

Enunciado de la actividad PAC3

Actividad evaluable

Individual

Inicio

16/04/24

Entrega

06/05/24

Calificación

21/05/24

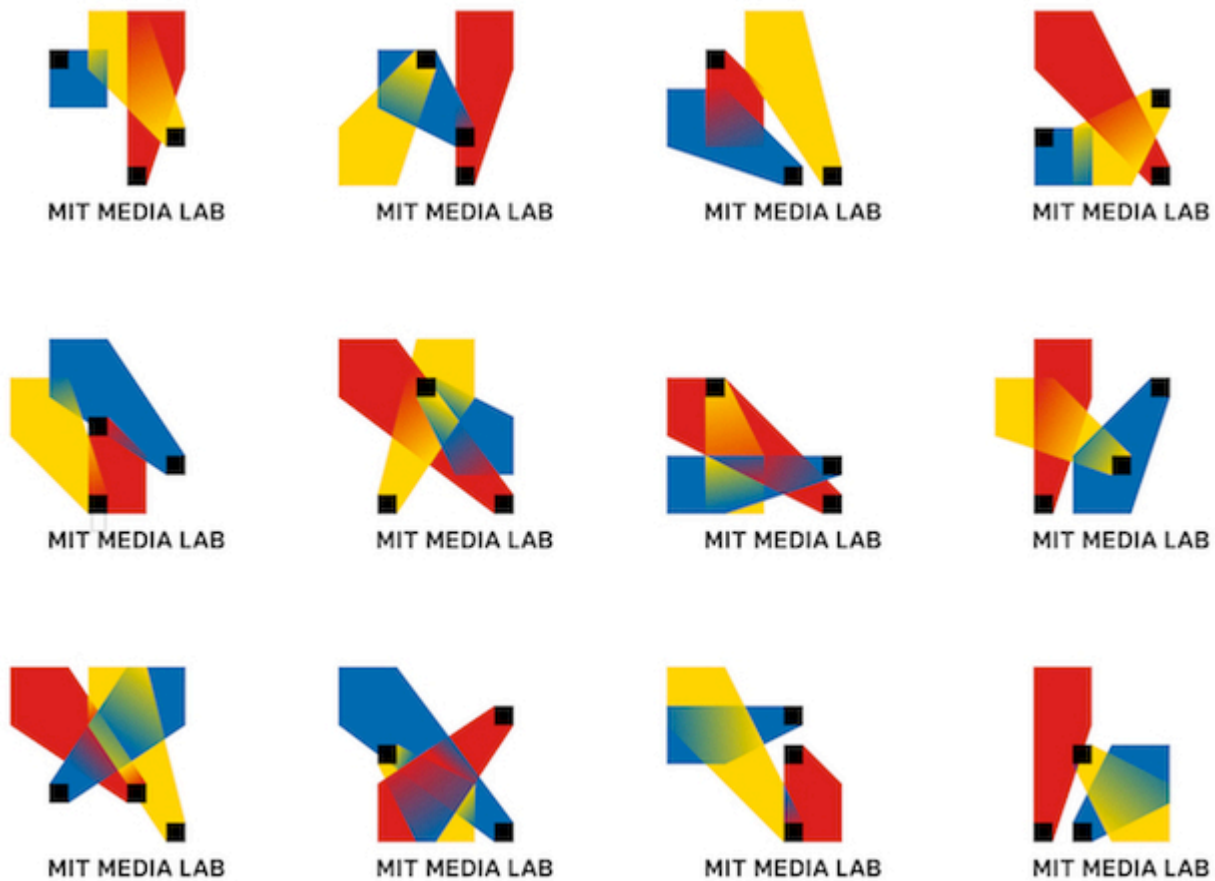
PAC 3

En esta PEC se plantea diseñar un modelo mediante un diseño paramétrico y animarlo mediante una interacción basada en la modificación de esos parámetros. Un diseño paramétrico consiste en crear un modelo cuya forma y funcionamiento dependen de unos parámetros determinados y unas reglas que determinan, a partir de estos parámetros, el resto de sus características y su organización.

