1. Linear Algebra

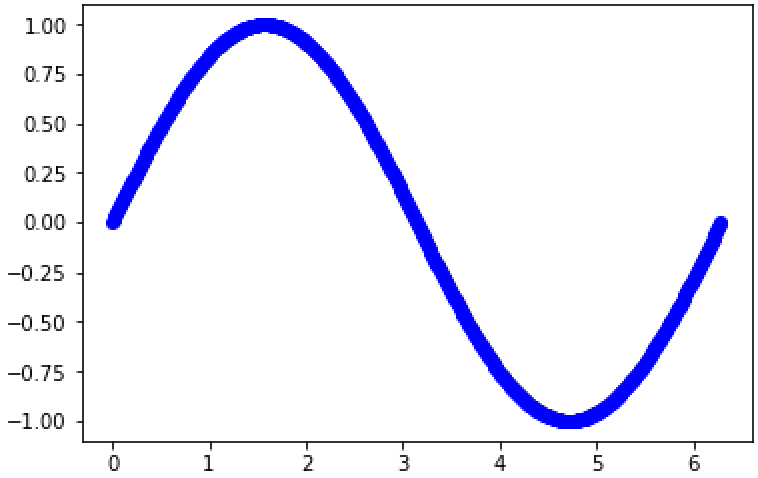
In [74]:

```
# the required imports
import numpy as np
import matplotlib.pyplot as plt
from ipywidgets import interactive
import pywt
```

The sine curve

In [16]:

```
# plot the sine curve
x = np.arange(0, 2*np.pi, 0.01)

plt.plot(x, np.sin(x), 'bo')
plt.show()
```



In [9]:

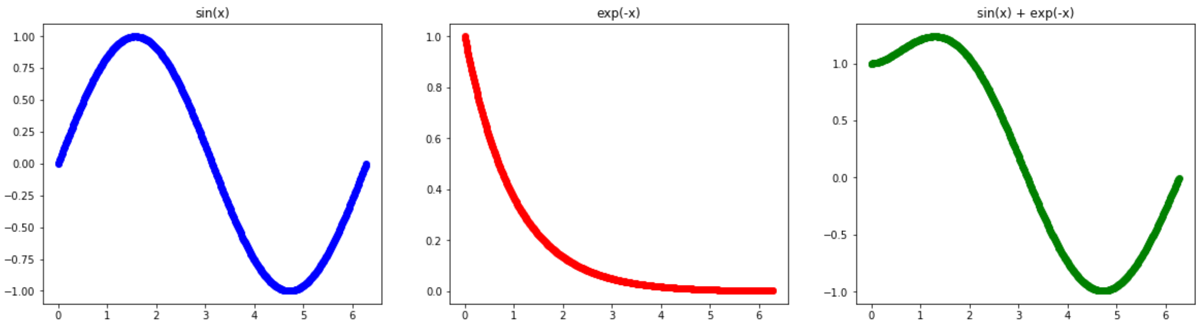
```
# let's explore this 'curve' a bit

def fsine(step):
    plt.figure(1) #* create 1 new figure
    x = np.arange(0, 2*np.pi, step) #* (x-axis,y-axis,step)
    plt.plot(x, np.sin(x), 'bo')
    plt.show()

iplot = interactive(fsine, step=(0.01, 1))
iplot
```

```
interactive(children=(FloatSlider(value=0.505, description='step', max=
1.0, min=0.01), Output()), _dom_classes…
```

Another curve - can we add them?

In [80]:

