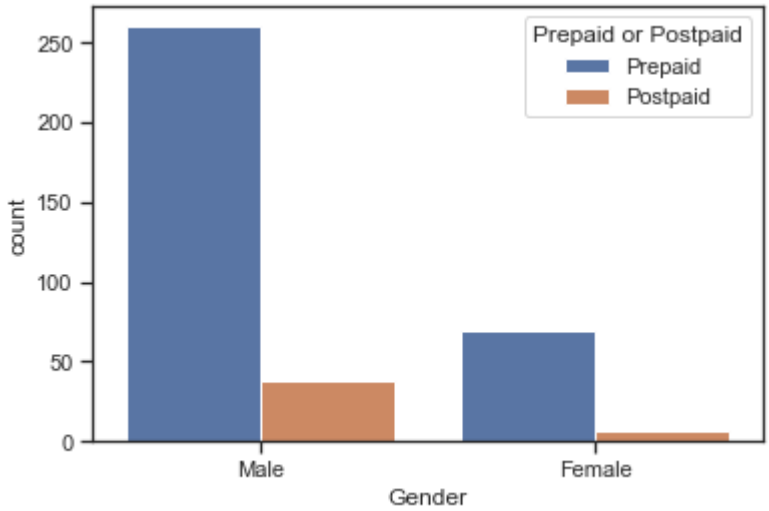
In [14]:
```
sns.countplot(x="Gender" , hue="Which mobile sim do you use" , data=df)
```

Out[14]: <AxesSubplot:xlabel='Gender', ylabel='count'>



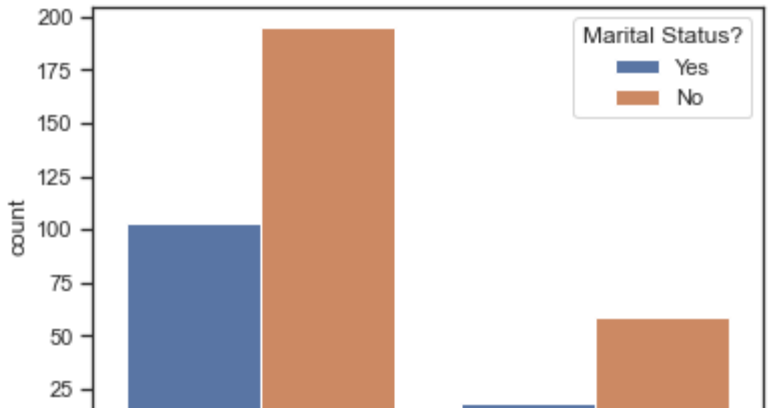
In [15]:
```
sns.countplot(x="Gender" , hue="Prepaid or Postpaid"  , data=df)
```

Out[15]: <AxesSubplot:xlabel='Gender', ylabel='count'>



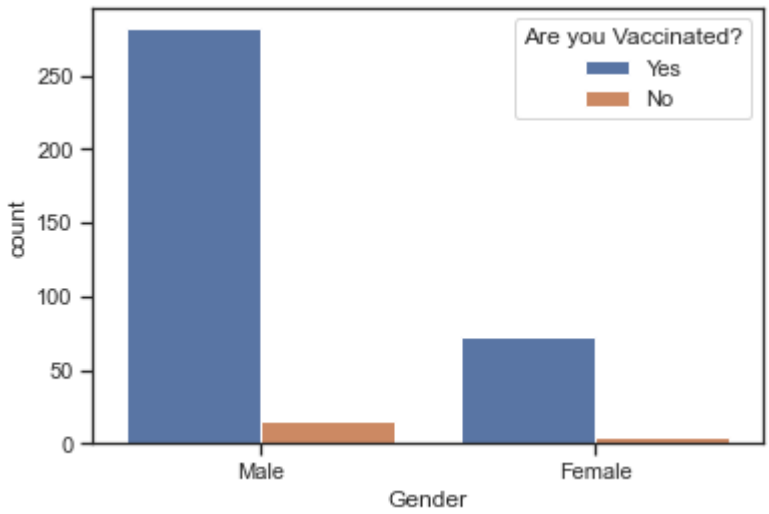
In [16]:
```
sns.countplot(x="Gender" , hue="Marital Status?", data=df)
```

