In [7]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
math=pd.read_csv('encoded_math_data.csv',sep=',')
por=pd.read_csv('encoded_por_data.csv',sep=',')
```

In [8]:
```
math
```

Out[8]:

	Unnamed: 0	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	...	famrel
0	0	GP	0	18	1	GT3	0	4	4	at_home	...	4
1	1	GP	0	17	1	GT3	1	1	1	at_home	...	5
2	2	GP	0	15	1	LE3	1	1	1	at_home	...	4
3	3	GP	0	15	1	GT3	1	4	2	health	...	3
4	4	GP	0	16	1	GT3	1	3	3	other	...	4
5	5	GP	1	16	1	LE3	1	4	3	services	...	5
6	6	GP	1	16	1	LE3	1	2	2	other	...	4
7	7	GP	0	17	1	GT3	0	4	4	other	...	4
8	8	GP	1	15	1	LE3	0	3	2	services	...	4
9	9	GP	1	15	1	GT3	1	3	4	other	...	5
10	10	GP	0	15	1	GT3	1	4	4	teacher	...	3
11	11	GP	0	15	1	GT3	1	2	1	services	...	5
12	12	GP	1	15	1	LE3	1	4	4	health	...	4
13	13	GP	1	15	1	GT3	1	4	3	teacher	...	5
14	14	GP	1	15	1	GT3	0	2	2	other	...	4
15	15	GP	0	16	1	GT3	1	4	4	health	...	4
16	16	GP	0	16	1	GT3	1	4	4	services	...	3
17	17	GP	0	16	1	GT3	1	3	3	other	...	5
18	18	GP	1	17	1	GT3	1	3	2	services	...	5
19	19	GP	1	16	1	LE3	1	4	3	health	...	3
20	20	GP	1	15	1	GT3	1	4	3	teacher	...	4
21	21	GP	1	15	1	GT3	1	4	4	health	...	5
22	22	GP	1	16	1	LE3	1	4	2	teacher	...	4
23	23	GP	1	16	1	LE3	1	2	2	other	...	5
24	24	GP	0	15	0	GT3	1	2	4	services	...	4
25	25	GP	0	16	1	GT3	1	2	2	services	...	1
26	26	GP	1	15	1	GT3	1	2	2	other	...	4
27	27	GP	1	15	1	GT3	1	4	2	health	...	2
28	28	GP	1	16	1	LE3	0	3	4	services	...	5
29	29	GP	1	16	1	GT3	1	4	4	teacher	...	4
...	...	...	...	...	...	...	...	...	...	...	...	...
365	365	MS	1	18	0	GT3	1	1	3	at_home	...	3
366	366	MS	1	18	1	LE3	1	4	4	teacher	...	4
367	367	MS	0	17	0	GT3	1	1	1	other	...	5
368	368	MS	0	18	1	GT3	1	2	3	at_home	...	5

	Unnamed: 0	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	...	famrel
369	369	MS	0	18	0	GT3	1	4	4	other	...	3
370	370	MS	0	19	1	LE3	1	3	2	services	...	3
371	371	MS	1	18	0	LE3	1	1	2	at_home	...	4
372	372	MS	0	17	1	GT3	1	2	2	other	...	3
373	373	MS	0	17	0	GT3	1	1	2	other	...	3
374	374	MS	0	18	0	LE3	1	4	4	other	...	5
375	375	MS	0	18	0	GT3	1	1	1	other	...	4
376	376	MS	0	20	1	GT3	1	4	2	health	...	5
377	377	MS	0	18	0	LE3	1	4	4	teacher	...	5
378	378	MS	0	18	1	GT3	1	3	3	other	...	4
379	379	MS	0	17	0	GT3	1	3	1	at_home	...	4
380	380	MS	1	18	1	GT3	1	4	4	teacher	...	3
381	381	MS	1	18	0	GT3	1	2	1	other	...	4
382	382	MS	1	17	1	GT3	1	2	3	other	...	4
383	383	MS	1	19	0	GT3	1	1	1	other	...	4
384	384	MS	1	18	0	GT3	1	4	2	other	...	5
385	385	MS	0	18	0	GT3	1	2	2	at_home	...	5
386	386	MS	0	18	0	GT3	1	4	4	teacher	...	4
387	387	MS	0	19	0	GT3	1	2	3	services	...	5
388	388	MS	0	18	1	LE3	1	3	1	teacher	...	4
389	389	MS	0	18	1	GT3	1	1	1	other	...	1
390	390	MS	1	20	1	LE3	0	2	2	services	...	5
391	391	MS	1	17	1	LE3	1	3	1	services	...	2
392	392	MS	1	21	0	GT3	1	1	1	other	...	5
393	393	MS	1	18	0	LE3	1	3	2	services	...	4
394	394	MS	1	19	1	LE3	1	1	1	other	...	3

395 rows × 34 columns

In [10]:
```
math['M3'].describe()
```

Out[10]:
```
count    395.000000
mean      10.415190
std        4.581443
min        0.000000
25%        8.000000
50%       11.000000
75%       14.000000
max       20.000000
Name: M3, dtype: float64
```

In [11]:
```
por['P3'].describe()
```

Out[11]:
```
count    649.000000
mean      11.906009
std        3.230656
min        0.000000
25%       10.000000
50%       12.000000
75%       14.000000
max       19.000000
Name: P3, dtype: float64
```

In [12]:
```
math.shape
```

Out[12]:
```
(395, 34)
```

In [13]:
```
por.shape
```

Out[13]:
```
(649, 34)
```

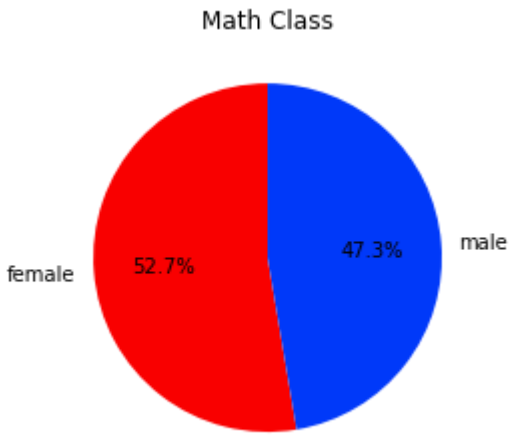
In [14]:
```
math.info()
por.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 395 entries, 0 to 394
Data columns (total 34 columns):
Unnamed: 0    395 non-null int64
school        395 non-null object
sex           395 non-null int64
age           395 non-null int64
address       395 non-null int64
famsize       395 non-null object
Pstatus       395 non-null int64
Medu          395 non-null int64
Fedu          395 non-null int64
Mjob          395 non-null object
Fjob          395 non-null object
reason        395 non-null object
guardian      395 non-null object
traveltime    395 non-null int64
studytime     395 non-null int64
failures      395 non-null int64
schoolsup     395 non-null int64
famsup        395 non-null int64
paid          395 non-null int64
activities    395 non-null int64
nursery       395 non-null int64
higher        395 non-null int64
internet      395 non-null int64
romantic      395 non-null int64
famrel        395 non-null int64
freetime      395 non-null int64
goout         395 non-null int64
Dalc          395 non-null int64
Walc          395 non-null int64
health        395 non-null int64
m_absences    395 non-null int64
M1            395 non-null int64
M2            395 non-null int64
M3            395 non-null int64
dtypes: int64(28), object(6)
memory usage: 105.0+ KB
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 649 entries, 0 to 648
Data columns (total 34 columns):
Unnamed: 0    649 non-null int64
school        649 non-null object
sex           649 non-null int64
age           649 non-null int64
address       649 non-null int64
famsize       649 non-null object
Pstatus       649 non-null int64
Medu          649 non-null int64
Fedu          649 non-null int64
Mjob          649 non-null object
Fjob          649 non-null object
reason        649 non-null object
guardian      649 non-null object
traveltime    649 non-null int64
studytime     649 non-null int64
```

