Randomness

Start With Something We’ve Seen Before

Pure Randomness is Pretty Jarring

A Better Approach – Add a Random Number to the Current Value

Computer Graphics Noise

Noise Octaves Create More Detail

- The built-in noise() function is a smoothly-changing sequence of values
- It returns values from 0. to 1.
- It is centered around 0.5, i.e., the midline
- It can be spread out (made smoother) by making the argument smaller.
- It can be compressed (made more jagged) by making the argument larger.
- It is **Coherent** in that the noise value at one point is close to the noise value at the next point
- Setting noiseSeed() makes it **Repeatable** in that the same input always gives the same output
float NoiseFactor = 200.; // larger to make the noise gentler
int NoiseSeed = 22019; // start the random number sequence
int MinOctaves = 1;
int MaxOctaves = 8;

void setup( )
{
  size(800, 600);
colorMode( RGB );
noFill( );
noiseSeed( NoiseSeed );
}

void draw( )
{
  background(200, 200, 255);
stroke(128, 0, 0);
strokeWeight(1.);
beginShape( );
vertex(0, height/2);
vertex(width, height/2);
endShape( );
for( int octaves = MinOctaves; octaves <= MaxOctaves; octaves = octaves*2 )
{
  noiseDetail(octaves);
  int green = int( map(octaves, MinOctaves, MaxOctaves, 0, 255) );
  stroke(255, green, 0);
  beginShape( );
  for( int x = 0; x < width; x = x + 5 )
  {
    int y = (height/2) + int((height)*(noise(x/NoiseFactor) - 0.5));
    vertex(x, y);
  }
  endShape( );
}
}

float NoiseFactor = 200.; // larger to make the noise gentler
int NoiseSeed = 22019; // start the random number sequence

void setup( )
{
  size(800, 800);
colorMode( RGB );
background(200, 200, 255);
fill(255, 255, 0);
stroke(0, 0, 0);
noiseSeed( NoiseSeed );
noiseDetail(4);
}

void draw( )
{
  if ( mousePressed )
  {
    float nx = noise(mouseX/NoiseFactor);
    float ny = noise(mouseY/NoiseFactor);
    fill(nx*255, ny*255, 0);
    ellipse(mouseX, mouseY, 100, 100);
  }
}
Using 2D Noise to Affect Color

It takes about 40 seconds to do 600x600 = approx 9,000 points/sec.

```cpp
void draw()
{
  for( int x = 0; x < width; x++ )
  {
    for( int y = 0; y < height; y++ )
    {
      noiseSeed(   0 );
      int red = int( 255.*noise( x/NoiseFactor, y/NoiseFactor ) );
      noiseSeed( 1000 );
      int green = int( 255.*noise( x/NoiseFactor, y/NoiseFactor ) );
      noiseSeed( 2000 );
      int blue = int( 255.*noise( x/NoiseFactor, y/NoiseFactor ) );
      stroke( red, green, blue );
      point( x, y );
    }
  }
  //noLoop( );
  //saveFrame( "ColorClouds.png" );
}
```

Here are some fun things to try (make the window size smaller first!):
- What happens if you make NoiseFactor larger? Smaller?
- What happens if you only stroke with (red, green, 0.)?
- What if you only use red and blue? Green and blue?