Out[16]: <AxesSubplot:xlabel='Gender', ylabel='count'>



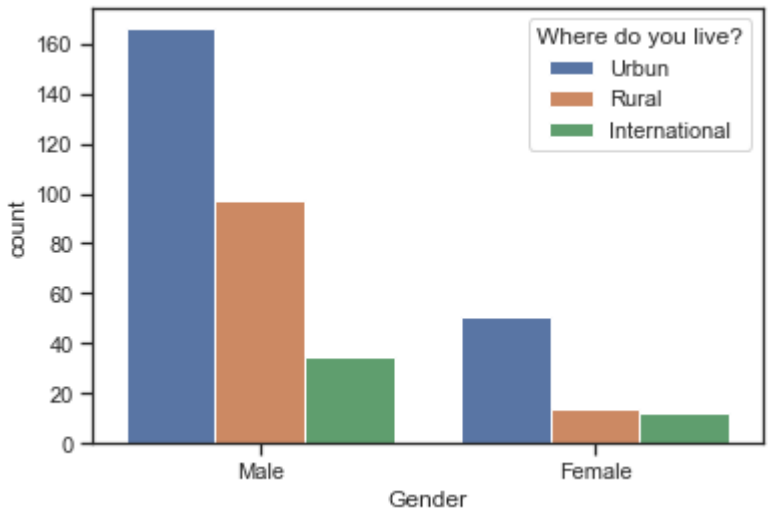
In [17]:
```
sns.countplot(x="Gender" , hue="Are you Vaccinated?", data=df)
```

Out[17]: <AxesSubplot:xlabel='Gender', ylabel='count'>

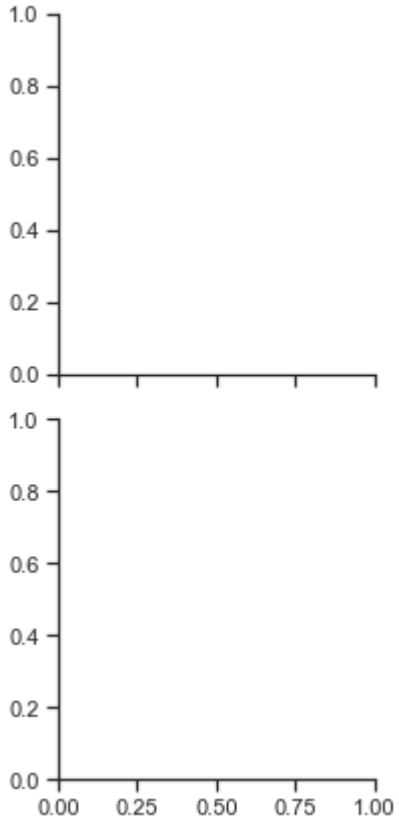


In [18]:
```
sns.countplot(x="Gender" , hue="Where do you live?", data=df)
```

Out[18]: <AxesSubplot:xlabel='Gender', ylabel='count'>

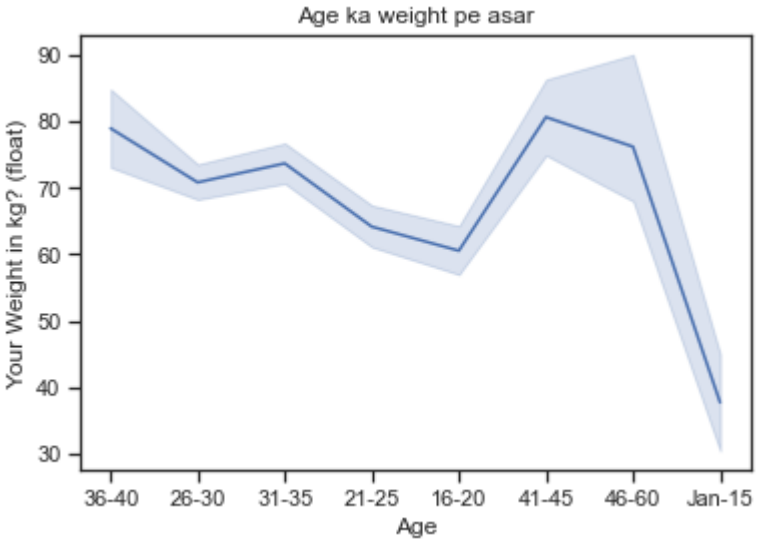


In [19]:
```
g=sns.FacetGrid(row="Gender" , hue="Your Weight in kg? (float)", data=df)
```



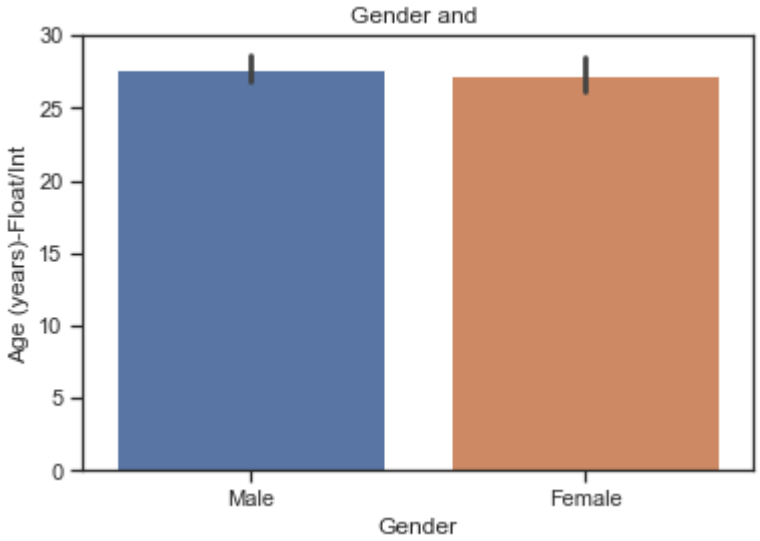
In [20]:
```
df
#Draw a line plot
plt.title("Age ka weight pe asar")
sns.lineplot(x="Age" , y="Your Weight in kg? (float)" , data=df)
```