```
failures      649 non-null int64
schoolsup     649 non-null int64
famsup        649 non-null int64
paid          649 non-null int64
activities    649 non-null int64
nursery       649 non-null int64
higher        649 non-null int64
internet      649 non-null int64
romantic      649 non-null int64
famrel        649 non-null int64
freetime      649 non-null int64
goout         649 non-null int64
Dalc          649 non-null int64
Walc          649 non-null int64
health        649 non-null int64
p_absences    649 non-null int64
P1            649 non-null int64
P2            649 non-null int64
P3            649 non-null int64
dtypes: int64(28), object(6)
memory usage: 172.5+ KB
```

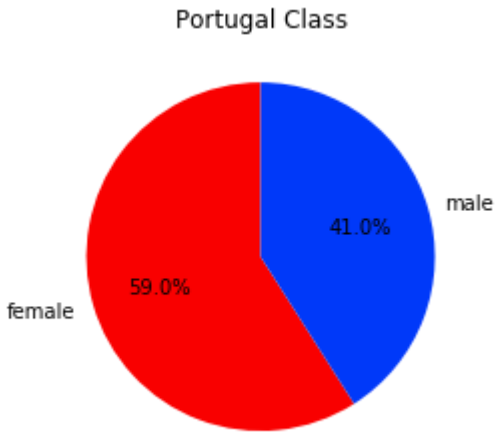
In [17]:
```
colors = ['#F90000','#0039F9']
plt.pie(math['sex'].value_counts(),startangle=90, labels=['female','male'], co
lors = colors,  autopct='%1.1f%%')
plt.title('Math Class')
```

Out[17]:
```
Text(0.5, 1.0, 'Math Class')
```



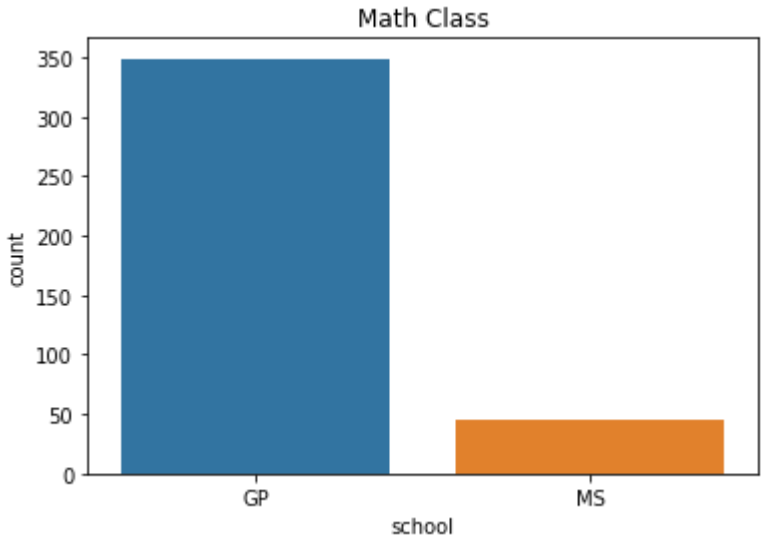
In [18]:
```
colors = ['#F90000', '#0039F9']
plt.pie(por['sex'].value_counts(),startangle=90, labels=['female','male'], col
ors = colors,  autopct='%1.1f%%')
plt.title('Portugal Class')
```

Out[18]: Text(0.5, 1.0, 'Portugal Class')



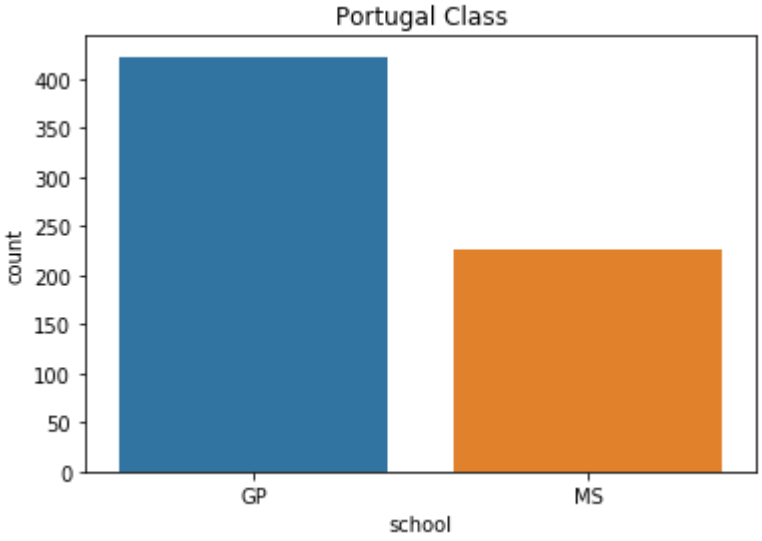
In [19]:
```
sns.countplot(data = math, x = 'school')
plt.title('Math Class')
```

Out[19]: Text(0.5, 1.0, 'Math Class')



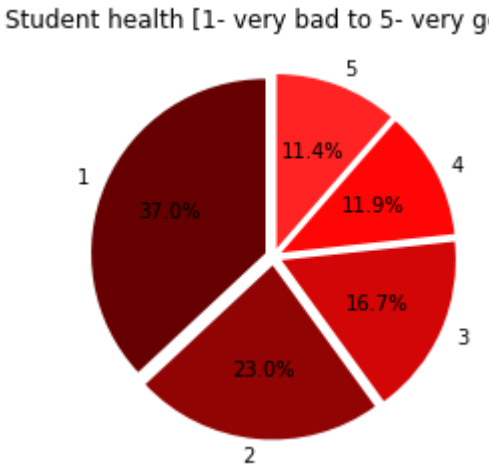
In [20]:
```
sns.countplot(data = por, x = 'school')
plt.title('Portugal Class')
```

Out[20]: Text(0.5, 1.0, 'Portugal Class')

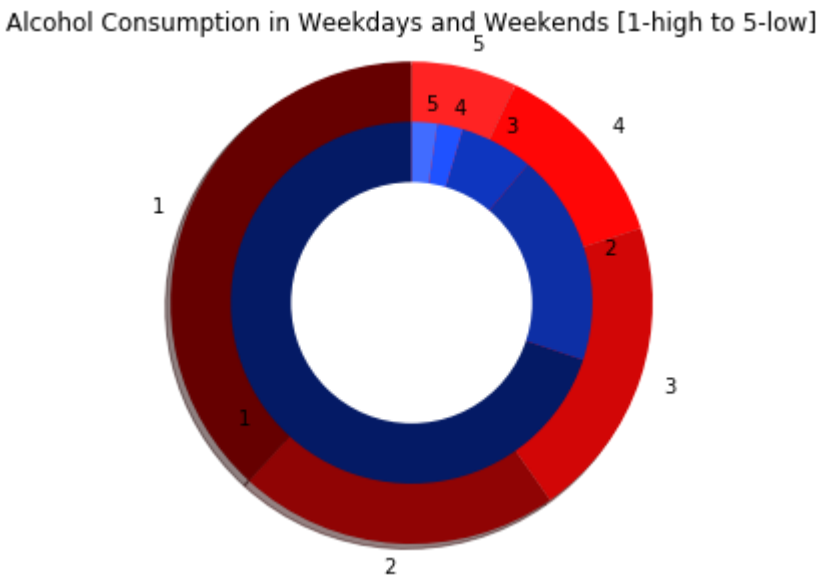


In [21]:
```
colors = ['#660000','#900404','#D20606','#FF0606', '#FF2323']
explode = (0.05,0.05,0.05,0.05,0.05)
plt.pie(math['health'].value_counts(),startangle=90, labels =['1','2','3','4',
'5'], colors=colors, explode = explode, autopct='%1.1f%%')
plt.title('Student health [1- very bad to 5- very good]')
```

Out[21]: Text(0.5, 1.0, 'Student health [1- very bad to 5- very good]')



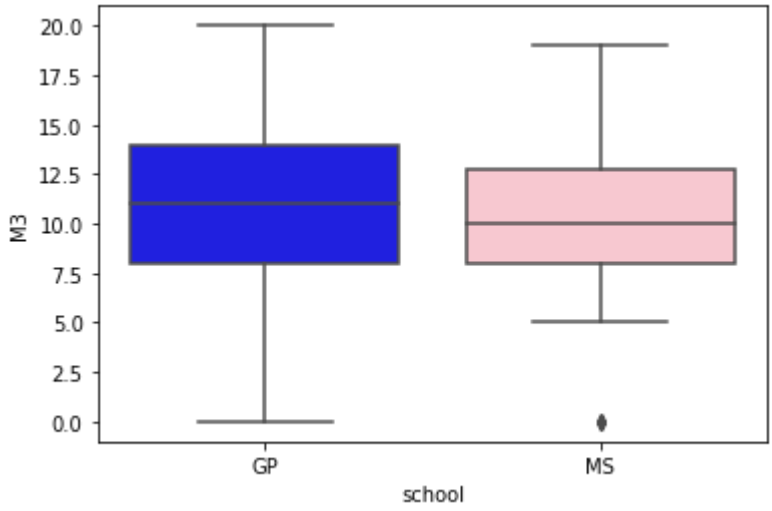
In [22]:
```
# Data to plot
labels_walc = ['1', '2', '3', '4', '5']
sizes_walc = [504, 337, 415, 280]
labels_dalc = ['1','2','3','4','5']
sizes_dalc = [315,189,125,212,270]
colors_walc = ['#660000','#900404','#D20606','#FF0606', '#FF2323']
colors_dalc = ['#041A65', '#0D2FA5', '#0E36BF', '#1F52FF', '#416CFF']

