```
import matplotlib.pyplot as pltfrom matplotlib import cm, colorsfrom mpl_toolkits.mplot3d import Axes3Dimport numpy as npfrom scipy.special import sph_harmphi = np.linspace(0, np.pi, 100)theta = np.linspace(0, 2*np.pi, 100)phi, theta = np.meshgrid(phi, theta)# The Cartesian coordinates of the unit spherex = np.sin(phi) * np.cos(theta)y = np.sin(phi) * np.sin(theta)z = np.cos(phi)m, l = 2, 3# Calculate the spherical harmonic Y(l,m) and normalize to [0,1]fcolors = sph_harm(m, l, theta, phi).realfmax, fmin = fcolors.max(), fcolors.min()fcolors = (fcolors - fmin)/(fmax - fmin)# Set the aspect ratio to 1 so our sphere looks sphericalfig = plt.figure(figsize=plt.figaspect(1.))ax = fig.add_subplot(111, projection='3d')ax.plot_surface(x, y, z,  rstride=1, cstride=1, facecolors=cm.seismic(fcolors))# Turn off the axis planesax.set_axis_off()plt.show()
```