Out[20]:
```
<AxesSubplot:title={'center':'Age ka weight pe asar'}, xlabel='Age', ylabel='Y
our Weight in kg? (float)'>
```

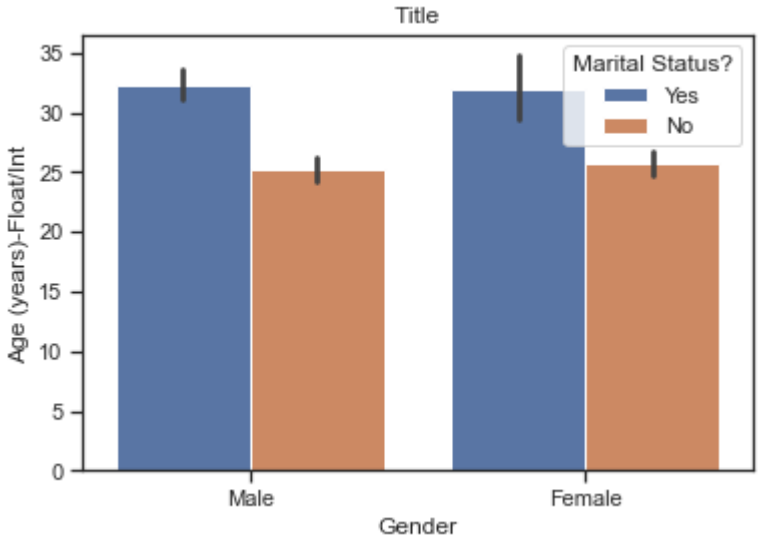


In [21]:
```
#BAR PLOT
sns.barplot(x="Gender" , y="Age (years)-Float/Int" , data=df)
plt.title("Gender and ")
```

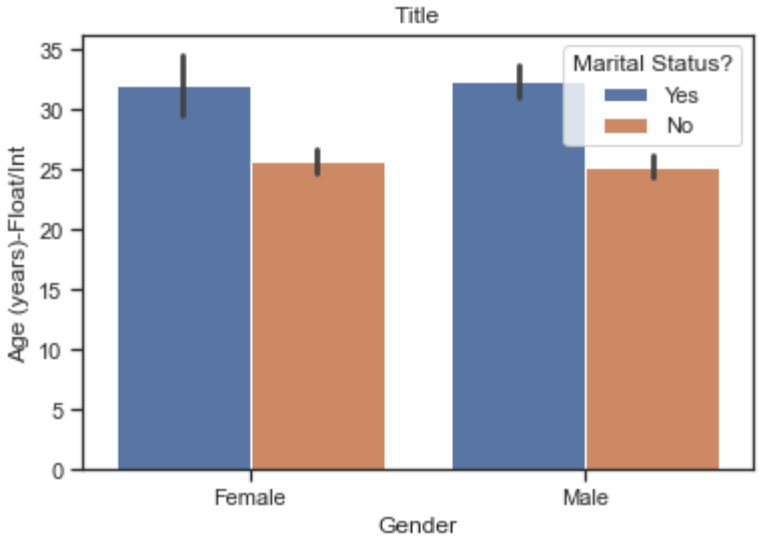
Out[21]: Text(0.5, 1.0, 'Gender and ')



In [22]:
```
#BAR PLOT
sns.barplot(x="Gender" , y="Age (years)-Float/Int" , hue="Marital Status?" , c
plt.title(" Title ")
plt.show()
```

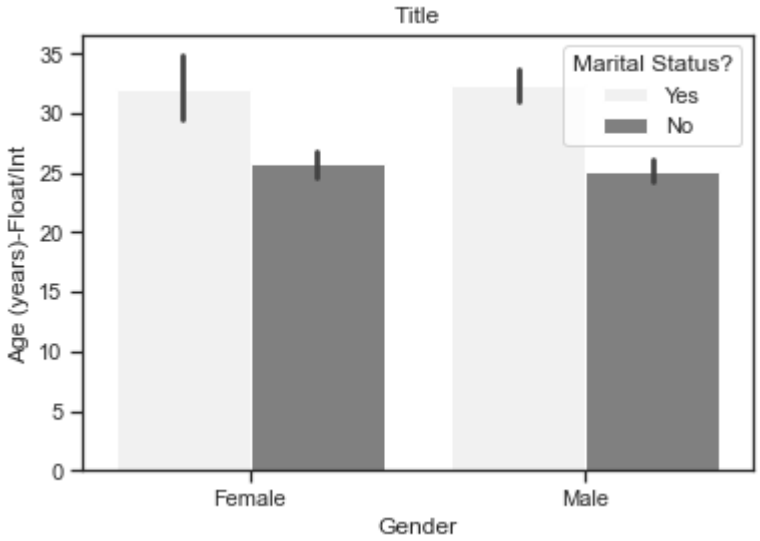


In [23]:
```
#BAR PLOT
# switch female to left and male to right
sns.barplot(x="Gender" , y="Age (years)-Float/Int" , hue="Marital Status?" , c
plt.title(" Title ")
plt.show()
```