# Plot
plt.pie(math['Walc'].value_counts(), labels=labels_walc, colors=colors_walc, s
tartangle=90,frame=True, shadow=True)
plt.pie(math['Dalc'].value_counts(), labels=labels_dalc, colors=colors_dalc ,r
adius=0.75, startangle=90, shadow=True)
centre_circle = plt.Circle((0,0),0.5,color='black', fc='white',linewidth=0)
fig = plt.gcf()
fig.gca().add_artist(centre_circle)

plt.axis('equal')
plt.tight_layout()
plt.title('Alcohol Consumption in Weekdays and Weekends [1-high to 5-low]')
plt.show()
print('Legend : Red - Weekdays, Blue - Weekends')
```




```
Legend : Red - Weekdays, Blue - Weekends
```

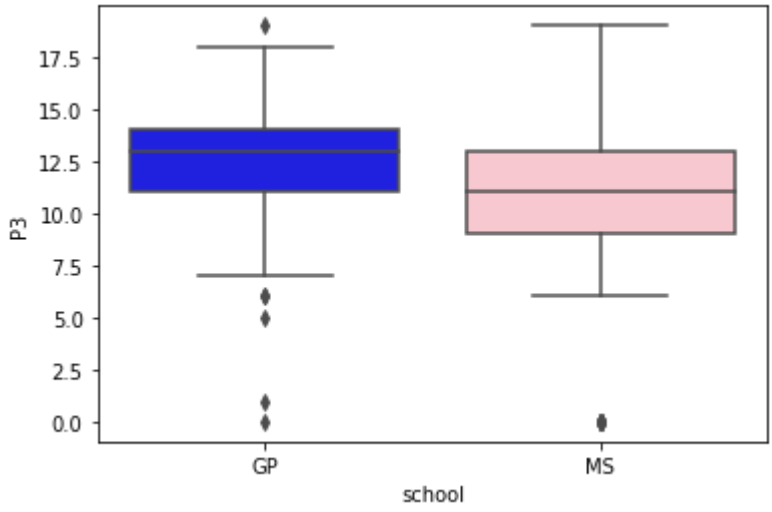
In [23]:
```
sns.boxplot(data = math,palette=["blue", "pink"], x='school', y='M3')
```

Out[23]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x256b130acf8>
```



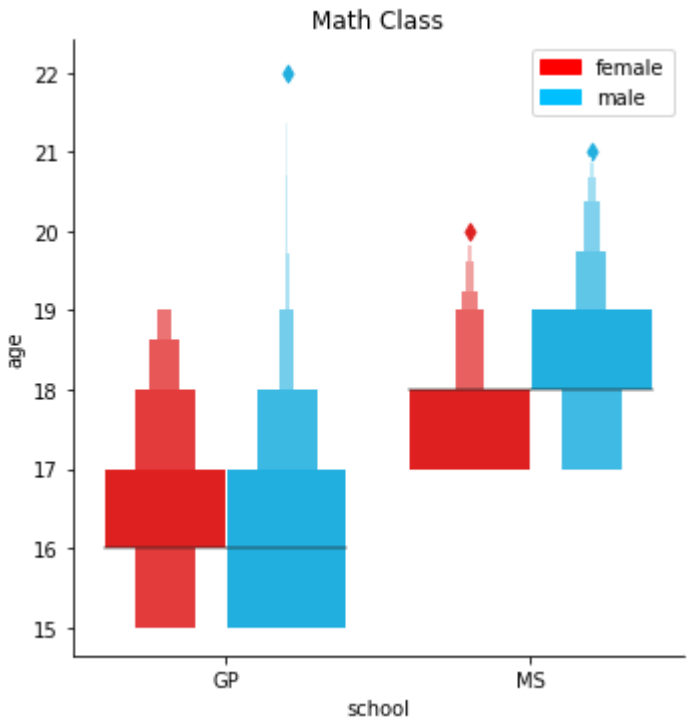
In [24]:
```
sns.boxplot(data = por ,palette=["blue", "pink"], x='school', y='P3')
```

Out[24]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x256b13f9f98>
```



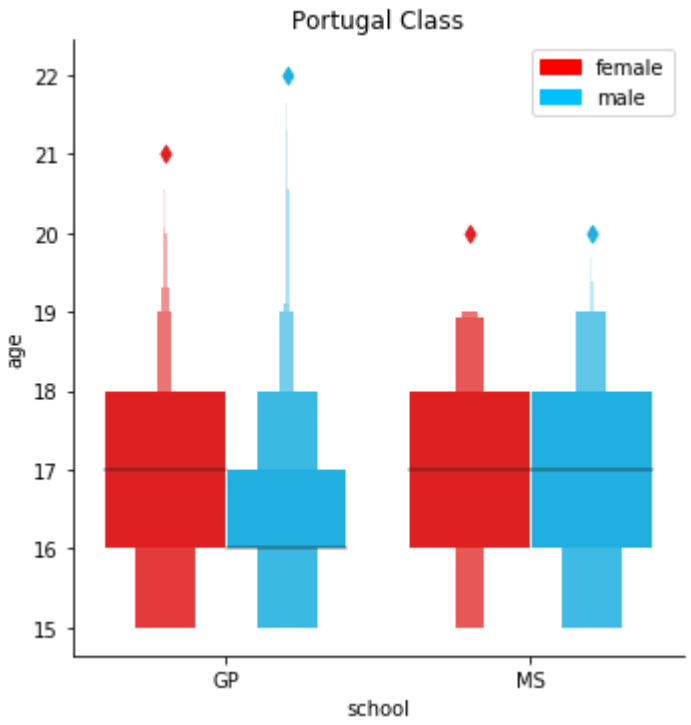
In [25]:
```
import matplotlib.patches as mpatches

