PAT 204/504 (Fall 2024)

Creative Coding

Lecture 1: Introduction

Instructor: Hao-Wen Dong

SCHOOL OF MUSIC, THEATRE & DANCE
PERFORMING ARTS TECHNOLOGY
UNIVERSITY OF MICHIGAN

Welcome! Tell Us about Yourself!

- Name
- Pronouns
- Major/year
- Have you ever coded? Which programming language?

About Me

- Hao-Wen (**Herman**) Dong
- Pronouns: he/him
- Email: **hwdong**@umich.edu
- Office: **Stearns 131** (10–15 min walk to the north from Moore)
- Office hours: 3–4pm, Mondays & Wednesdays
- Research areas: Generative AI for music and audio creation

Creative **Coding**

Creative Coding

What is this course all about?

An introduction to principles and practices of computer programming for musical applications. Emphasis is on **creative and artistic uses of code**.

Processing

Max

Learning Goals

- Gain an understanding of **programming for music and multimedia**
- Learn to use programming languages commonly used in artistic projects
- Gain hands-on experience through implementing **audiovisual** systems using creative coding
- Gain a critical comprehension of common concepts and theories employed in creative coding practices

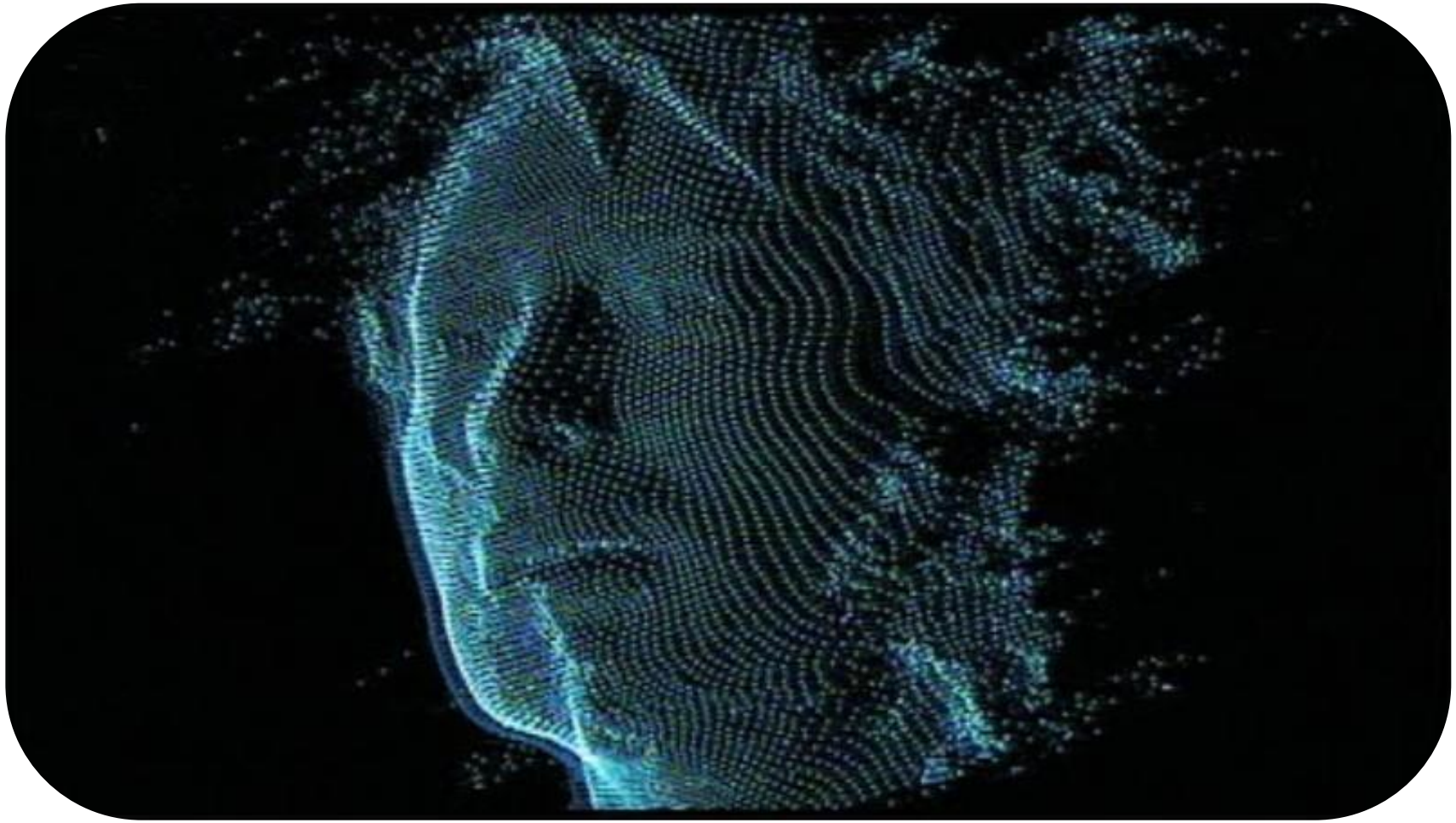
Intro to *Processing*

What is *Processing*?

- A free **programming language** built for **electronic arts**, **new media art** and **visual design**
- Easy to get started with!
- Based on **Java**
  - 3rd most popular programming language on GitHub
  - Can leverage the power of Java

Processing

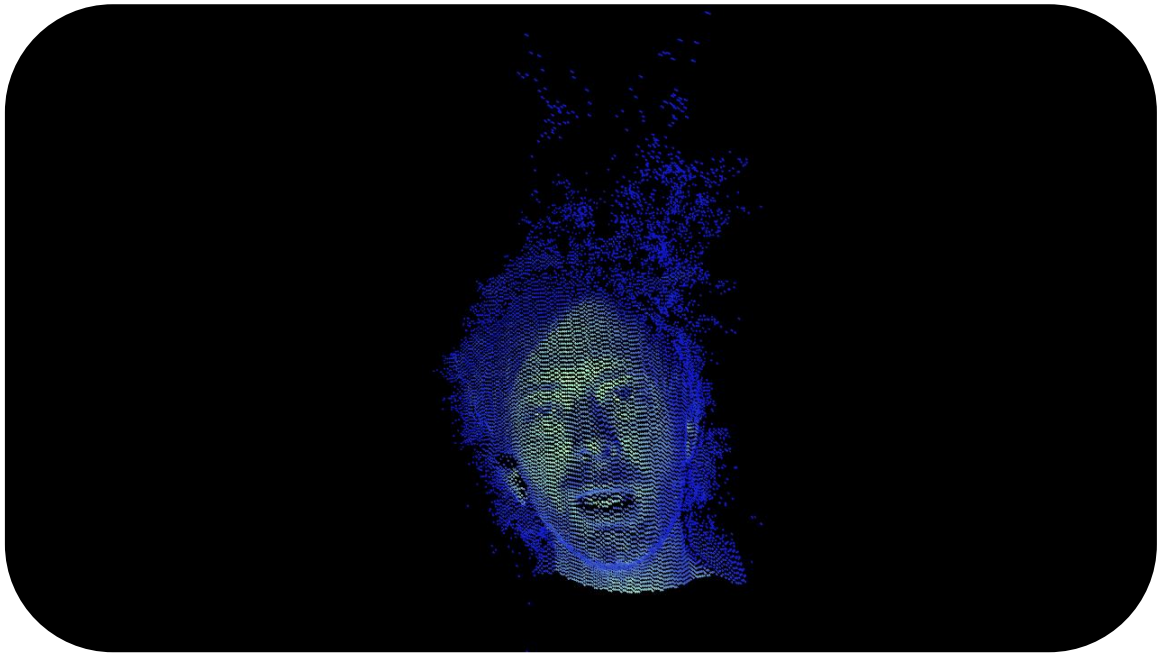
Example: Radiohead – House of Cards (2007)