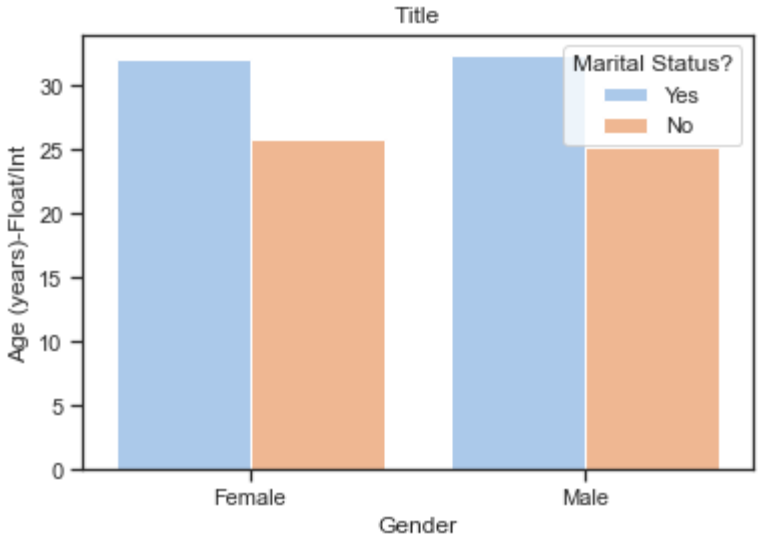
In [24]:

```
#BAR PLOT
# Color Pelets
sns.barplot(x="Gender" , y="Age (years)-Float/Int" , hue="Marital Status?" , c
plt.title(" Title ")
plt.show()
```



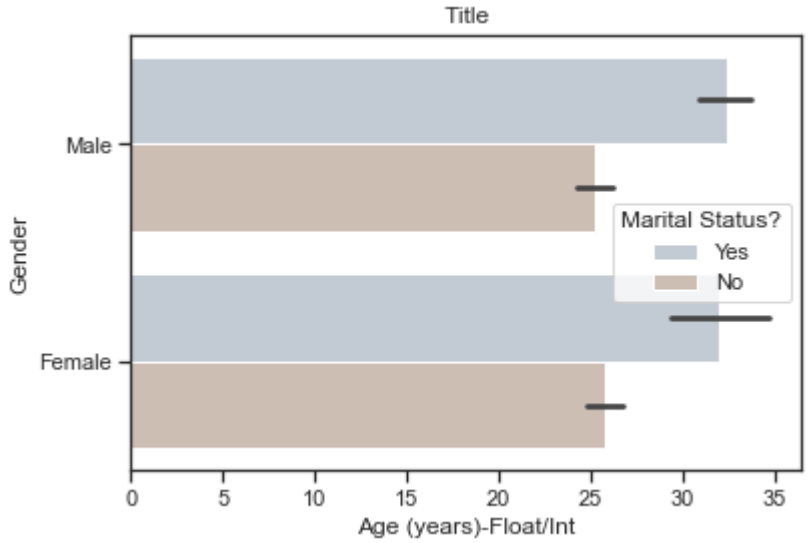
In [25]:

```
#BAR PLOT
# Color Pelets
# ci = none removes or stick the arrow on top of bar
# and pallete gives the sea born pallet some color we can find different palle
sns.barplot(x="Gender" , y="Age (years)-Float/Int" , hue="Marital Status?" , c
            palette = 'pastel' ,  ci=None)
plt.title(" Title ")
plt.show()
```



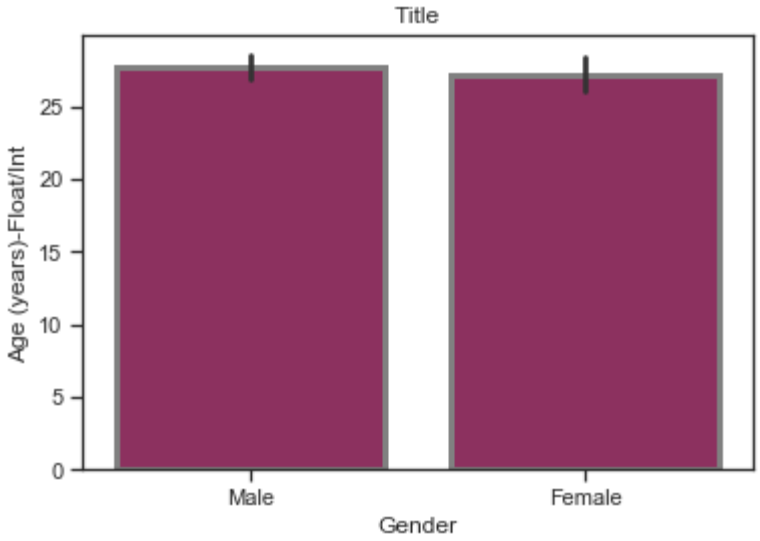
In [26]:

```
#Horizontal Plot by taking string variable on y axis and numeric variable on
#BAR PLOT

from numpy import mean
sns.barplot(x="Age (years)-Float/Int" , y="Gender" , hue="Marital Status?" ,
            palette = 'pastel',estimator=mean , saturation=0.2)
plt.title(" Title ")
plt.show()
```



