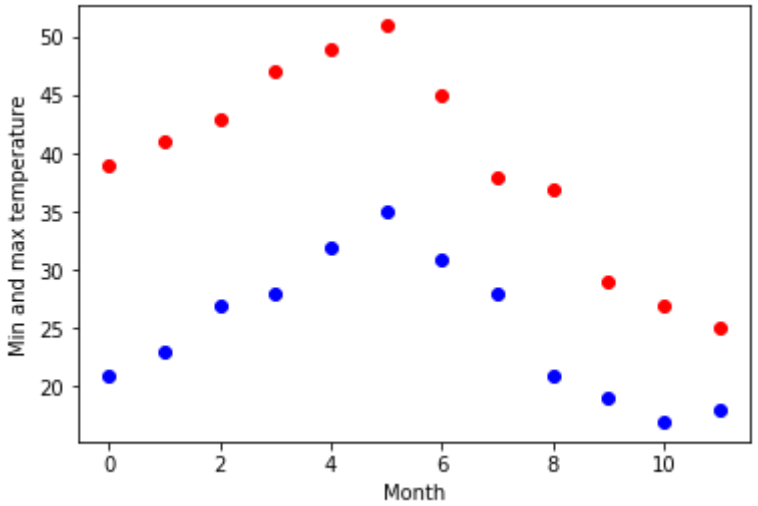
In [1]:

```
"""
Scipy:
We have the min and max temperatures in a city In India for each months of the year.
We would like to find a function to describe this and show it graphically, the dataset
given below.
Task:
1. fitting it to the periodic function
2. plot the fit
Data
Max = 39, 41, 43, 47, 49, 51, 45, 38, 37, 29, 27, 25
Min = 21, 23, 27, 28, 32, 35, 31, 28, 21, 19, 17, 18
"""

import matplotlib.pyplot as plt
%matplotlib inline
import numpy as np
import pandas as pd

Max=[39, 41, 43, 47, 49, 51, 45, 38, 37, 29, 27, 25]
Min=[21, 23, 27, 28, 32, 35, 31, 28, 21, 19, 17, 18]
months = np.arange(12)
days = np.linspace(0, 12, num=365)
plt.plot(months, Max, 'ro')
plt.plot(months, Min, 'bo')

plt.xlabel('Month')
plt.ylabel('Min and max temperature')

```

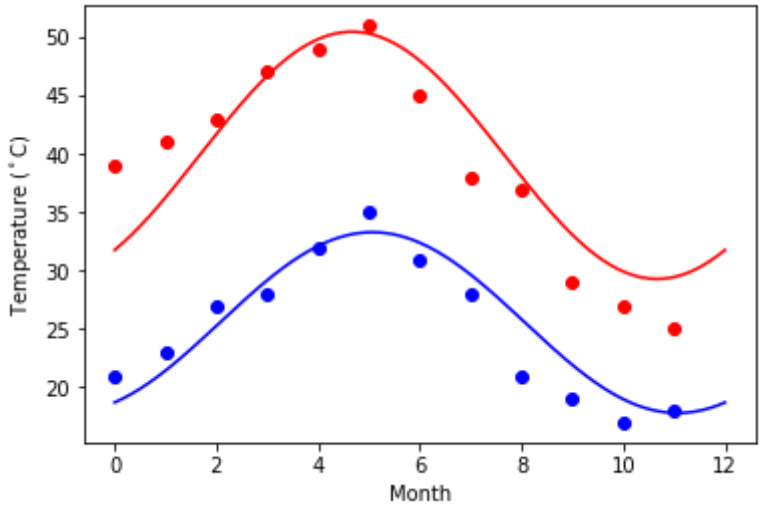
Out[1]:

```
Text(0,0.5,'Min and max temperature')
```


Min and max temperature
Month
50
45
40
35
30
25
20
0
2
4
6
8
10

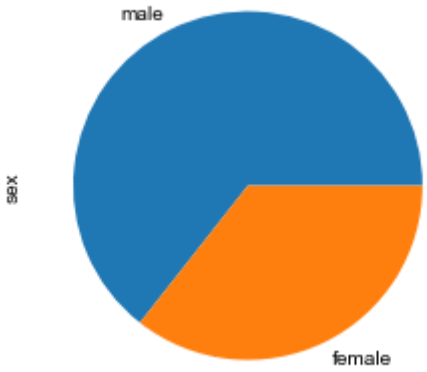
In [2]:

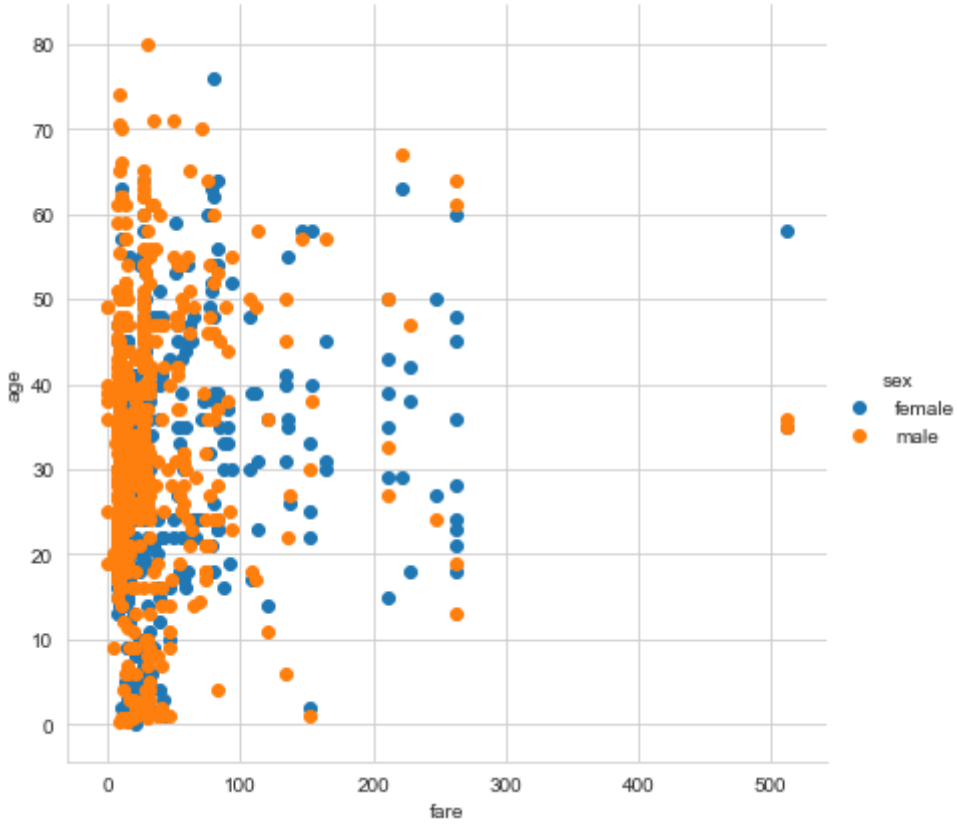
```
from scipy import optimize
def yearly_temps(times, avg, ampl, time_offset):
    return (avg + ampl * np.cos((times + time_offset) * 2 * np.pi / times.max()))
res_max, cov_max = optimize.curve_fit(yearly_temps, months,Max, [20, 10, 0])
res_min, cov_min = optimize.curve_fit(yearly_temps, months,Min, [-40, 20, 0])

days = np.linspace(0, 12, num=365)
plt.figure()
plt.plot(months, Max, 'ro')
plt.plot(days, yearly_temps(days, *res_max), 'r-')
plt.plot(months, Min, 'bo')
plt.plot(days, yearly_temps(days, *res_min), 'b-')
plt.xlabel('Month')
plt.ylabel('Temperature ($^\circ$C)')
plt.show()
```



In [3]:

```
"""
Matplotlib:
This assignment is for visualization using matplotlib:
data to use:
url=
https://raw.githubusercontent.com/Geoyi/Cleaning-Titanic-Data/master/titanic_original.cs
v
titanic = pd.read_csv(url)
Charts to plot:
1. Create a pie chart presenting the male/female proportion
2. Create a scatterplot with the Fare paid and the Age, differ the plot color by gender

"""
import pandas as pd
import seaborn as sns
titanic = pd.read_csv("https://raw.githubusercontent.com/Geoyi/Cleaning-Titanic-Data/master/titanic_original.csv")
titanic['sex'].value_counts().plot.pie()
plt.gca().set_aspect("equal")
#titanic.plot(kind='scatter', x='Fare paid', y='age') ;
#plt.show()
sns.set_style("whitegrid");
sns.FacetGrid(titanic, hue="sex", height=6).map(plt.scatter, "fare", "age").add_legend();
```




80
70
60
50
40
30
20
10
0
age
0
100
200
300
400
500
fare
sex
female
male

In [ ]:
```
1
```