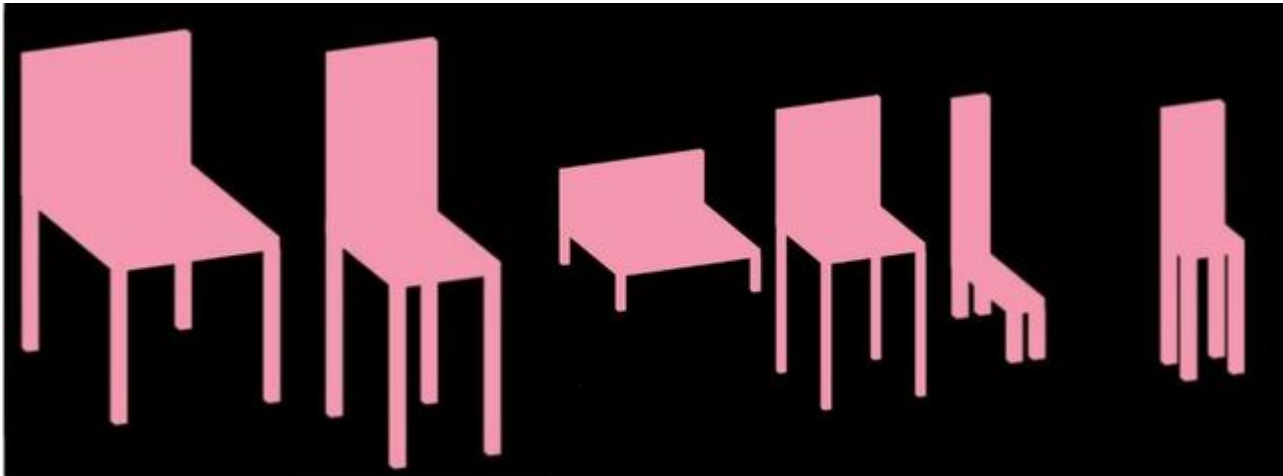
Por ejemplo, se puede plantear una escena con uno o varios elementos en el que las dimensiones de sus rasgos están interrelacionadas y dependen exclusivamente de unos pocos parámetros: posición, giro, tamaño, estado...

Algunos ejemplos de diseños paramétricos:

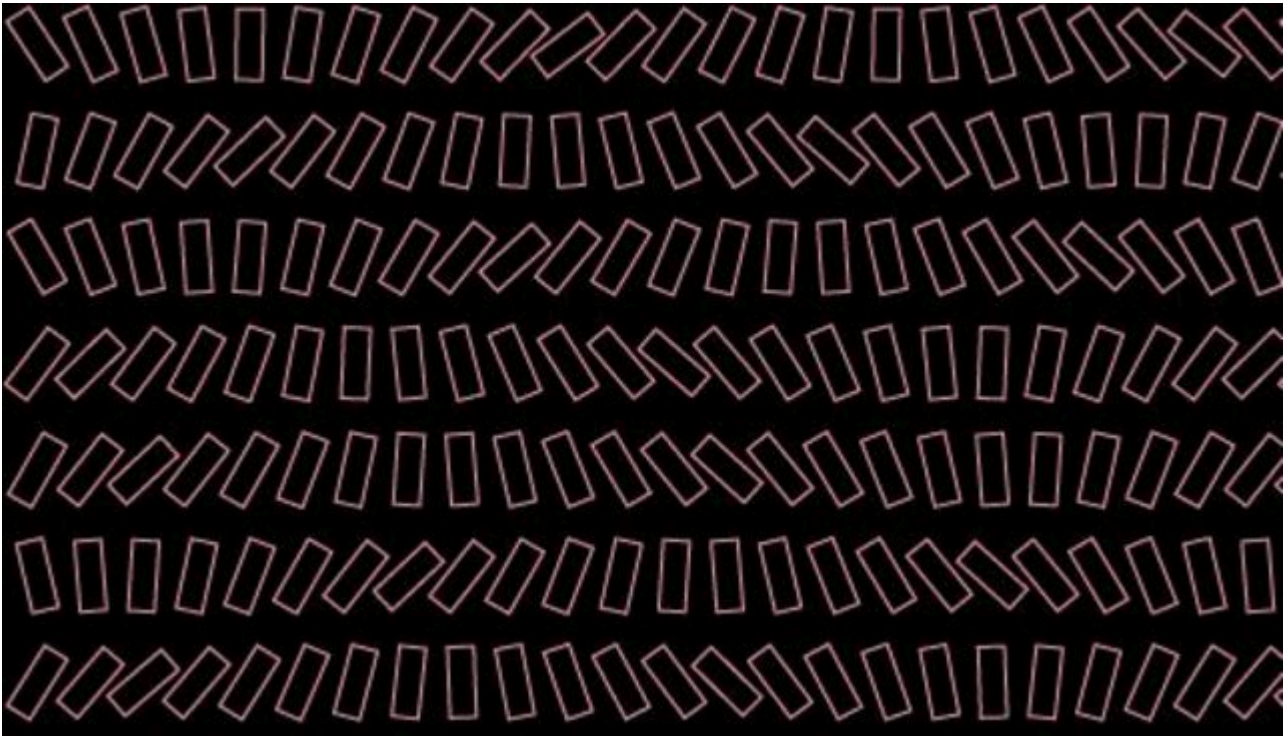
- **logo MIT Media Lab Identity** (enlace, otro enlace)