sns.catplot(x="school", y="age", hue="sex", kind="boxen", palette=["r", "deepskyblue"], data= math, legend_out = False);

red_patch = mpatches.Patch(color='r', label='female')
cyan_patch = mpatches.Patch(color='deepskyblue', label='male')
plt.legend(handles=[red_patch, cyan_patch])
plt.title('Math Class')
```

Out[25]:
```
Text(0.5, 1, 'Math Class')
```

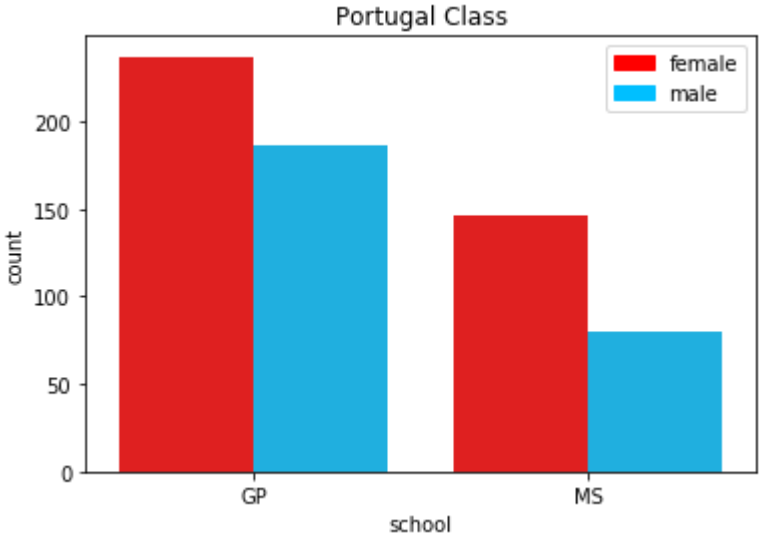


In [26]:
```
import matplotlib.patches as mpatches

sns.catplot(x="school", y="age", hue="sex", kind="boxen", palette=["r", "deepskyblue"], data= por, legend_out = False);

red_patch = mpatches.Patch(color='r', label='female')
cyan_patch = mpatches.Patch(color='deepskyblue', label='male')
plt.legend(handles=[red_patch, cyan_patch])
plt.title('Portugal Class')
```

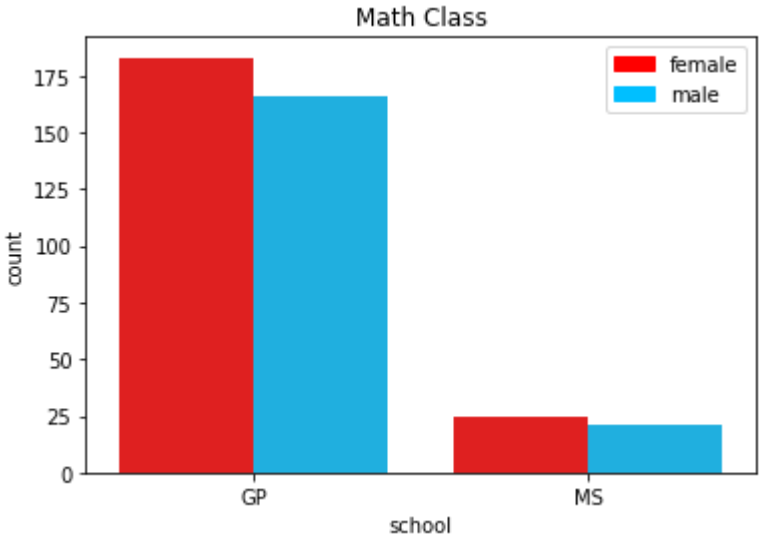
Out[26]: Text(0.5, 1, 'Portugal Class')



In [49]:
```
sns.countplot(x="school", hue="sex", data=por, palette=["r", "deepskyblue"])
red_patch = mpatches.Patch(color='r', label='female')
cyan_patch = mpatches.Patch(color='deepskyblue', label='male')
plt.legend(handles=[red_patch, cyan_patch])
plt.title('Portugal Class')
plt.show()
```

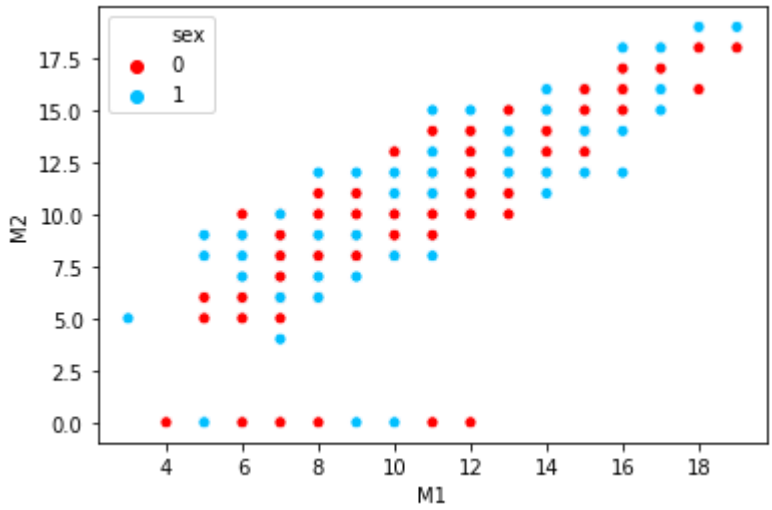


In [50]:
```
sns.countplot(x="school", hue="sex", data=math, palette=["r", "deepskyblue"])

red_patch = mpatches.Patch(color='r', label='female')
cyan_patch = mpatches.Patch(color='deepskyblue', label='male')
plt.legend(handles=[red_patch, cyan_patch])