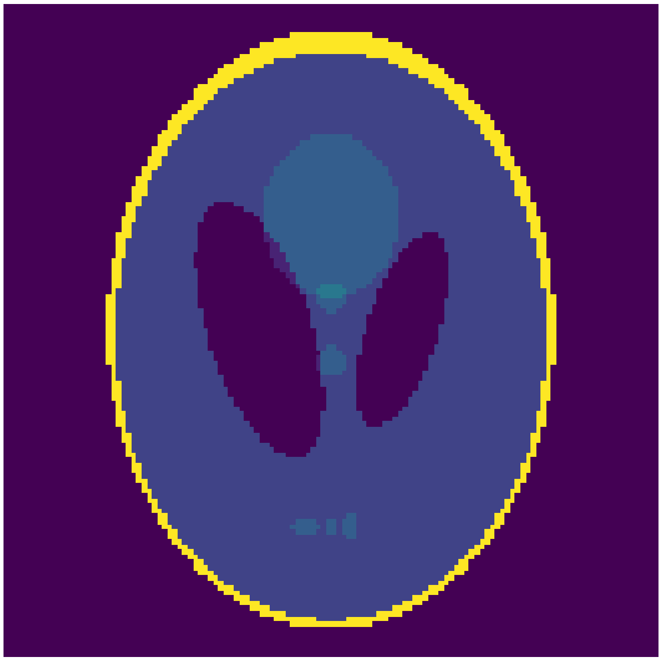
```
# plot more curves
x = np.arange(0, 2*np.pi, 0.01)

plt.figure(figsize=(20,5))
plt.subplot(1, 3, 1, title='sin(x)') #* plt.subplot(nrows,ncols,inde
plt.plot(x, np.sin(x), 'bo')

plt.subplot(1, 3, 2, title='exp(-x)')
plt.plot(x, np.exp(-x), 'ro')

plt.subplot(1, 3, 3, title='sin(x) + exp(-x)')
plt.plot(x, np.sin(x) + np.exp(-x), 'go')
plt.show()
```



What about images?

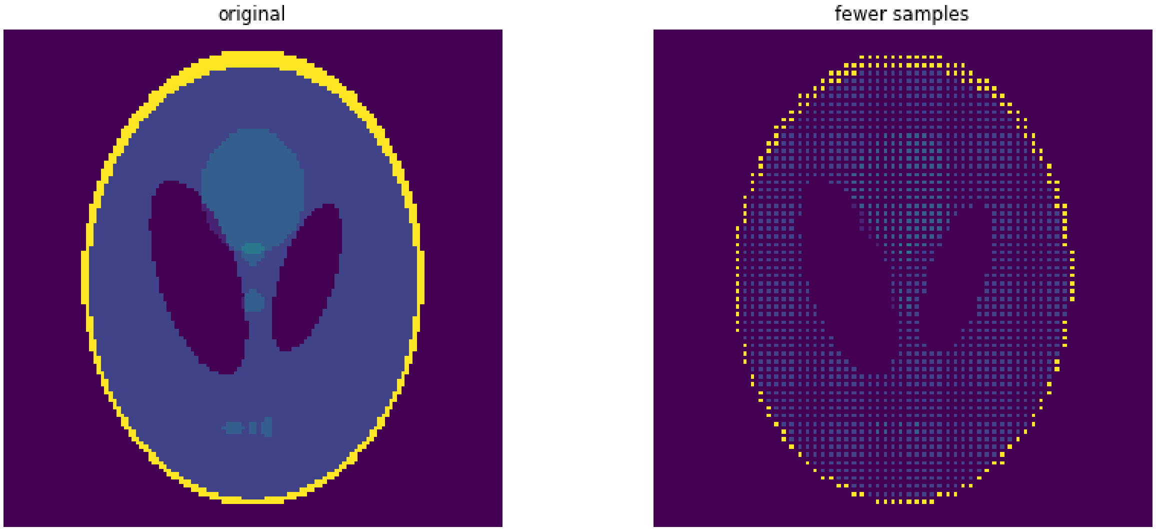
In [83]:

```
# plot an image
phantom = plt.imread('phantom.png')

plt.figure(figsize=(7,6))
plt.imshow(phantom); plt.axis('off')
plt.show()
```



In [82]:

```
# the original
plt.figure(figsize=(14,6))
plt.subplot(1, 2, 1, title='original')
plt.imshow(phantom); plt.axis('off')

# the original with fewer samples
plt.subplot(1, 2, 2, title='fewer samples')
fewer = np.copy(phantom); fewer[::2, ::] = 0; fewer[::, ::2] = 0 # o
plt.imshow(fewer); plt.axis('off')
plt.show()
```



Can we also add images?