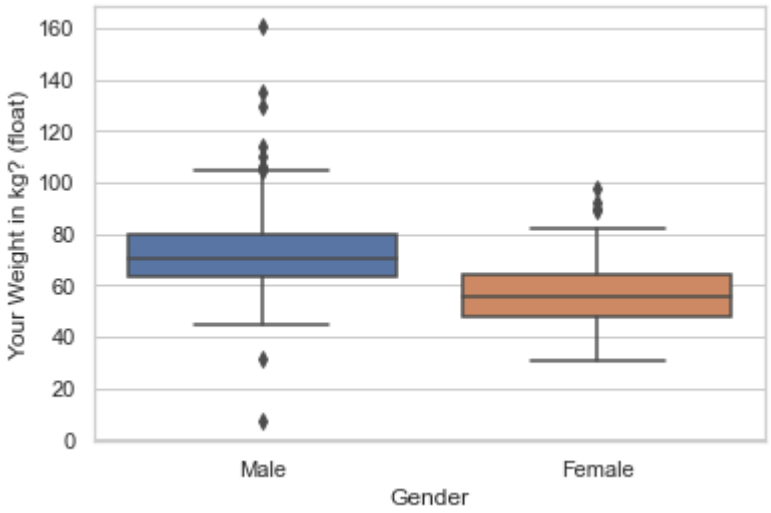
In [27]:

```
# now instead of palette entering our own values
# line width changes the motai of line
# errcolor err is actually this arrow in graph in centre of bar %
# err & edge color value ranges from 0.0 to 0.9
# here in facecolor 4 values are RGB is red green blue values in an image and
# first RGB 3 values combine to give a color and then in last Alpha give valu
sns.barplot(x="Gender" , y="Age (years)-Float/Int", data=df ,
            linewidth=3, facecolor=(0.5,0.1,0.3,0.9),
             errcolor="0.2", edgecolor="0.5")
plt.title(" Title ")
plt.show()
```



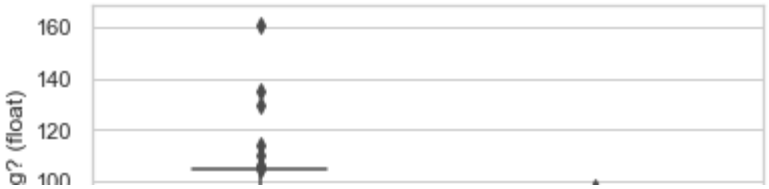
In [34]:
```
sns.set(style='whitegrid')

sns.boxplot(x="Gender",y="Your Weight in kg? (float)" , data=df)
```

Out[34]: <AxesSubplot:xlabel='Gender', ylabel='Your Weight in kg? (float)'>



In [33]:
```
sns.set(style='whitegrid')

sns.boxplot(x="Gender",y="Your Weight in kg? (float)" ,saturation=0.9, data=d
```

Out[33]: <AxesSubplot:xlabel='Gender', ylabel='Your Weight in kg? (float)'>



In [32]:
```
#this will describe the data for understanding
df.describe()
```

Out[32]:

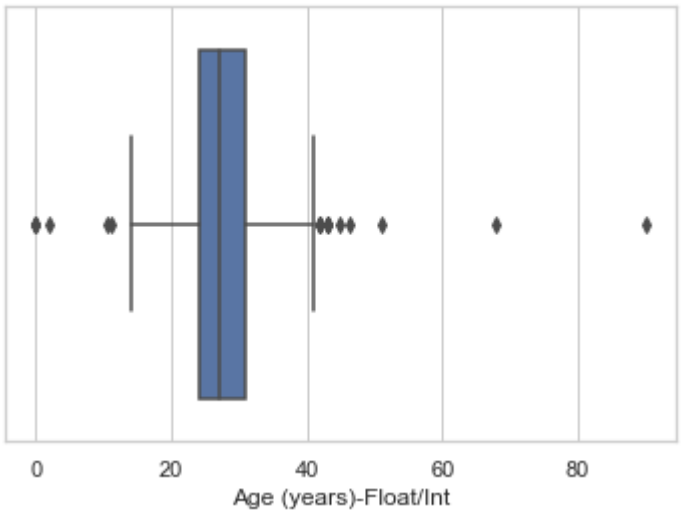
	Age (years)-Float/Int	Your Weight in kg? (float)	Height in cm? Freelancer- (Float)	How many hours you code a day? (int) e.g: 5,4,3	Light kitni der band hti hy? int
count	375.000000	375.000000	375.000000	375.000000	375.000000
mean	27.576933	69.321147	162.679282	2.976027	3.618667
std	7.224460	16.264434	172.246844	2.088115	7.407986
min	0.000000	7.000000	0.000000	0.000000	0.000000
25%	24.000000	58.050000	158.000000	2.000000	0.000000
50%	27.000000	68.300000	169.000000	3.000000	2.000000
75%	31.000000	78.500000	175.225000	4.000000	4.000000
max	90.000000	161.000000	1661.160000	18.000000	72.000000