github.com/dataarts/radiohead

Let’s Try It

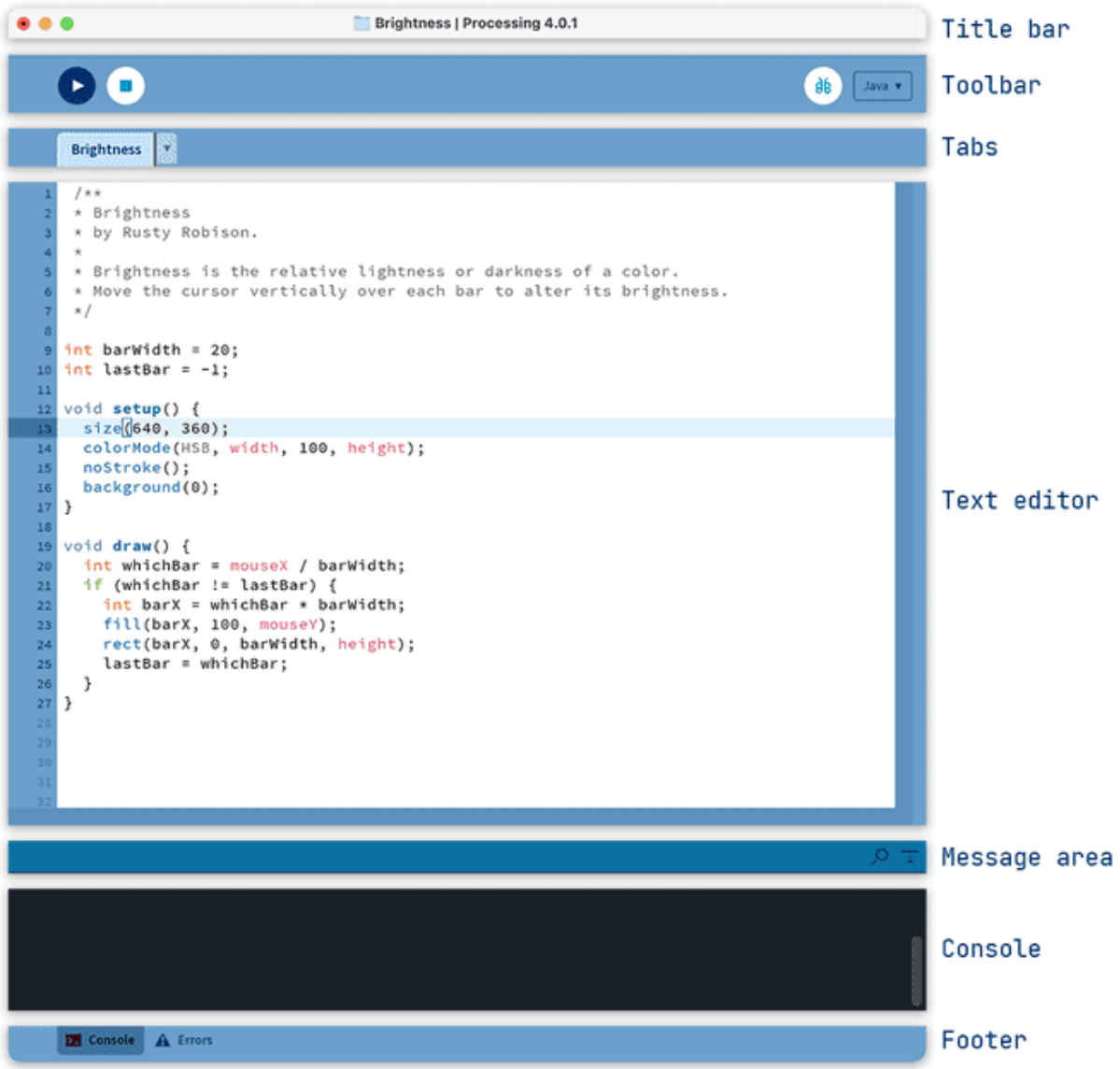
- github.com/dataarts/radiohead



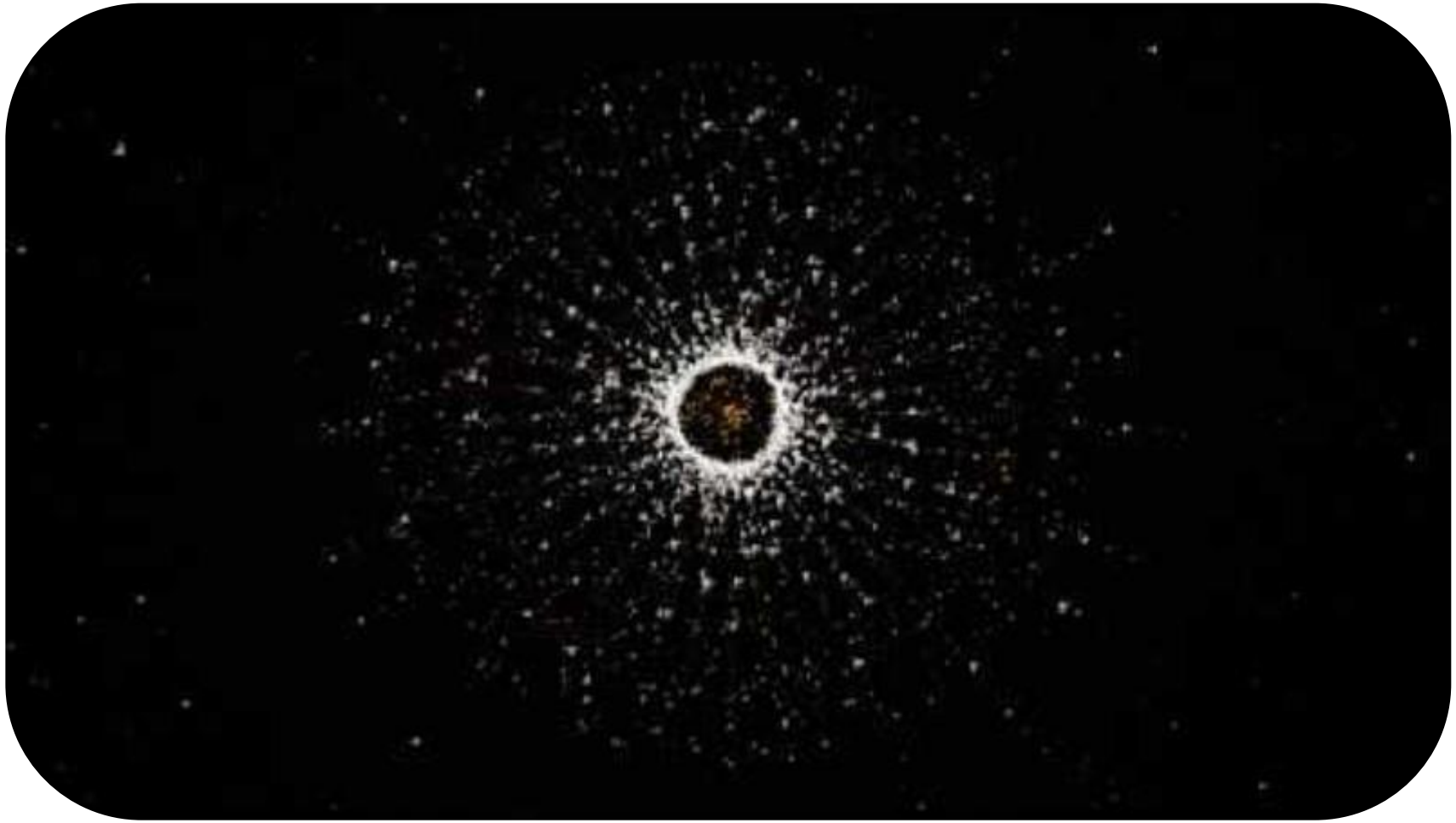
A Processing Sketch

- Processing comes with an IDE
  (Integrated Development Environment)
  - A **text editor**
  - A **console**
  - A **display window** (when you click the *run* button)