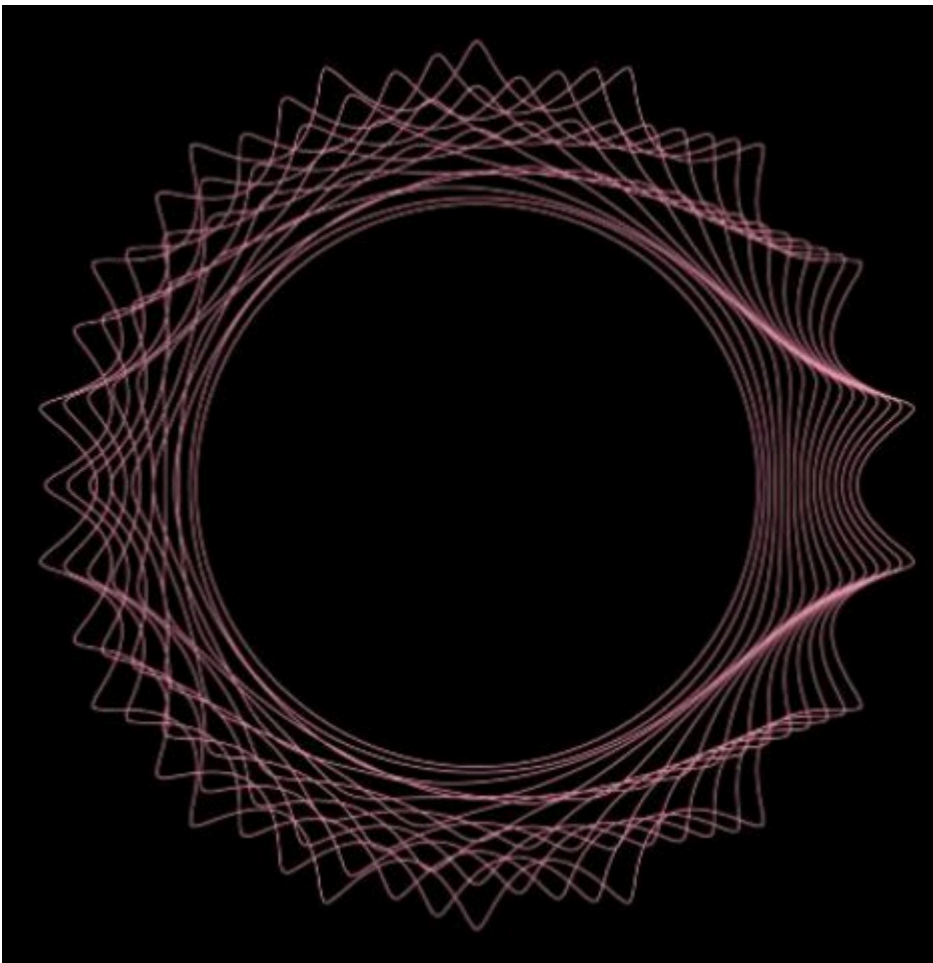
- **Sillas** (Form And Code, enlace)



- **Wave** (Form And Code, enlace)



- **Superformula** (Form And Code, enlace). (wiki) (demo en p5.js) (demo en codepen)



