Saving an Image

When we save the image, we can use a different format from the original. For instance, **JPEG** doesn't have transparent colors, but we can write the image back out as a **PNG**, which does. The ***stb_image_write*** library has different functions for each image type. Here's the code to save our image as **PNG**.

```
stbi_write_png("pete.png", width, height, channels, pete,
        width * channels);
```

Each file type you want to use has its own function, but the first five arguments are the same for each file type:

1. The file name as a C-style string. Here we've hard-coded `pete.png`
2. The `width` and `height` returned from *`stbi_load()`*.
3. The number of channels (or bytes-per-pixel) used in memory to represent the image.
4. The pointer to the first byte of the image data in memory.

The last argument, `width * channels`, is unique to **PNG** files. It tells the function at what byte the next row begins.

Freeing the Memory

The *`stbi_load()`* function returns a pointer, but inside that function it asks the operating system to **allocate enough memory** to hold the image that it loads from disk. This memory is **on the heap**, which you met in an earlier lesson. In the C programming language, you have to remember to **free** that memory before your program ends. We do that by using the function *`stbi_image_free(pete)`*.