Display window



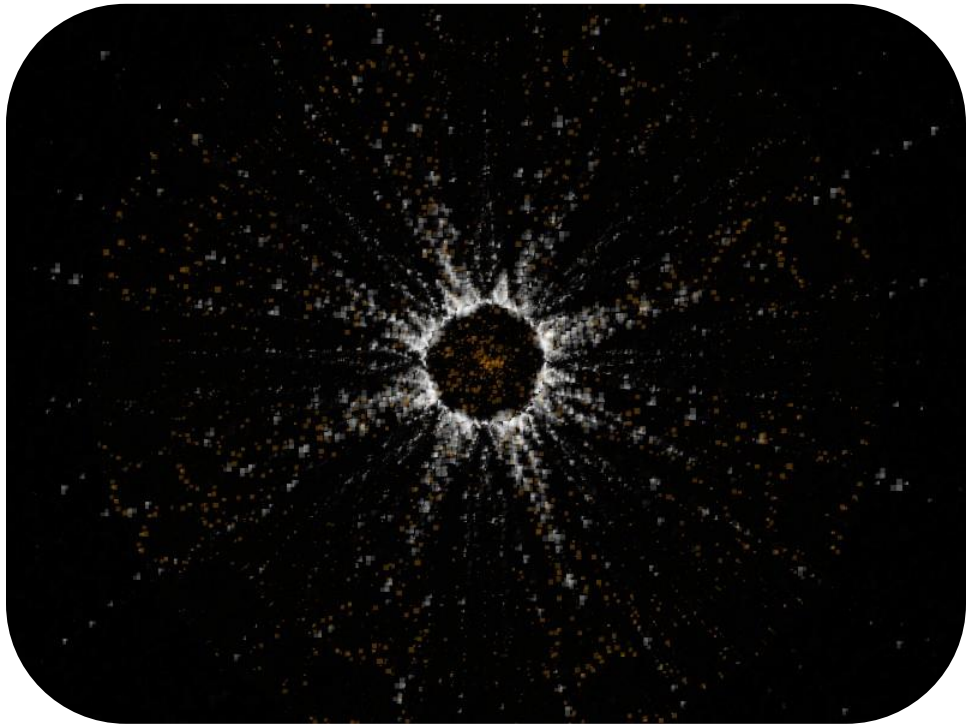
Example: Music Visualizer



youtu.be/283rmgvFDE0 & pastebin.com/JtAn1mV5

Let’s Try It

- pastebin.com/JtAn1mV5



Course Logistics

Course Website

- **Main website**: hermandong.com/teaching/pat204_504_fall2024
  - Syllabus, schedule, lecture slides, code examples, etc.
- **Piazza**: Announcements, Q&A
- **Gradescope**: Assignment submission, grading, regrade requests
- **Canvas**: recordings

Prerequisites

- **None!**
- We'll learn to code in Processing and Max from scratch!

Tentative Schedule

Week	Date	Lecture
1	8/26	Introduction
		Processing Programming
	8/28	├ Programming basics
2	~~9/2~~	├ ~~No Class (Labor Day)~~
	9/4	├ Syntax, variables, data structures
3	9/9	├ Functions, scope & conditionals
	9/11	├ Simple interaction, loops, recursion
4	9/16	├ Arrays & iteration
	9/18	├ Object orientation
5	9/23	├ 3D graphics, camera, lighting
	9/25	├ Scene transformation, motion & physics
6	9/30	├ Review
	10/2	└ Midterm assignment discussions