plt.title('Math Class')
plt.show()
```



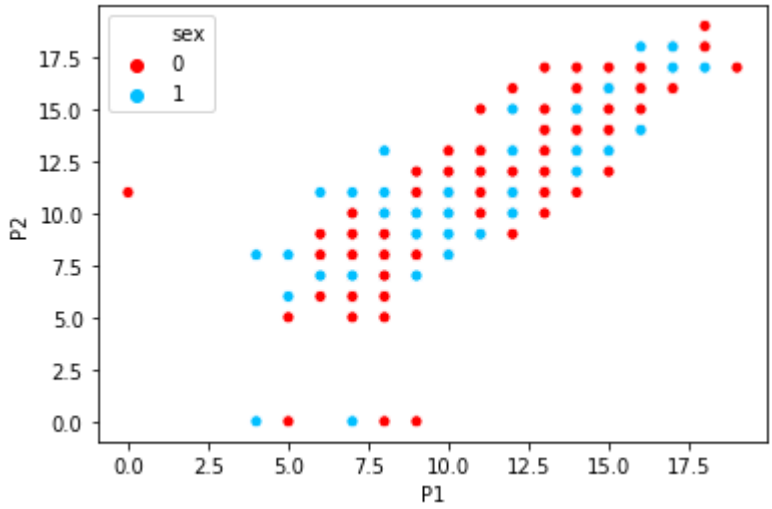
In [51]:
```
sns.scatterplot(x="M1", y="M2", hue="sex", size=None, data=math, legend='brief', palette=["r", "deepskyblue"])
```

Out[51]: <matplotlib.axes._subplots.AxesSubplot at 0x256b18c57f0>



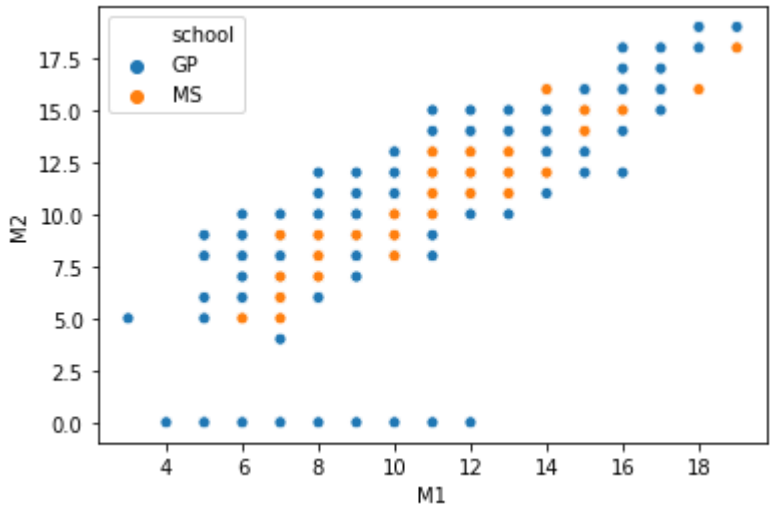
In [52]:
```
sns.scatterplot(x="P1", y="P2", hue="sex", size=None, data=por,  palette=["r", "deepskyblue"])
```

Out[52]: <matplotlib.axes._subplots.AxesSubplot at 0x256b2a5dcc0>



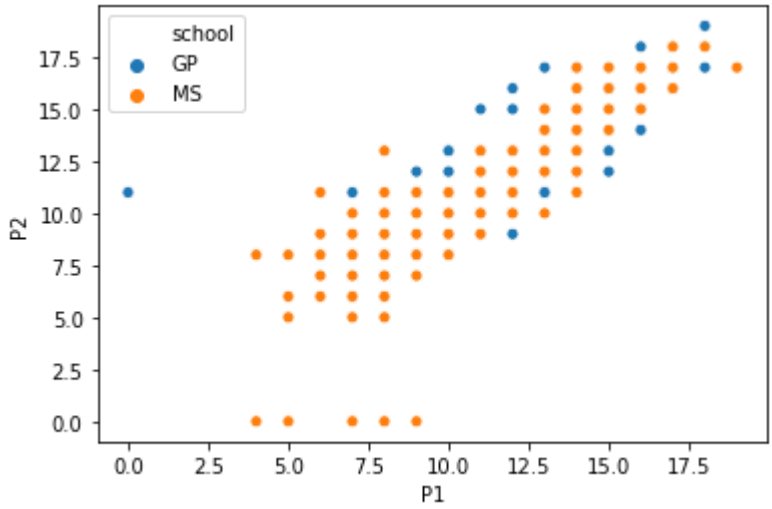
In [53]:
```
sns.scatterplot(x="M1", y="M2", hue="school", size=None, data=math, legend='brief')
```

Out[53]: <matplotlib.axes._subplots.AxesSubplot at 0x256b2e39438>

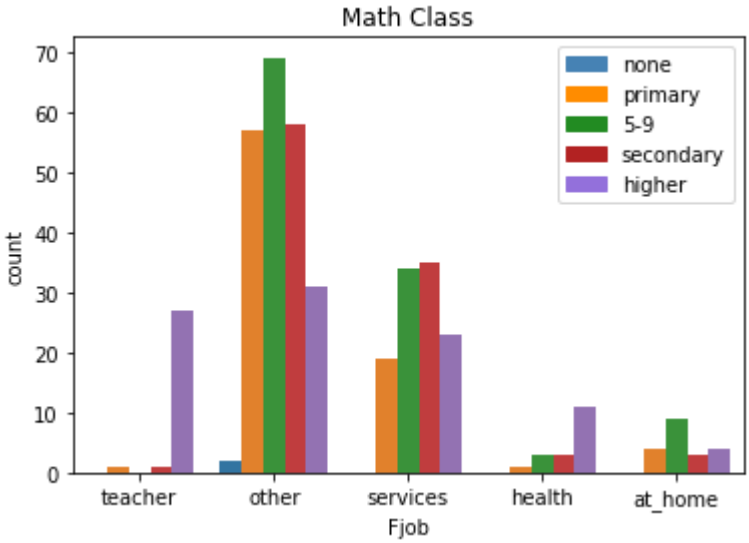


In [54]:
```
sns.scatterplot(x="P1", y="P2", hue="school", size=None, data=por, legend='brief')
```

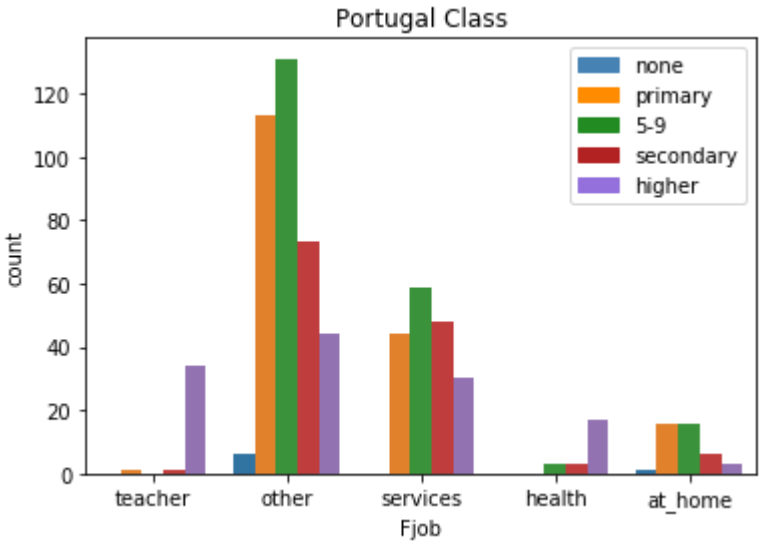
Out[54]: <matplotlib.axes._subplots.AxesSubplot at 0x256b2dcaf28>



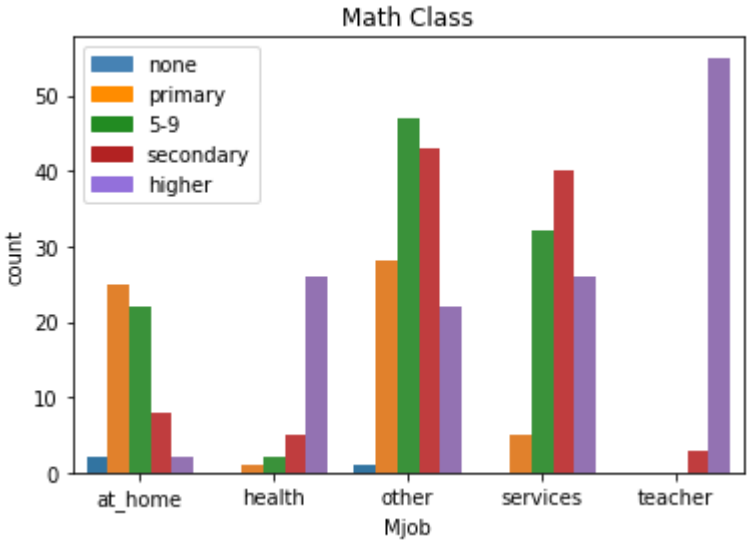
In [55]:
```
sns.countplot(x="Fjob", hue="Fedu", data=math)
plt.title("Father's job compared to their education level")
s_patch = mpatches.Patch(color='steelblue', label='none')
d_patch = mpatches.Patch(color='darkorange', label='primary')
f_patch = mpatches.Patch(color='forestgreen', label='5-9')
fi_patch = mpatches.Patch(color='firebrick', label='secondary')
m_patch = mpatches.Patch(color='mediumpurple', label='higher')
plt.legend(handles=[s_patch, d_patch, f_patch, fi_patch, m_patch])
plt.title('Math Class')
plt.show()
```



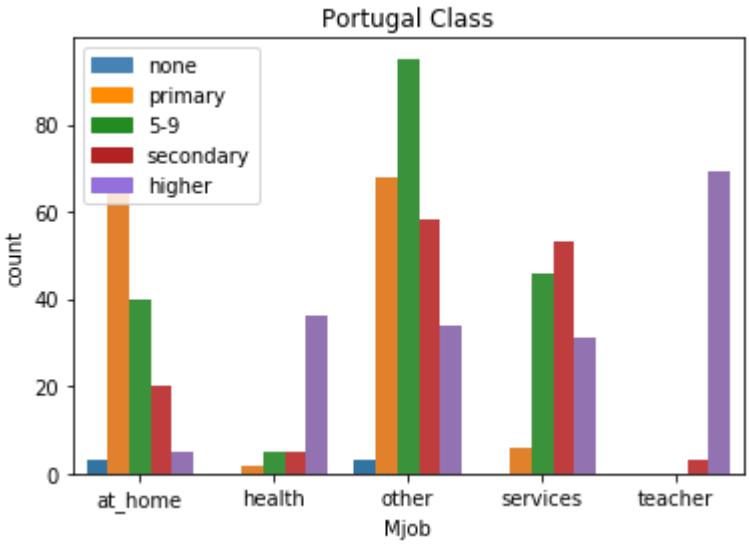
In [56]:
```
s = sns.countplot(x="Fjob", hue="Fedu", data=por)
plt.title("Father's job compared to their education level")
s.legend_.remove()
s_patch = mpatches.Patch(color='steelblue', label='none')
d_patch = mpatches.Patch(color='darkorange', label='primary')
f_patch = mpatches.Patch(color='forestgreen', label='5-9')
fi_patch = mpatches.Patch(color='firebrick', label='secondary')
m_patch = mpatches.Patch(color='mediumpurple', label='higher')