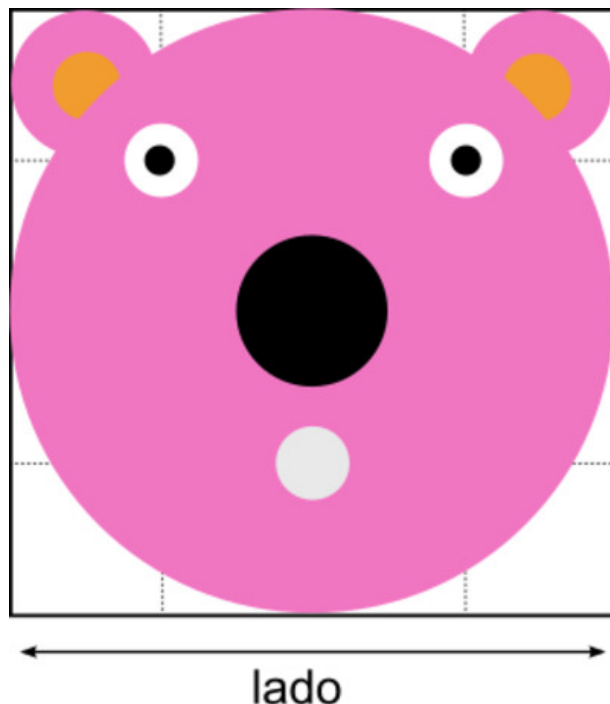
Les proponemos seguir los siguientes pasos para desarrollar la PEC:

1. Leer el **apartado 9 “Funciones”** del recurso de aprendizaje **“Programación en Processing: actividad práctica”** para familiarizarse con el uso de funciones, en particular las funciones con parámetros.
2. Leer el **apartado 4.3 “Objetos”** del **“Cuaderno de programación Creativa”** para aprender a diseñar clases e instanciar objetos a partir de ellas. Esto puede ayudar a diseñar un modelo basado en parámetros.
3. Leer el **apartado 4.6 “Traducción y Rotación”** del **“Cuaderno de programación Creativa”** para aprender a transformar geométricamente una serie de comandos gráficos, asignándoles una posición mediante un cambio de coordenadas y una rotación respecto al origen de coordenadas.
4. Realizar el **ejercicio preliminar** descrito en el siguiente apartado. Se trata de un ejercicio para crear de forma guiada un sencillo modelo regido por pocos parámetros. Este ejercicio no será evaluable. Se anima a subir una imagen del resultado al foro para compartir con los compañeros.
5. Realizar el **ejercicio evaluable**.

Descripción de las actividades a realizar

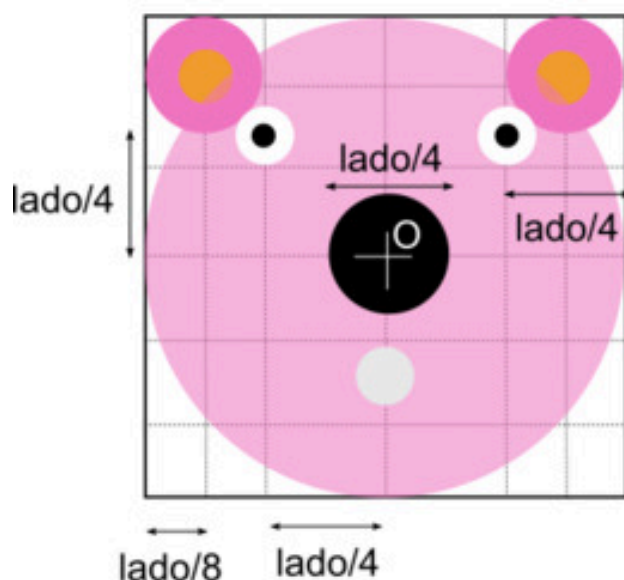
a) Ejercicio preliminar (no evaluable)

Otro ejemplo práctico y muy básico sería este diseño de un oso como el de la siguiente imagen.



El oso estaría completamente representado por el parámetro **lado** y los **colores de algunas de sus partes**, por ejemplo de la piel, de las orejas y de la boca. El parámetro lado determina el resto de dimensiones del modelo (tamaño y posición de boca, orejas, ojos, nariz...). También haremos que el tamaño de la pupila sea independiente y que pueda variar entre 0 y el tamaño del ojo.