Catagorical variable mostly on x axis or on hue

numeric variable mostly on y axis , we dont take this on hue

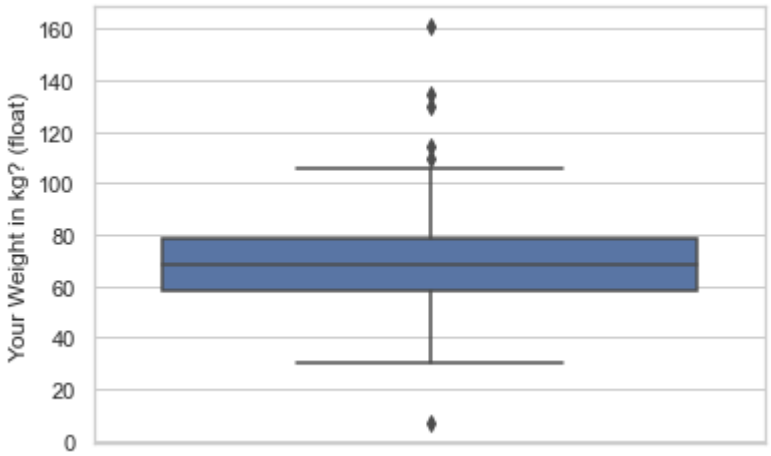
In [31]:
```
#we can draw box plot of one thing as well, we will take on of value shown in
sns.set(style='whitegrid')

sns.boxplot(x=df['Age (years)-Float/Int'])
```

Out[31]: <AxesSubplot:xlabel='Age (years)-Float/Int'>



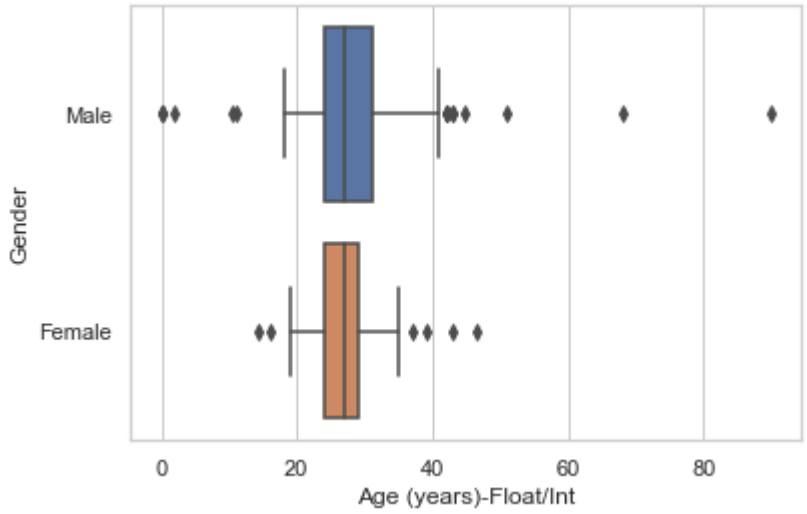
In [35]:
```
# or hama yaha par y axis par be bana skty hain , specially ek ek chez ka ban
sns.set(style='whitegrid')

sns.boxplot(y=df['Your Weight in kg? (float)'])
```

Out[35]: <AxesSubplot:ylabel='Your Weight in kg? (float)'>



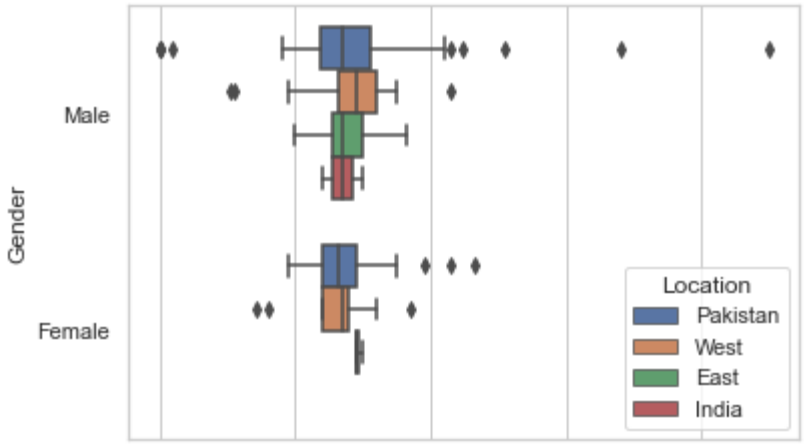
In [44]:
```
#hama yaha par ek string variable dal dain gy gender
sns.set(style='whitegrid')

sns.boxplot(x="Age (years)-Float/Int" ,
            y="Gender" , data=df)
```