plt.legend(handles=[s_patch, d_patch, f_patch, fi_patch, m_patch])
plt.title('Portugal Class')
plt.show()
```



In [57]:
```
sns.countplot(x="Mjob", hue="Medu", data=math)
plt.title("Mother's job compared to their education level")
s_patch = mpatches.Patch(color='steelblue', label='none')
d_patch = mpatches.Patch(color='darkorange', label='primary')
f_patch = mpatches.Patch(color='forestgreen', label='5-9')
fi_patch = mpatches.Patch(color='firebrick', label='secondary')
m_patch = mpatches.Patch(color='mediumpurple', label='higher')
plt.legend(handles=[s_patch, d_patch, f_patch, fi_patch, m_patch])
plt.title('Math Class')
plt.show()
```

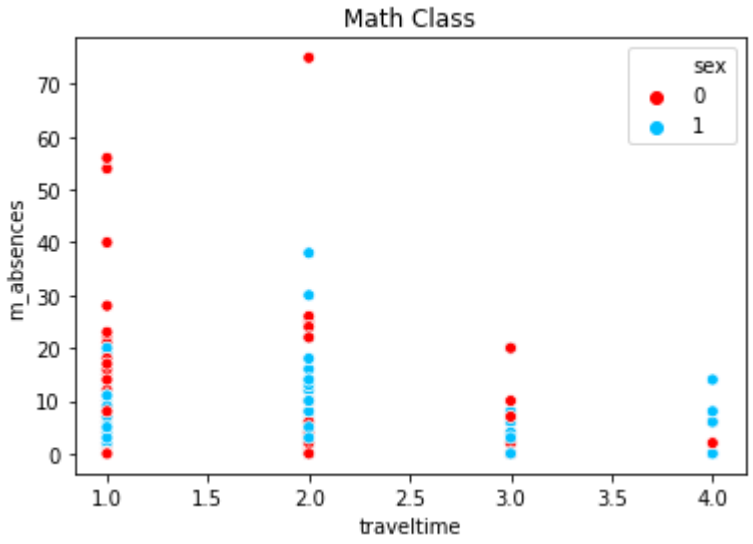


In [58]:
```
r = sns.countplot(x="Mjob", hue="Medu", data=por)
plt.title("Mother's job compared to their education level")
r.legend_.remove()
s_patch = mpatches.Patch(color='steelblue', label='none')
d_patch = mpatches.Patch(color='darkorange', label='primary')
f_patch = mpatches.Patch(color='forestgreen', label='5-9')
fi_patch = mpatches.Patch(color='firebrick', label='secondary')
m_patch = mpatches.Patch(color='mediumpurple', label='higher')
plt.legend(handles=[s_patch, d_patch, f_patch, fi_patch, m_patch])
plt.title('Portugal Class')
plt.show()
```



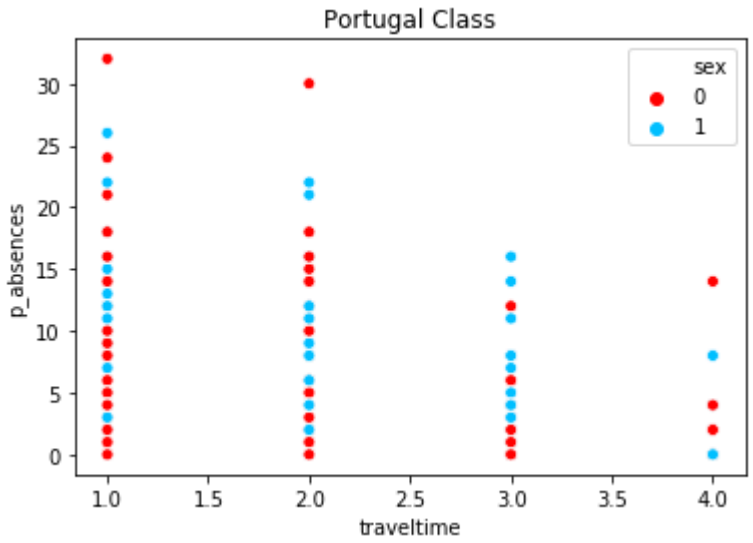
In [59]:
```
sns.scatterplot(x="traveltime", y="m_absences", hue="sex" , size=None, palette
=["r", "deepskyblue"], data=math, legend='brief')
plt.title('Math Class')
```

Out[59]:
```
Text(0.5, 1.0, 'Math Class')
```



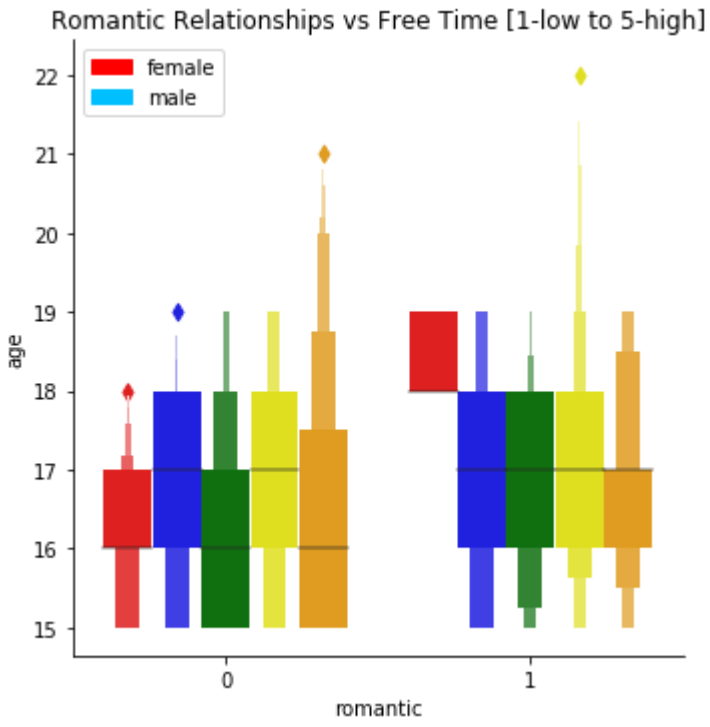
In [60]:
```
sns.scatterplot(x="traveltime", y="p_absences", hue="sex" , size=None, palette
=["r", "deepskyblue"], data=por, legend='brief')
plt.title('Portugal Class')
```

Out[60]: Text(0.5, 1.0, 'Portugal Class')

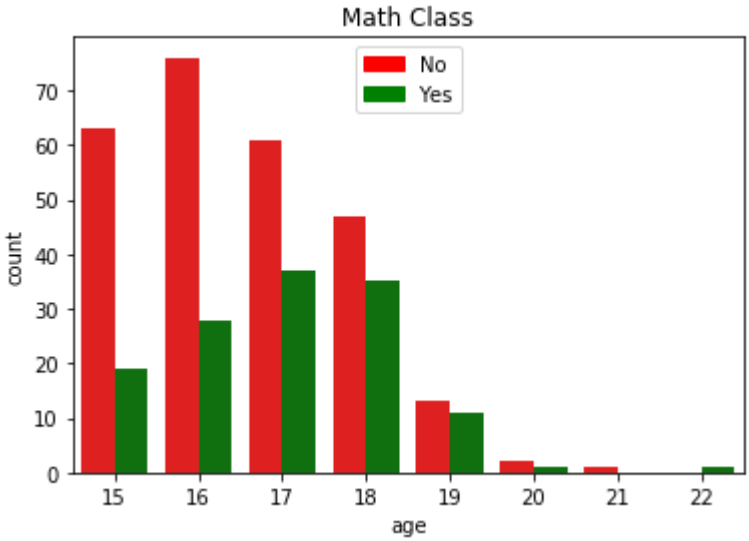


In [32]:
```
sns.catplot(x="romantic", y="age", hue="freetime", kind="boxen", palette=["re
d","blue","green","yellow","orange"], data= math, legend_out = False);