Processing

Week	Date	Lecture
		MAX Programming
7	10/7	├ Syntax, core objects, control & signal flows
	10/9	├ Oscillators, filters, envelopes
8	~~10/14~~	├ ~~No Class (Fall Study Break)~~
	10/16	├ Subtractive, FM, AM, additive synthesis
9	10/21	├ Handling multiple synth instances, polyphony
	10/23	├ Sample playback & manipulation
10	10/28	├ Granular synthesis
	10/30	├ Audio effects: delay, chorus, flanger, reverb, bit-crusher
11	11/4	├ Networking: MIDI, OpenSoundControl
	11/6	└ Connecting Max to Processing
12	~~11/11~~	~~No Class (Travel)~~
	~~11/13~~	~~No Class (Travel)~~

Max

Advanced Topics

Week	Date	Lecture
		Advanced Topics
13	11/18	├ Generative systems: musical stochastics, autonomy
	11/20	├ Input tracking & interaction
14	11/25	├ Physical modeling: formant & Karplus-Strong synthesis
	~~11/27~~	├ ~~No Class (Thanksgiving)~~
15	12/2	├ Machine learning, feature extraction
	12/4	└ Sound spatialization
16	12/9	**Project presentation**

Homework (5% x 8)

- Programming exercises to get you familiar with Processing & Max
- Due at **11:59pm ET** on the date specified
- Late submissions: **1 point deducted per day**

Midterm Assignment (20%)

- An **open-ended project** using Processing
- Report is due at **11:59pm ET** on **October 6** (*tentative*)
- Late submissions: **Not Accepted**
  - You can always submit your work early and update it later

Project (40%)

- An **open-ended project** using Processing & Max
  - Focused on **creative & artistic use of code**
  - Individual project is expected → You may only work in a group of two for performance purpose
- Proposal is due on **November 17** (*tentative*)
- Final presentation is scheduled on **December 9**
  - Let me know as soon as possible if you can't make it
- Final report is due on **December 15** (*tentative*)
- Late submissions: **Not Accepted**
  - You can always submit your work early and update it later

Grading

Homework	**40%**	**Midterm assignment**	**20%**
├ Homework 1	5%	**Project**	**40%**
├ Homework 2	5%	├ Proposal	5%
├ Homework 3	5%	├ Final report	15%
├ Homework 4	5%	└ Presentation	20%
├ Homework 5	5%		
├ Homework 6	5%		
├ Homework 7	5%		
└ Homework 8	5%		

Resources

- **Processing**
  - Free software with an LGPL license available at https://processing.org/download
  - **Documentation**: https://processing.org/reference/
  - Official **tutorials**: https://processing.org/tutorials
  - Official **examples**: https://processing.org/examples
- **Max**
  - Licensed software with a 30-day trial available at https://cycling74.com/shop
  - **Documentation**: https://docs.cycling74.com/max8/
- Both Processing and Max are available at the **Music Tech Lab**

Optional Reading

- *"Processing: A Programming Handbook for Visual Designers and Artists"* by Casey Raes and Ben Fry (MIT Press) (Amazon)
- *"The Nature of Code"* by Daniel Shiffman (website)
- *"Electronic Music and Sound Design: Theory and Practice with Max/MSP"* by Alessandro Cipriani and Maurizio Giri (Amazon)
- Miller Puckette, 2006. *"The Theory and Technique of Electronic Music"* (book)

Academic Integrity

- Plagiarism and cheating violate SMTD's Academic Code of Conduct. **All plagiarism, cheating and other academic misconduct cases will be reported to SMTD's Office of Academic and Student Affairs.**
- **All assignments must be completed on your own**. You are welcome to exchange ideas with your peers, but this should be in the form of concepts and discussion, not in the form of writing and code.
- You must **provide proper citations/references for any external resources** you use in your writing and code.

Any Questions on the Syllabus?

Let’s Get Started with *Processing*!