Out[44]: <AxesSubplot:xlabel='Age (years)-Float/Int', ylabel='Gender'>



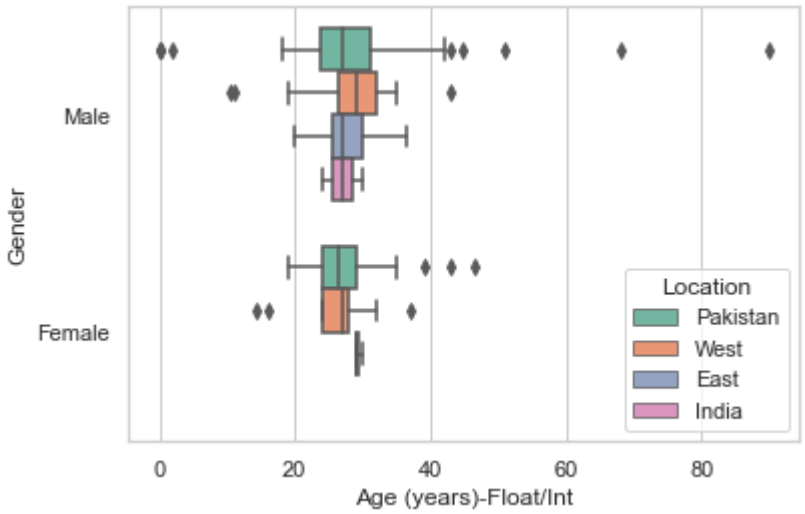
In [45]:
```
# aba yaha par hue introduce kardety hain
sns.set(style='whitegrid')

sns.boxplot(x="Age (years)-Float/Int" , hue="Location" ,y="Gender" , data=df)
```

Out[45]: <AxesSubplot:xlabel='Age (years)-Float/Int', ylabel='Gender'>



In [49]:
```
# Palette and dodge
sns.set(style='whitegrid')

sns.boxplot(x="Age (years)-Float/Int" , hue="Location" ,y="Gender" , data=df
       , palette="Set2" , dodge=True)
```

Out[49]: <AxesSubplot:xlabel='Age (years)-Float/Int', ylabel='Gender'>

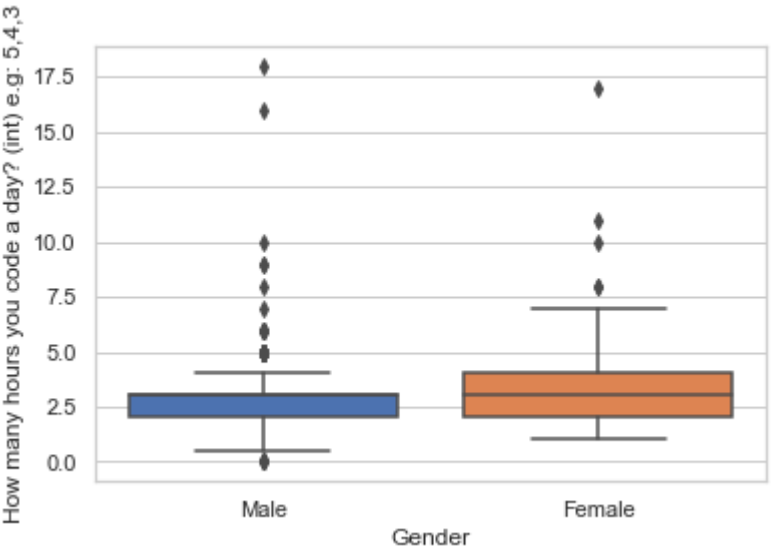


In [53]:
```
sns.set(style='whitegrid')

sns.boxplot(x="Gender" ,y="Age (years)-Float/Int" , data=df
       , saturation=1)
```

Out[53]: <AxesSubplot:xlabel='Gender', ylabel='Age (years)-Float/Int'>

In [59]:
```
# ORIENT
# here argument orient='v' is working but orient ='h' is not
sns.set(style='whitegrid')

sns.boxplot(x="Gender" ,y="How many hours you code a day? (int) e.g: 5,4,3" ,
       , saturation=1 , orient='v')
#
```

Out[59]:
```
<AxesSubplot:xlabel='Gender', ylabel='How many hours you code a day? (int) e.
g: 5,4,3'>
```