red_patch = mpatches.Patch(color='r', label='female')
blue_patch = mpatches.Patch(color='deepskyblue', label='male')
plt.legend(handles=[red_patch, cyan_patch])

plt.title("Romantic Relationships vs Free Time [1-low to 5-high]")
```

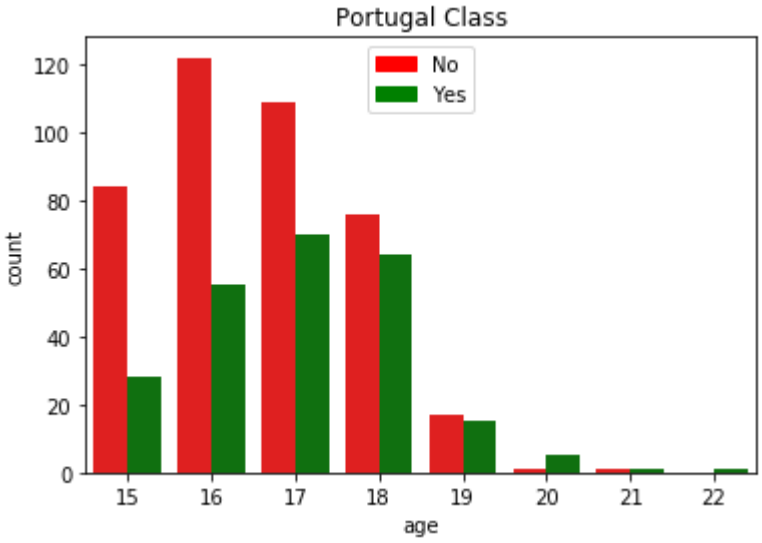
Out[32]: Text(0.5, 1, 'Romantic Relationships vs Free Time [1-low to 5-high]')



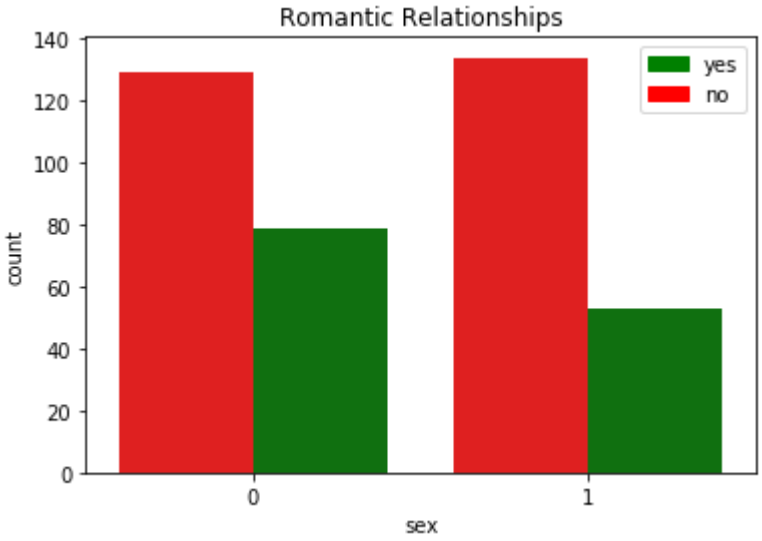
In [39]:
```
sns.countplot(x="age", hue="romantic", data=math, palette=["red","green"])
plt.title("Romantic Relationships")
s_patch = mpatches.Patch(color='red', label='No')
d_patch = mpatches.Patch(color='green', label='Yes')
plt.legend(handles=[s_patch, d_patch])
plt.title('Math Class')

plt.show()
```



In [40]:
```
sns.countplot(x="age", hue="romantic", data=por, palette=["red","green"])
plt.title("Romantic Relationships")
s_patch = mpatches.Patch(color='red', label='No')
d_patch = mpatches.Patch(color='green', label='Yes')

plt.legend(handles=[s_patch, d_patch])
plt.title('Portugal Class')

plt.show()
```

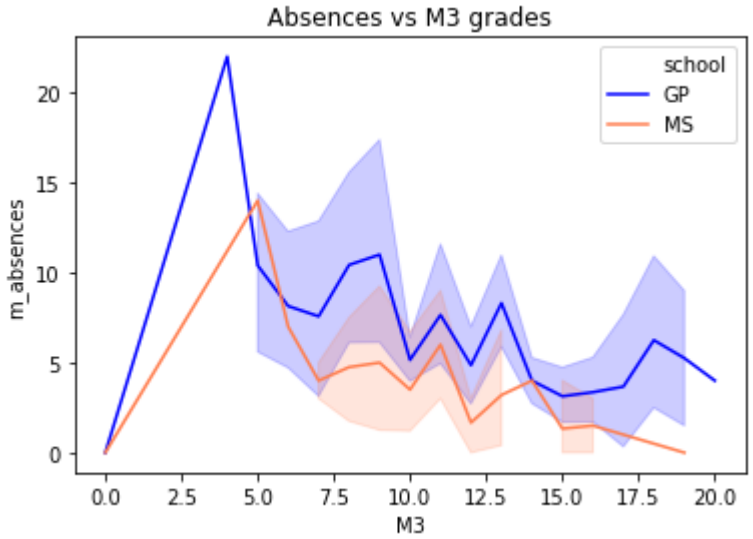


In [41]:
```
sns.countplot(x="sex", hue="romantic", data=math, palette=["red","green"])
plt.title("Romantic Relationships")
s_patch = mpatches.Patch(color='green', label='yes')
d_patch = mpatches.Patch(color='red', label='no')

