



letterwaves

letterwaves

Find rythm in the unknown of each wave.

Copyright © 2025 Mia Baedke
No part of this book may be used
or reproduced in any manner
without written permission from
the publisher. Errors or omissions
will be corrected in subsequent
editions.

Book designer - Mia Baedke
Editor - Mia Baedke
Cover designer - Mia Baedke
Primary Typeface - Quincy CF,
Connary Fagen
First print edition 2025
Printed by Blurb.com
Suite 300, 580 California Street
San Francisco, USA
Special thanks to - my cat Aspen

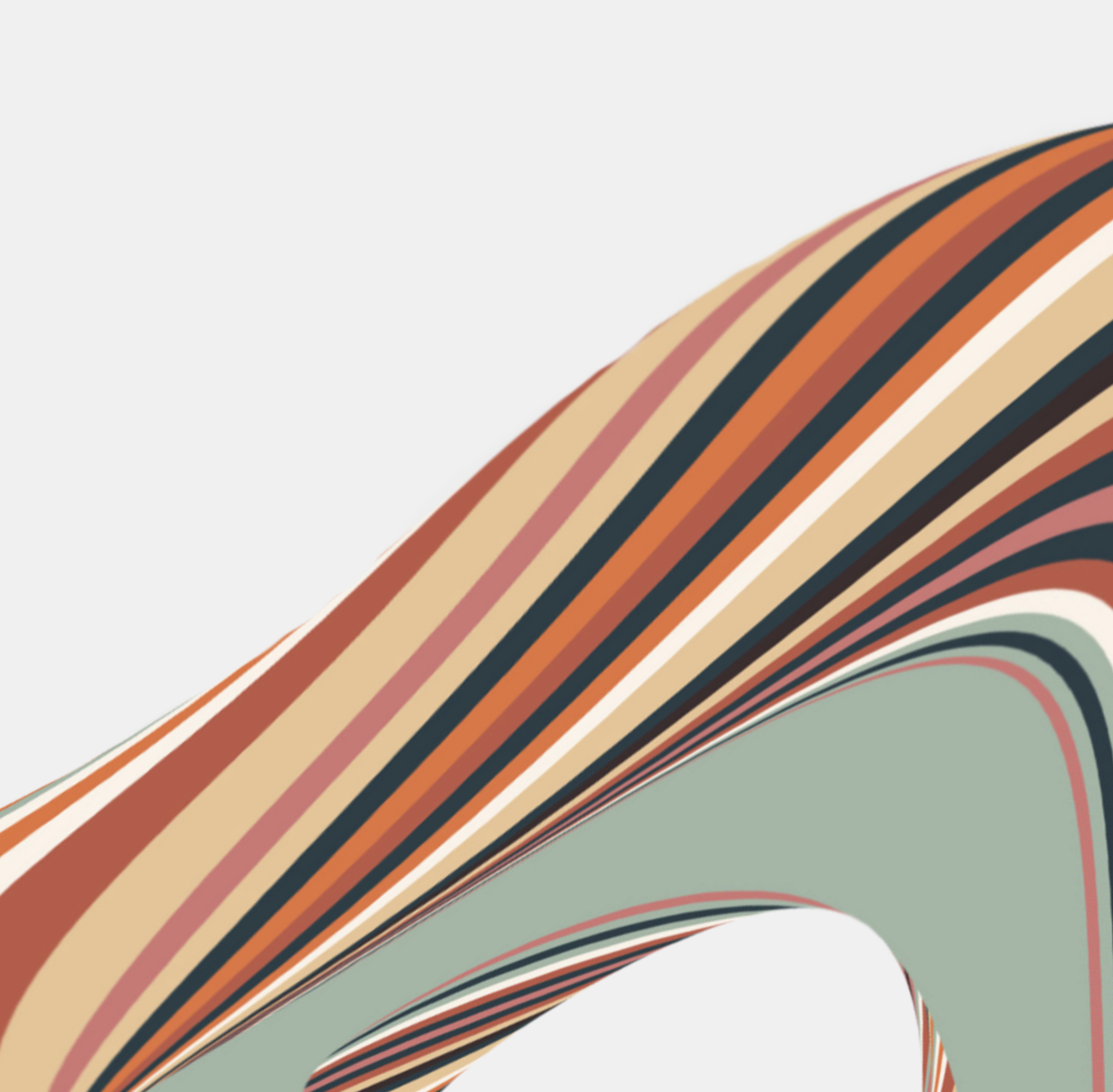




Introduction

Letterwaves explores movement within stillness, how a single letter, when repeated through code, can create rhythm, balance, and flow. Using Processing, this project turns letters into visual waves that guide your eye, each unique by random color and its shape. At first, learning this process was overwhelming as I struggled to see how art can correspond with code, but as I spent time with it, I began to see the creativity behind it unfold. Through this book, I hope to inspire everyone to step out of their comfort zones with art, and try something new.













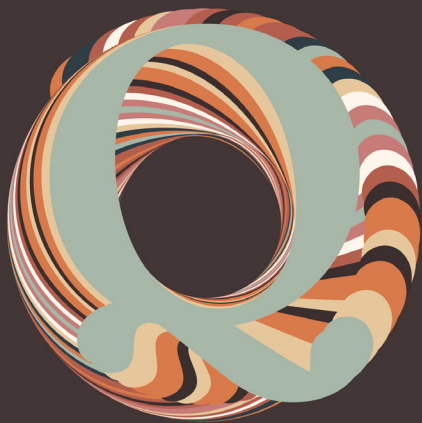




















The Process

