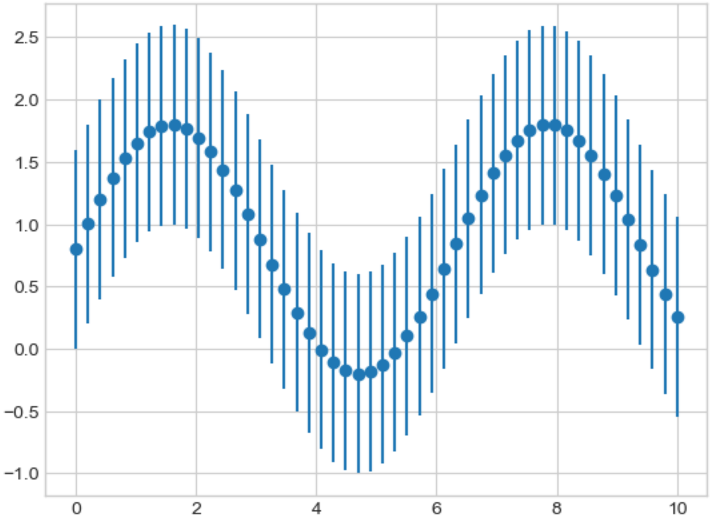
```
In [16]: #visualising errors in matplotlib
```

```
In [17]: #basic errorbars
         %matplotlib inline
         import matplotlib.pyplot as plt
         plt.style.use('seaborn-whitegrid')
         import numpy as np
```

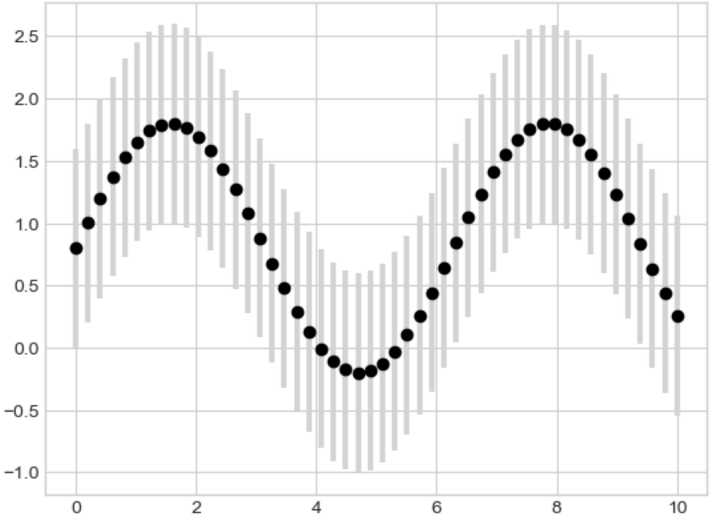
```
In [18]: x=np.linspace(0,10,50)
         dy=0.8
         y=np.sin(x)+dy
         plt.errorbar(x,y,yerr=dy,fmt="o")
```

```
Out[18]: <ErrorbarContainer object of 3 artists>
```



```
In [19]: x=np.linspace(0,10,50)
         dy=0.8
         y=np.sin(x)+dy
         plt.errorbar(x,y,yerr=dy,fmt="o",color="black",ecolor="lightgrey",elinewidth=3,capsize=0)
```

```
Out[19]: <ErrorbarContainer object of 3 artists>
```