In [71]:

```
# add images
flipped = np.flipud(phantom)

plt.figure(figsize=(7,6))
plt.imshow(phantom + 2*flipped)
plt.axis('off')
plt.show()
```

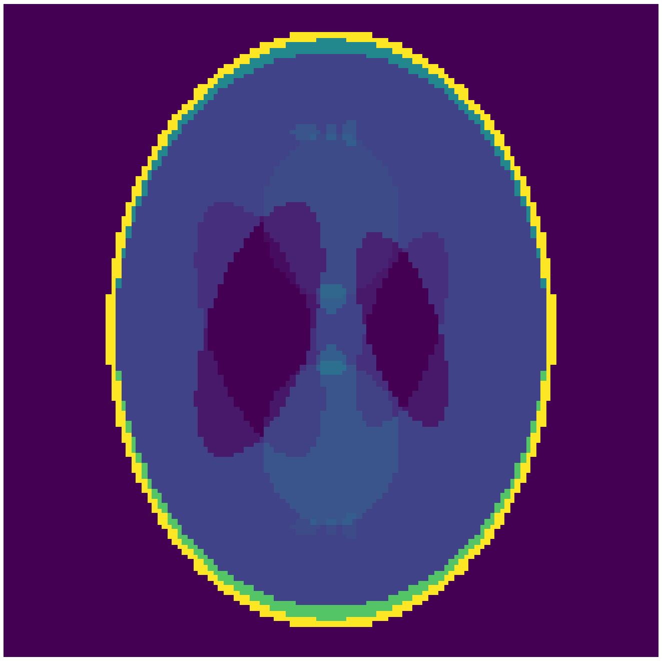
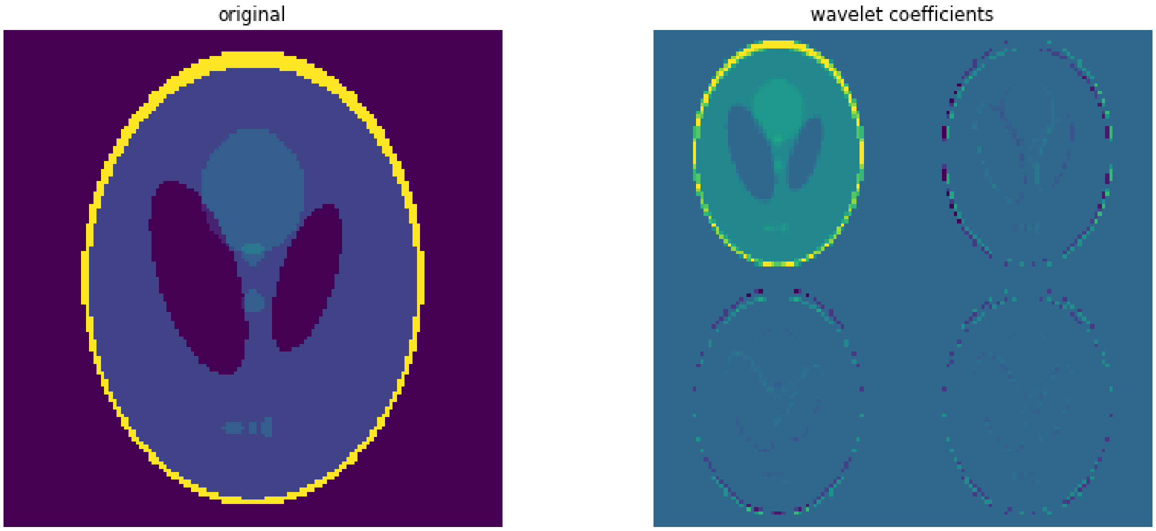


Image representation

In [78]:

```
# try a different image representation -> wavelets!

coeffs = pywt.wavedecn(phantom, 'haar', 'symmetric', level=1) # comp
array, slices = pywt.coeffs_to_array(coeffs) # reorder them as an ar

plt.figure(figsize=(14,6))
plt.subplot(1, 2, 1, title='original')
plt.imshow(phantom); plt.axis('off')

plt.subplot(1, 2, 2, title='wavelet coefficients')
plt.imshow(array); plt.axis('off')
plt.show()
```



original

wavelet coefficients