For this project, I chose my typeface, Quincy CF. I had many ideas for how it could look, but I knew I wanted it to feel like it was in motion. I didn't want something that felt stuck or still. Using the letterwaves code I created, I got a spiral effect by rotating each letter slightly underneath the other. By stopping the rotation at 180 degrees, the spiral only goes halfway around, leaving open spaces that make the design feel light and balanced. The result is a typeface that feels calm, steady, and always moving.

```

float curveRadius = 300;

fill(240,10);
rect(0,0,width,height);

pushMatrix();
translate(centerX,centerY);
rotate(PI);
translate(-centerX,-centerY);

for(int a = 0; a <= 180; a += 1)
{
  fill(colors[int(random(colors.length))]);

  float angle = radians(a);
  float x = centerX + cos(angle) * curveRadius;
  float y = centerY + sin(angle) * curveRadius;

  pushMatrix();
  translate(x,y);
  rotate(angle + (PI/2));
  text("d",0,0);
  popMatrix();
}

```



```

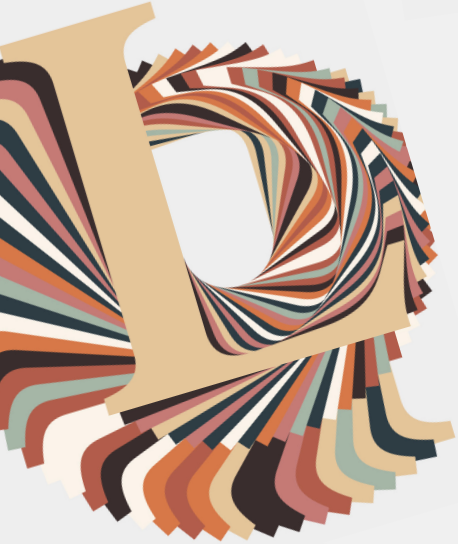
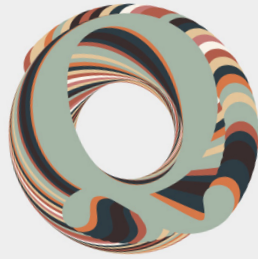
PFont myFont;

void setup()
{
  size(1000,1000);
  background(240);
  noStroke();
  myFont = createFont("Fontspring-DEMO-quince-
of-medium.otf",700);
  textFont(myFont);
  textAlign(CENTER,CENT
  noloop();
}

void draw()
{
  color[] colors = {
    #FFDF6EC, #D97A4A,
    #3B2F2F, #C77C78,
    #A7B7A8, #E6C79C,
    #2F3E46, #B45F4D
  };

  float centerX = width/2;
  float centerY = height/2;
  float curveY = centerY + 100;