En la siguiente imagen se muestran los parámetros que definen al oso:



Posiciones:

$\text{posNariz} = (0,0)$
 $\text{posBoca} = (0, \text{lado} * \text{fBocaY})$
 $\text{fBocaY} = 0.25$
 $\text{posOjoD} = (\text{lado}/4, -\text{lado}/4)$
 $\text{posOrejaD} = (\text{lado} * 3/8, \text{lado} * 3/8)$

Diametros:

$\text{diamNariz} = \text{lado}/4$
 $\text{diamBoca} = \text{lado}/8$
 $\text{diamOjo} = \text{lado}/8$
 $\text{diamPupila} = \text{diamOjo} * \text{fPupila}$
 $\text{fPupila} = 0.5$
 $\text{diamOreja} = \text{lado} * 1/4$
 $\text{diamOrejaInt} = \text{diamOreja} * \text{fOreja}$
 $\text{fOreja} = 0.5$

Colores:

$\text{cPiel} = \text{color}(240,120,200)$
 $\text{cNariz} = \text{color}(0)$
 $\text{cBoca} = \text{color}(180)$
 $\text{cOjo} = \text{color}(255)$
 $\text{cPupila} = \text{color}(0)$
 $\text{cOreja} = \text{color}(255,120,0)$

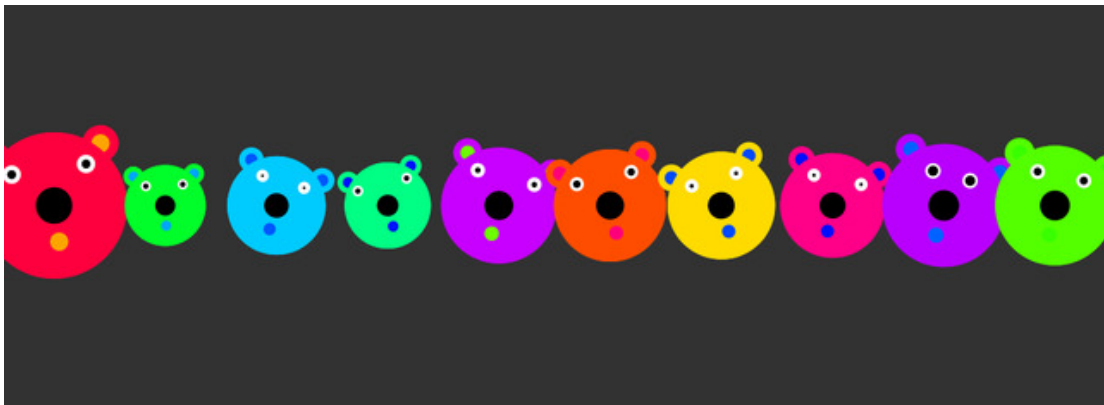
El modelo queda definido con la variable **lado**, que redimensiona todas las características principales del oso, y con las **variables globales fPupila, fOreja, fBocaY...** que establecen la proporción entre ciertas partes del oso. Los **colores** también son variables.

Además, tendría parámetros comunes como son la **posición** y su **rotación** respecto al su centro (hocico). Estas las podemos utilizar con las transformaciones **translate()** y **rotate()**, que se describen en el **apartado “4.6 - Transacción y Rotación” del “Cuaderno de programación Creativa”**.