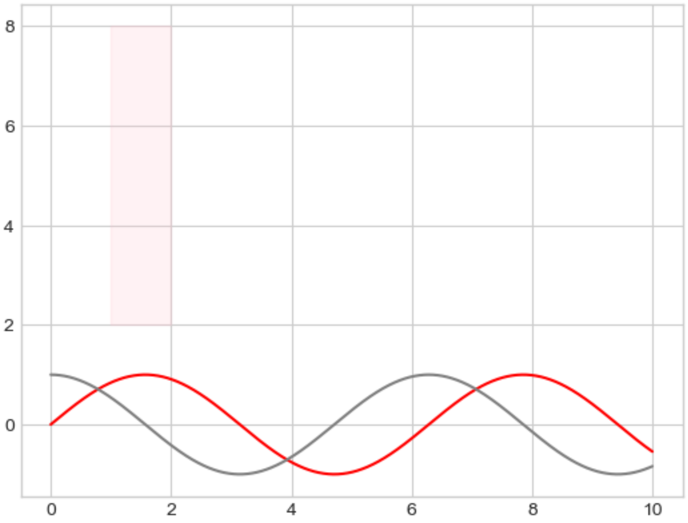


In [38]:

```
x=np.linspace(0,10,1000)
x1=np.linspace(0,10,10000)
plt.plot(x,np.sin(x),"r")
plt.plot(x1,np.cos(x1),'-',color='gray')
plt.fill_between([1,2],2,8,color="pink",alpha=0.2)#filling area that has x coordiantes from 1 to 2 and y coordiantes from 2 to 8
```

Out[38]:

```
<matplotlib.collections.PolyCollection at 0x26ebb9cc460>
```

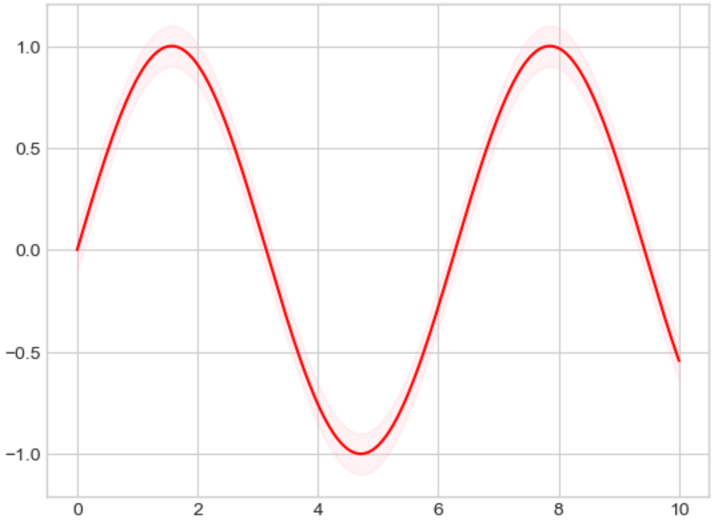


In [50]:

```
x=np.linspace(0,10,1000)
x1=np.linspace(0,10,10000)
plt.plot(x,np.sin(x),"r")
plt.fill_between(x,np.sin(x)-0.1,np.sin(x)+0.1,color="pink",alpha=0.2)#error of 0.1
#filling error area by giving y1 and y2 with y minus error and y plus errors
#we are using errorbars as we would have 10000 points and hence 10000 error bars which would make the plot messy
```

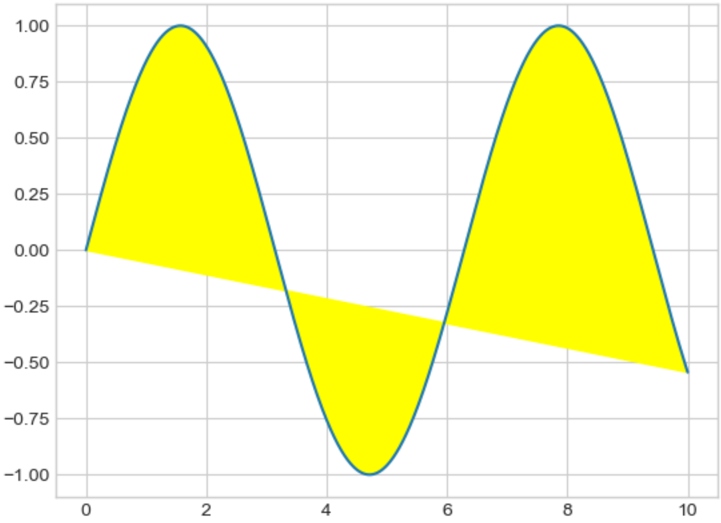
Out[50]:

```
<matplotlib.collections.PolyCollection at 0x26ebcd36eb0>
```



```
In [56]: #fill function in matplotlib
         x=np.linspace(0,10,1000)
         plt.plot(x,np.sin(x))
         plt.fill(x,np.sin(x),c=(1,1,0))
```

```
Out[56]: [<matplotlib.patches.Polygon at 0x26ebd093160>]
```


1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
2
4
6
8
10