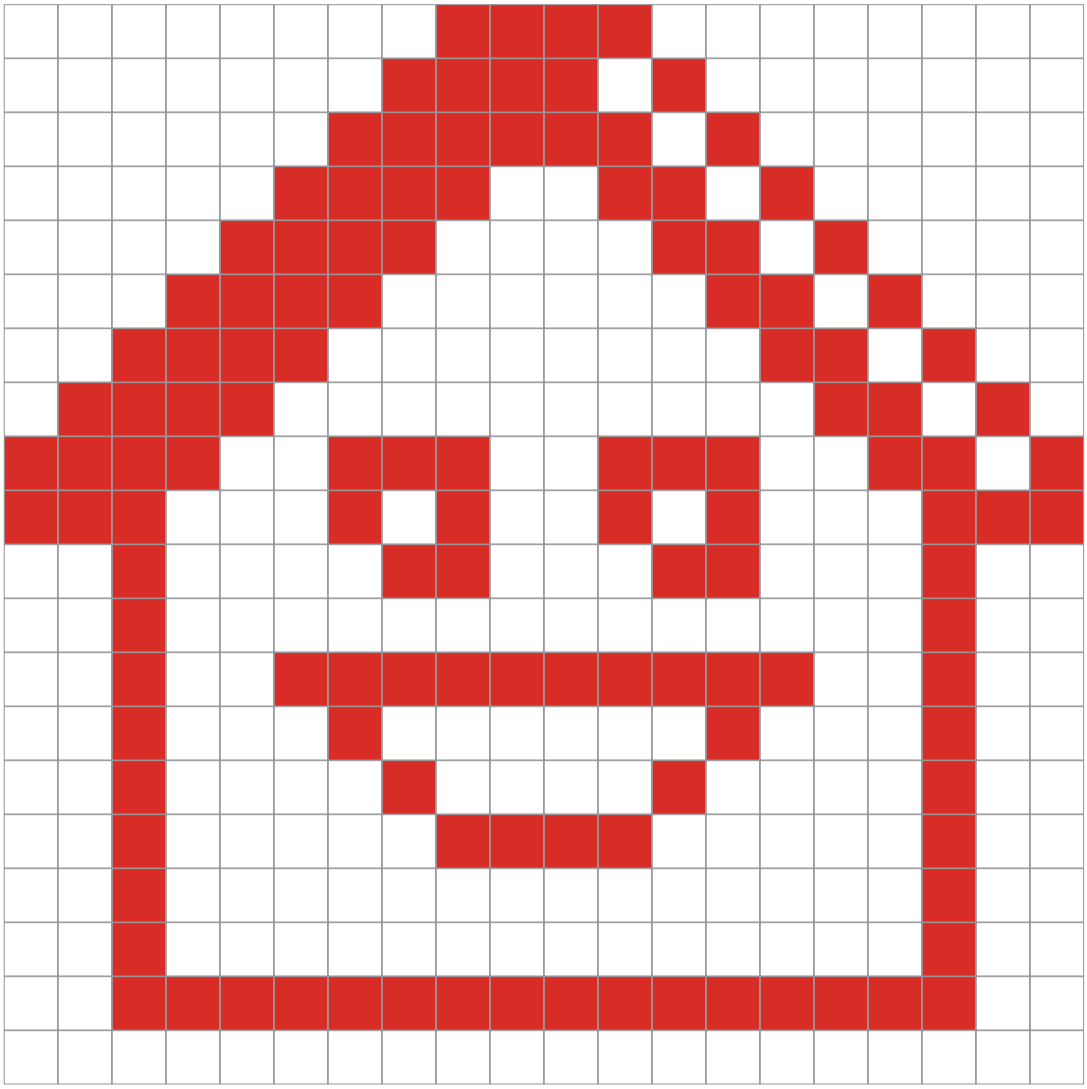
Digital Game Design 1

Image Arrays for Step Two

PIXEL CHARACTER

Look at the pixel character below. Then, look at the array on the next page.



Turn the page to see the array to create the character!

PIXEL CHARACTER ARRAY

It has been formatted to show each row of pixels below. Ones are green, zeros are blanks as laid out below. First, the imageSize= function tells you the size of the grid for the image, and then the characterIcon= function tells you how to fill in the pixels.

This array creates the Pixel Character:

imageSize = [20, 20] // this tells you the grid has 20 rows and 20 columns

characterIcon =

[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0,
0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0,
0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0,
0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0,
0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0,
1, 1, 1, 1, 0, 0, 1, 1, 1, 0, 0, 1, 1, 1, 0, 0, 1, 1, 0, 1,
1, 1, 1, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 1,
0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 0,
0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0,
0, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 1, 0, 0,
0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0,
0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0,
0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0,
0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0,
0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0,
0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0,
0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

THINGS TO KNOW:

This grid can represent a 20x20 pixel image.

- Computer screens come in different sizes, but a programmer can set the size of her arrays.
- Digital images can be stored different ways, but every way contains information about how to display specific pixels.
- A still image can be stored as a JPEG or PNG, for example. These file types use their own kinds of arrays. Often their code is shortened using a coding trick called compression, making them smaller files. Why could that be useful?
- This image's array is a really long set of numbers, and a larger image would be even longer! Actual pixels are so small that if an image was really only 20x20 pixels, it would be too small to see clearly!
- You can think of each square in this grid as a **bit**, the smallest form of digital information. It can only be colored or blank, a one or a zero.

MYSTERY ARRAY

Your challenge: Using the array below, create the image.

You can do this by:

- Coloring in the grid below, or
- Creating the image using 2 colors of sticky notes or squares of paper, or
- Creating the image using just one color of paper or sticky notes. Do this by leaving blank spaces roughly the same size as the paper or sticky where the 0's go — this is a nice way to show that a pixel is "on" or "off," since the 0's are really just the absence of a color.

Image array shown without formatting:

myArray = [1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 1, 0, 1, 0, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 0, 0, 1, 0, 0, 0, 1, 1, 0, 1, 0, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1];

Image array shown with line breaks:

```
myArray = [ myArray = [
        1, 1, 1, 0, 1, 1, 1,
        1, 0, 1, 0, 1, 0, 1,
        1, 0, 1, 0, 1, 0, 0,
        1, 1, 1, 0, 1, 1, 1,
        0, 0, 1, 0, 0, 0, 1,
        1, 0, 1, 0, 1, 0, 1,
        1, 1, 1, 0, 1, 1, 1
];
```