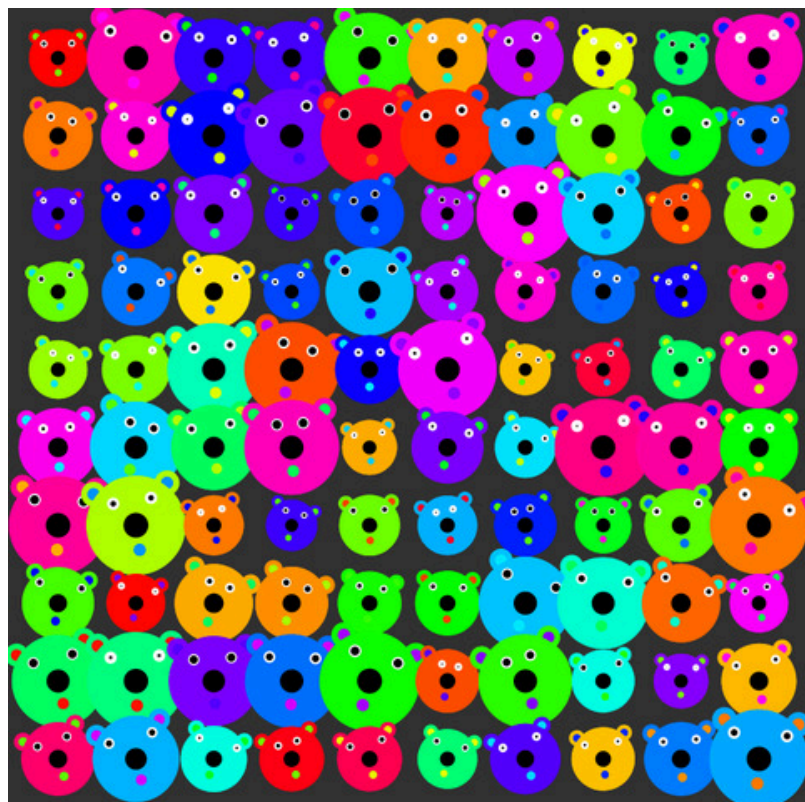
Jugando con los valores de las variables e interactuando con ellas se pueden generar composiciones como las siguientes:



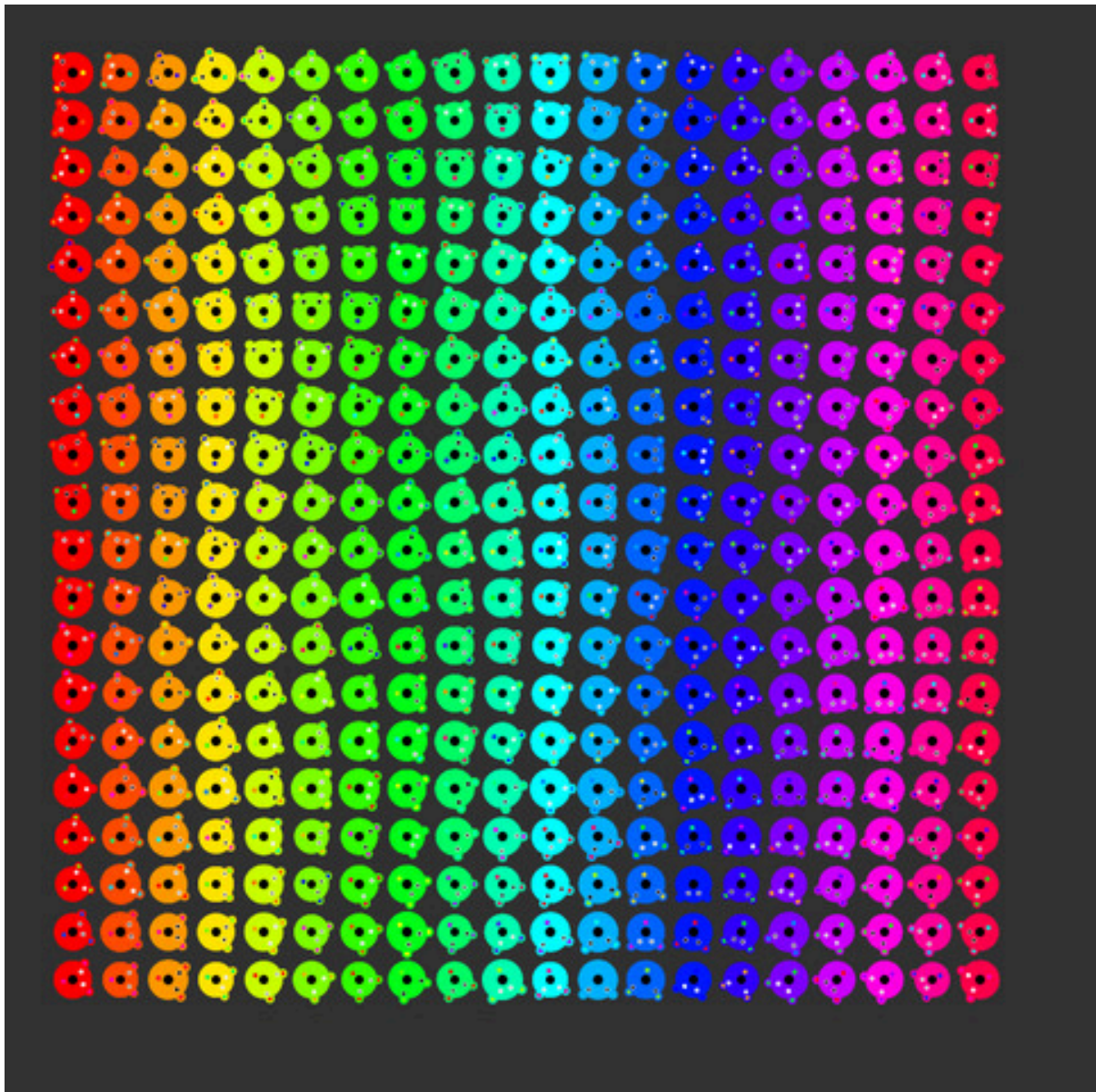
3 osos diferentes



Una fila de osos con características aleatorias



Grid de osos con propiedades aleatorias



Grid de ositos. La rotación y color depende de la posición

INSTRUCCIONES

1. Crea un sketch con un tamaño de 600x600 y el color de fondo que nos apetezca.
2. Define las variables globales de las que depende nuestro modelo: lado (tipo float), colorPiel, colorBoca, colorPupila (las 3 tipo color), y fPupila, fOreja y fBoca (las 3 tipo float y con valores entre 0 y 1). También habrá que crear variables para la posición: px, py.
3. En setup() asigna valores a las variables.
4. Construye una función dibujarOso() sin parámetros que dibuje el oso con las variables globales. Las instrucciones iniciales serán similares a las de la imagen del ejemplo, deben calcular las posiciones y dimensiones a partir de los parámetros globales. Posteriormente hay que dibujar el modelo.
5. Llama a esa función desde draw().

6. Una vez que funcione correctamente esa función, crea una nueva función del mismo nombre pero con un parámetro `_lado`. En la función sustituye `lado` por el parámetro de la función `_lado`.
7. Sustituye la función `dibujarOso()` por la nueva `dibujarOso(parametro)` y comprueba si la función redibuja el oso correctamente pero con el tamaño indicado en el parámetro.
8. Cuando funcione correctamente puedes añadir más parámetros que sustituyan a las variables globales.
9. Sube la imagen generada con un oso a la carpeta del foro creada para tal fin. Puedes subir varias imágenes indicando que el oso varía, o bien crear una composición con varias figuras creadas con sucesivas llamadas a la función de `dibujarOso(parametros)`

También es posible, y recomendable, crear una clase `Oso` que guarde esas variables globales como sus atributos y crear métodos para dibujarla.

b) Actividad evaluable

La actividad consiste en diseñar un modelo, escena, avatar... que es completamente representada por un número de variables independientes (al menos 3). A partir de esas variables, algunas otras características (variables) del modelo se recalcularán y actualizarán, modificando la representación y funcionamiento del modelo.

Hay que incorporar alguna forma de **interacción** con la pieza mediante mouse, teclado, GUI... de forma que cambien los valores de algunas de las variables y, a su vez, la representación del modelo.

Una técnica interesante y muy sencilla de aplicar para animar una variable es el uso de la función **map()** para transformar el valor de una variable como pueda ser `mouseX` en un valor adecuado a un rango de valores aceptados por alguna(s) de las variables que definen el modelo.

INSTRUCCIONES

1. Diseña un modelo gráfico cuya representación dependa de, al menos, 3 variables.
2. Crea la función o clase que permita representar el modelo pasándole valores a esas variables principales.
3. Crea una escena en la que haya al menos una representación del modelo.
4. Añade varios tipos de interacción.
5. Comenta las acciones que realizamos en el código.
6. Escribe como comentario en la parte superior del mismo sketch un pequeño párrafo explicando la pieza.

7. Comprime la carpeta de nuestro sketch (con sus posibles complementos) en un zip con nuestra inicial nuestro apellido y “-PEC3” (ej. lsegura-PEC3.zip)
 8. Sube el trabajo al RE
-

Criterios de evaluación

- Código ordenado, limpio y comentado
- Uso de funciones y/o objetos
- Variedad en los modos de interacción
- Creatividad y composición