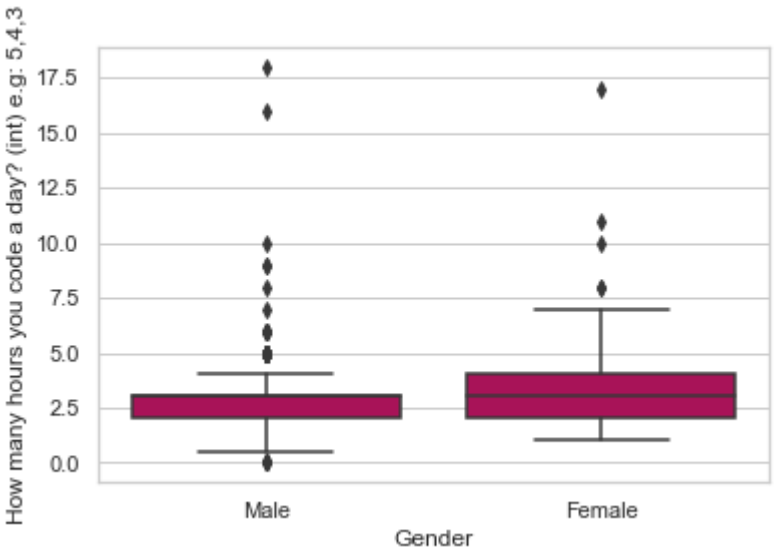


In [61]:
```
# COLOR
sns.set(style='whitegrid')

sns.boxplot(x="Gender" ,y="How many hours you code a day? (int) e.g: 5,4,3" ,
       , saturation=1 , color="red")
#
```

Out[61]:
```
<AxesSubplot:xlabel='Gender', ylabel='How many hours you code a day? (int) e.
g: 5,4,3'>
```

In [63]:
```
# google hex color picker
# chose any color from google hax code and write in color argument
# COLOR
#a81358
sns.set(style='whitegrid')

sns.boxplot(x="Gender" ,y="How many hours you code a day? (int) e.g: 5,4,3" ,
       , saturation=1 , color="#a81358")
# so we can take any color by googling hax color picker
```

Out[63]: <AxesSubplot:xlabel='Gender', ylabel='How many hours you code a day? (int) e.g: 5,4,3'>

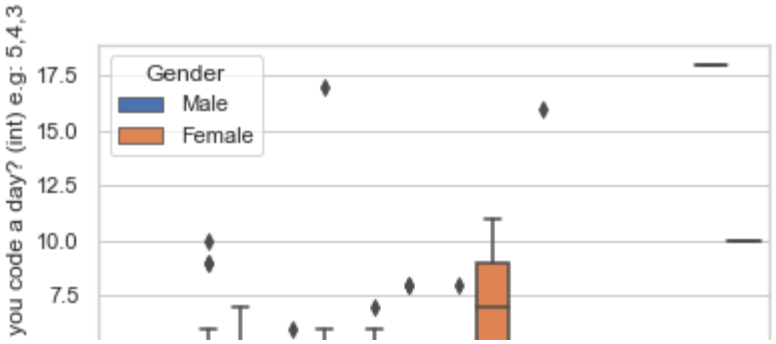


how to manage individual color for each hue color?

use this sns.boxplot(x=" ", y=" ", saturation= 1, data= df , orient=" ", hue=" " , palette ={"Yes": "0.4", "No":"0.8"})

In [80]:
```
# here palette ={"Yes": "0.4", "No":"0.8"} , its not working for me for my pl

sns.set(style='whitegrid')


sns.boxplot(x="Age", y="How many hours you code a day? (int) e.g: 5,4,3", satu
            hue="Gender" ,)
```

Out[80]: <AxesSubplot:xlabel='Age', ylabel='How many hours you code a day? (int) e.g: 5,4,3'>



In [78]:
```
df
```

Out[78]:

	Gender	Location	Age	Qualification_completed	field_of_study	Purpose_for_chilla	What are you?
0	Male	Pakistan	36-40	Masters	Natural Sciences	to boost my skill set	Unemplyed
1	Male	Pakistan	26-30	Bachelors	CS/IT	to boost my skill set	Student
2	Male	Pakistan	31-35	Masters	Enginnering	Switch my field of study	Employed
3	Female	Pakistan	31-35	Masters	CS/IT	to boost my skill set	Employed
4	Female	Pakistan	26-30	Masters	Enginnering	to boost my skill set	Student
...	...	...	...	...	...	...	...
370	Male	Pakistan	26-30	Masters	Enginnering	to boost my skill set	Employed
371	Male	Pakistan	31-35	Bachelors	Enginnering	to boost my skill set	Employed
372	Male	Pakistan	21-25	Bachelors	CS/IT	to boost my skill set	Employed
373	Male	Pakistan	26-30	Masters	Enginnering	to boost my skill set	Employed
374	Female	Pakistan	31-35	Masters	Mathematics	Switch my field of study	Unemplyed

375 rows × 23 columns

In [ ]: