Code Foo.

NOTES:

\* The program for Q3 can be started by going to ‘*licenseplategenerator.php’*   
Components: licenseplategenerator.php, licenseplategenerator-ajax.js, licenseplategenerator-style.css, generatedpattern.php

\* The liquid layout can be accessed through *‘liquid.php’*  
Components: liquid.php, liquid-ajax.js, liquid-style.css, getFeed.php, logo.png, Taiga!.png, Gundam!.png, preview\_arrow.png, drop\_arrow.png

\* Connect4 can be started by going to *‘connect4.php’*  
Components: connect4.php, connect4-ajax.js, connect4-style.css, newboard.php

\* ‘*Code Foo – README.txt’* has the same content as this document.

---

Question 1: 'How many ping pong balls would it take to fill an average-sized school bus? Describe each step in your thought process.'

Answer:

I would first find out the volume of what is considered to be an 'average-sized school bus' along with the diameter of the ping pong balls being used in this problem.

Then, I would calculate the volume of a cube based on the diameter of the ping pong balls. I would do this because if you’ve ever attempted to pack balls tightly together, say in a 2-by-2 cube, you’ve probably noticed that there is some wasted space where the surface of each ball curves away from the widest point. Not only is a cube's volume more accurate in this situation, it is also much simpler to deal with, as its value would most likely be an integer.

Next, I would check whether or not the volume given for the bus took into account the space occupied by the seats and other structures built into the bus such as the driver area. If not, I would need to do the accounting for myself, finding out and subtracting their occupying volume from the total volume of the bus.

After all that is done, it is a simple process of taking the available volume space within the bus and dividing it by the volume of the cube representing our 'ping pong balls', and taking the floor of that value to assure that we arrive at an integer value for the number of ping pong balls that can fit inside the school bus without exceeding its available space.

Once we arrive at that value, we have found our answer to the problem.

---

Question 2: 'Explain why you would use a liquid layout.'

Answer:

If I were designing a site intended to be a nexus of information, such as a news network, an RSS reader or an online encyclopedia, I would definitely use a liquid layout. These kinds of sites in particular leverage the advantage of liquid layouts in being able to efficiently utilize the entire screen space as they are expected to display lots of pertinent information to the user all at once. This is because they are designed with major areas of the site defined by percentage-based widths, allowing the layout to expand and contract as needed to fit the user’s screen resolution while at the same time maintaining its overall design. Heavy amounts of text-based information are a natural fit as content for a liquid site, because text naturally flows to match a container's dimensions.

Although it is more difficult, it is also possible to work with images in liquid layouts by scaling them with some Javascript and CSS techniques. However, working with many images eventually leads to cases where one must add a minimum width to containers that are being used to display large images in order to preserve the image quality, or even to allow the image to fit within the container at all.

In a real world situation where I would be dealing with mixed media, I would not use a strictly liquid layout design. Rather, I would used a hybrid of liquid and fixed-width layouts, defining at least a minimum width for the main column to accommodate larger, or poster-sized images and have the other less-essential and text-heavy areas of the site be elastic enough to fill up the remaining screen space. Alternatively, there are gallery-viewing technologies that can be used to view the full-sized versions of an image in a 'popup' frame from a 'thumbnail' version that sits on the page proper that can potentially mitigate the issue of handling images.