```

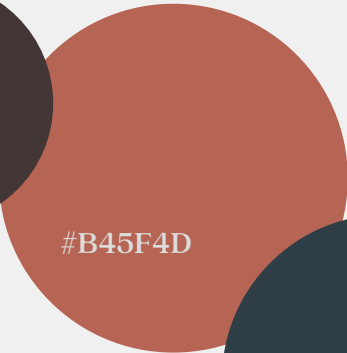


rythmic
organic
harmonious

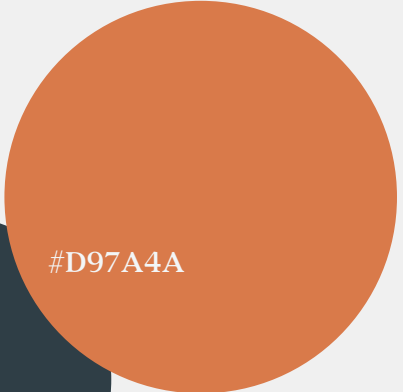




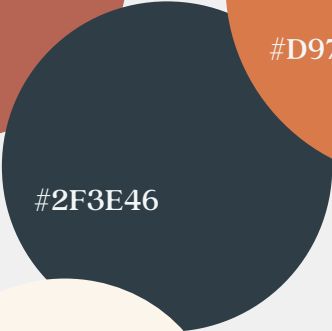
#3B2F2F



#B45F4D



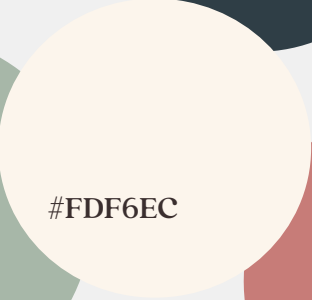
#D97A4A



#2F3E46



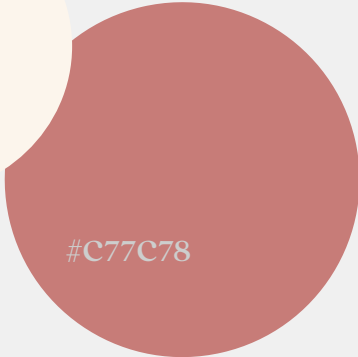
#E6C79C



#FDF6EC

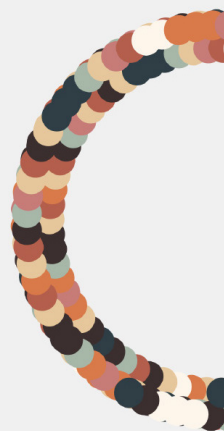


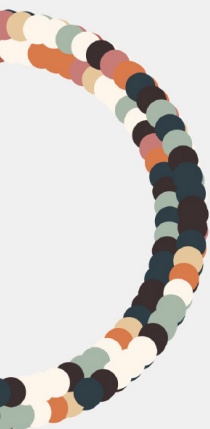
#A7B7A8

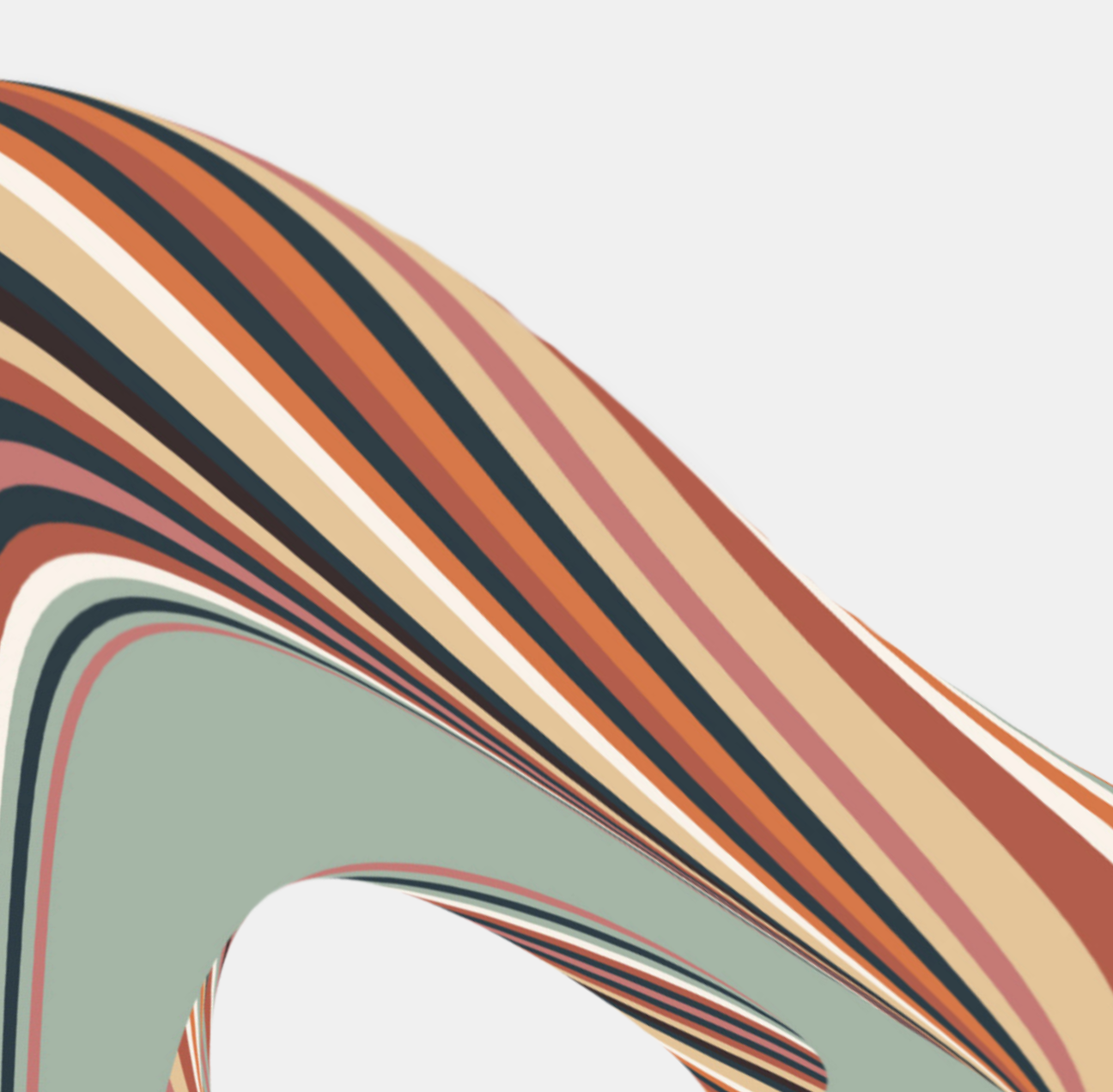


#C77C78

Colors







Conclusion

At first, I was overwhelmed. Learning something new in an unfamiliar program that felt more like math than art was intimidating. But as I took my time and tried new things, everything started to make sense. Like the rhythm of the waves you see in this typeface, steady yet unpredictable, the process of learning to code became its own kind of art form.



LEETE

BOOKS

