Brief Description and Concept:

My dancing object is a worm dancing disco. Its body is twisting and will slightly vibrate at the end of each round like what people do in disco. Each circle that forms its body changes when the program is restarted, as well as its color scheme. Without restarting the program, its color is flowing. When a user presses the mouse, the worm will stretch.

Visual Documentation:

Development and Technical Implementation:

I started with trying the vibration effect of one circle. After that I made a pile of circles doing the vibration, but the enlarged multiple of the sin function gradually changed. After finishing the dancer’s movement, I wanted to make its look exciting, so I added the code that assign random value to the size of each circle, and making the color flow.

Coding:

1. The biggest challenge to build my dancer was to make the spring back effect. Firstly, I used sin function to control the X-position of the circle. Then I thought it might work if the changing radians could “jump back” (be smaller) when it got to the vibrating point. I kind of worked, but the circle couldn’t return. I solved this by changing the moving speed of the circle from opposite to negative and it worked this time.
2. Another issue that I met was I accidently put the [code] inside for loop, and it turned out that the this.deg increased 10 each frame. I haven’t understood why yet.

Reflection:

What is the benefit of your class not relying on any code outside of its own definition?

Reuse

It will be easy if I want to reuse my dancing object in other codes. I can just copy the whole code of the dancing object class and write a few codes in the setup and draw function, without worrying about missing some parts of code or copying something I don’t need in the new program.

What make it challenging to write code that has to harmonize with the code other people have written?

The people involved need to write in the same language or making some bridge between different languages.

Cooperating people can name the parameters that refer to different variables the same name.

We might not understand the function of some parts of other people’s code.

Describe the terms modularity and reusability using the example of your dancer class.

I think modularity could be shown as my dancer class was split into update function and display function as two modules, which were built separately. The instance coming from my dancer class could use both functions or one of the functions. Reusability was interpreted as the class I made could be used to make many other instances by calling the function in the setup and/or draw function with a few lines of code.