plt.legend(handles=[s_patch, d_patch])

plt.show()
```



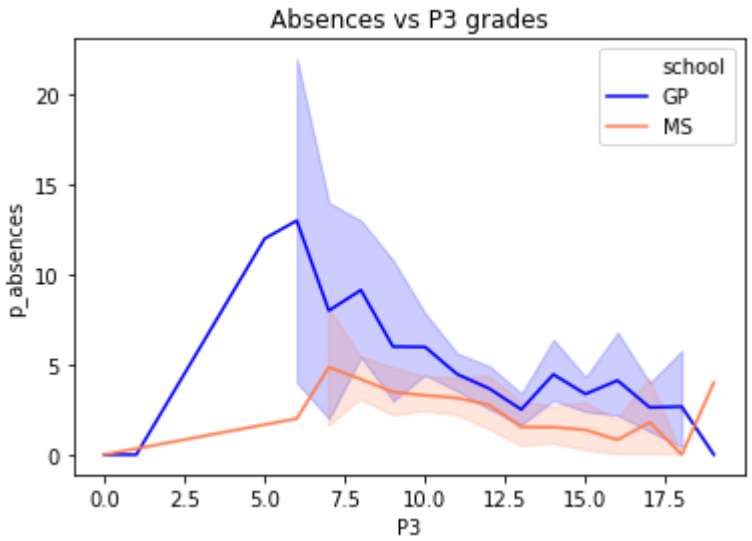
In [42]:
```
sns.lineplot(x="M3", y="m_absences",hue="school",size=None, palette=["blue",
"coral"], data=math, legend= 'brief')
plt.title("Absences vs M3 grades")
```

Out[42]: Text(0.5, 1.0, 'Absences vs M3 grades')



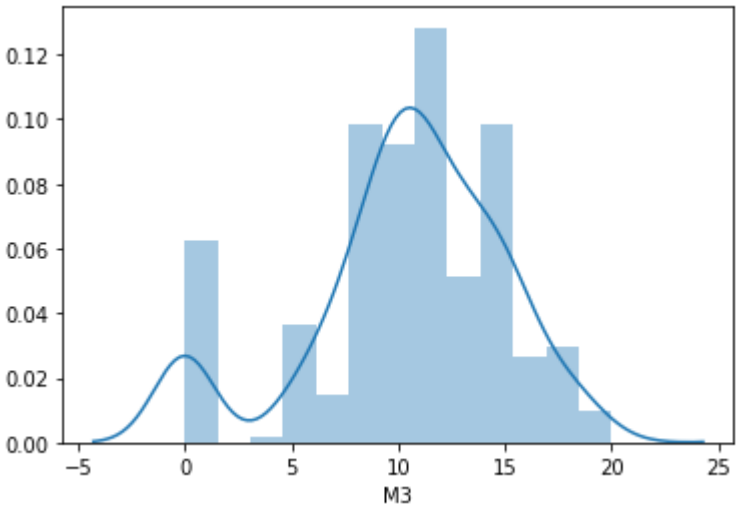
In [43]:
```
sns.lineplot(x="P3", y="p_absences",hue="school",size=None, palette=["blue",
"coral"], data=por, legend= 'brief')
plt.title("Absences vs P3 grades")
```

Out[43]: Text(0.5, 1.0, 'Absences vs P3 grades')



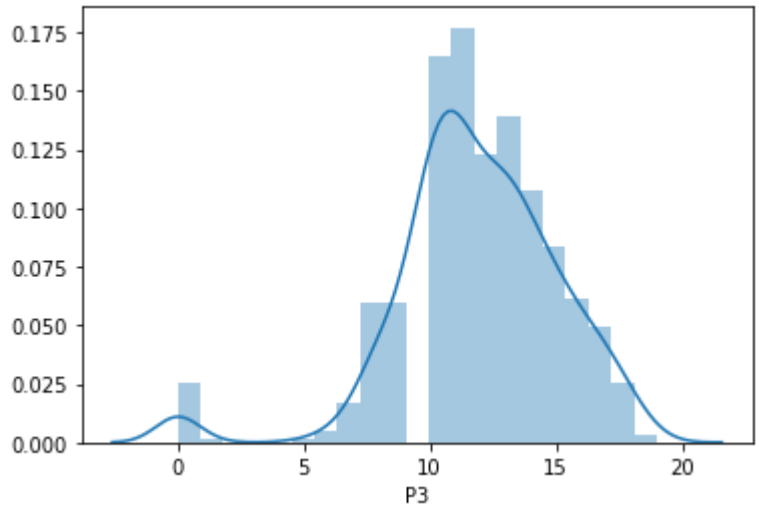
In [44]:
```
M3 = math["M3"]
sns.distplot(M3)
```

Out[44]: <matplotlib.axes._subplots.AxesSubplot at 0x256b2bfb5f8>

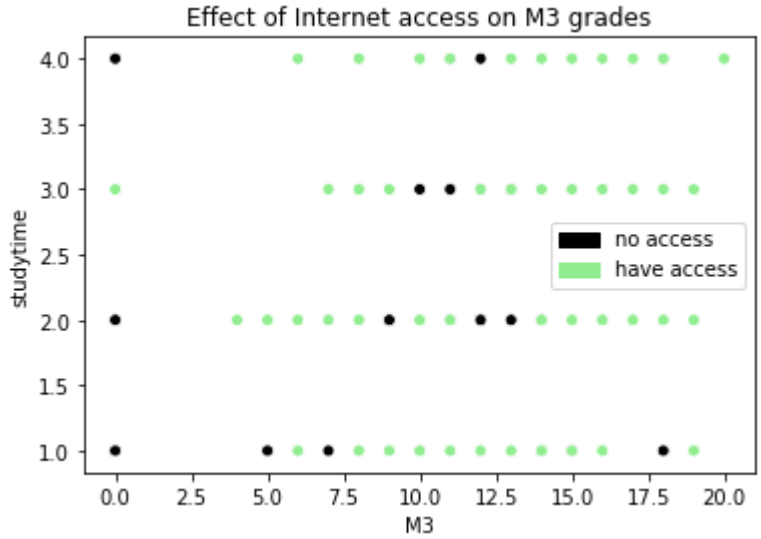


In [45]:
```
P3 = por["P3"]
sns.distplot(P3)
```

Out[45]: <matplotlib.axes._subplots.AxesSubplot at 0x256b2cdb898>



In [46]:
```
#internet access and grades
sns.scatterplot(x="M3", y="studytime", hue="internet", size=None, data=math, p
alette=["black", "lightgreen"], legend= False)

r_patch = mpatches.Patch(color='black', label='no access')
blue_patch = mpatches.Patch(color='lightgreen', label='have access')
plt.legend(handles=[r_patch, blue_patch])

plt.title("Effect of Internet access on M3 grades")

plt.show()
```



In [47]:
```
#internet access and grades
sns.scatterplot(x="P3", y="studytime", hue="internet", size=None, data=por, pa
lette=["black", "lightgreen"], legend= False)

r_patch = mpatches.Patch(color='black', label='no access')
blue_patch = mpatches.Patch(color='lightgreen', label='have access')
plt.legend(handles=[r_patch, blue_patch])
plt.title("Effect of Internet access on P3 grades")

plt.show()
```