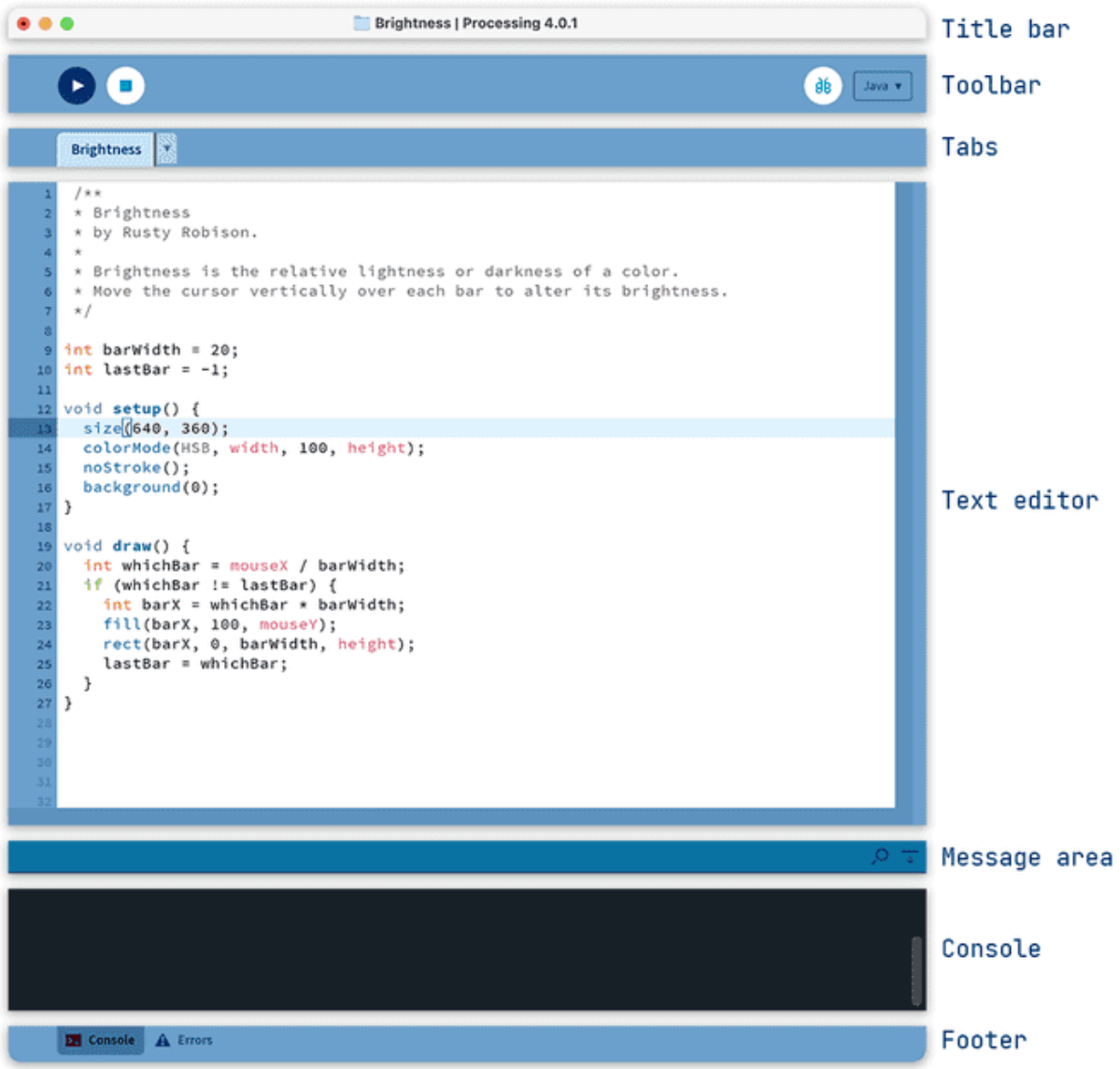
How to Access Processing?

- Processing can be downloaded at https://processing.org/download
- Processing is also available at the Music Teach Lab

A Processing Sketch

- Processing comes with an IDE
  (Integrated Development Environment)
  - A **text editor**
  - A **console**
  - A **display window** (when you click the *run* button)

Display window



The Canvas

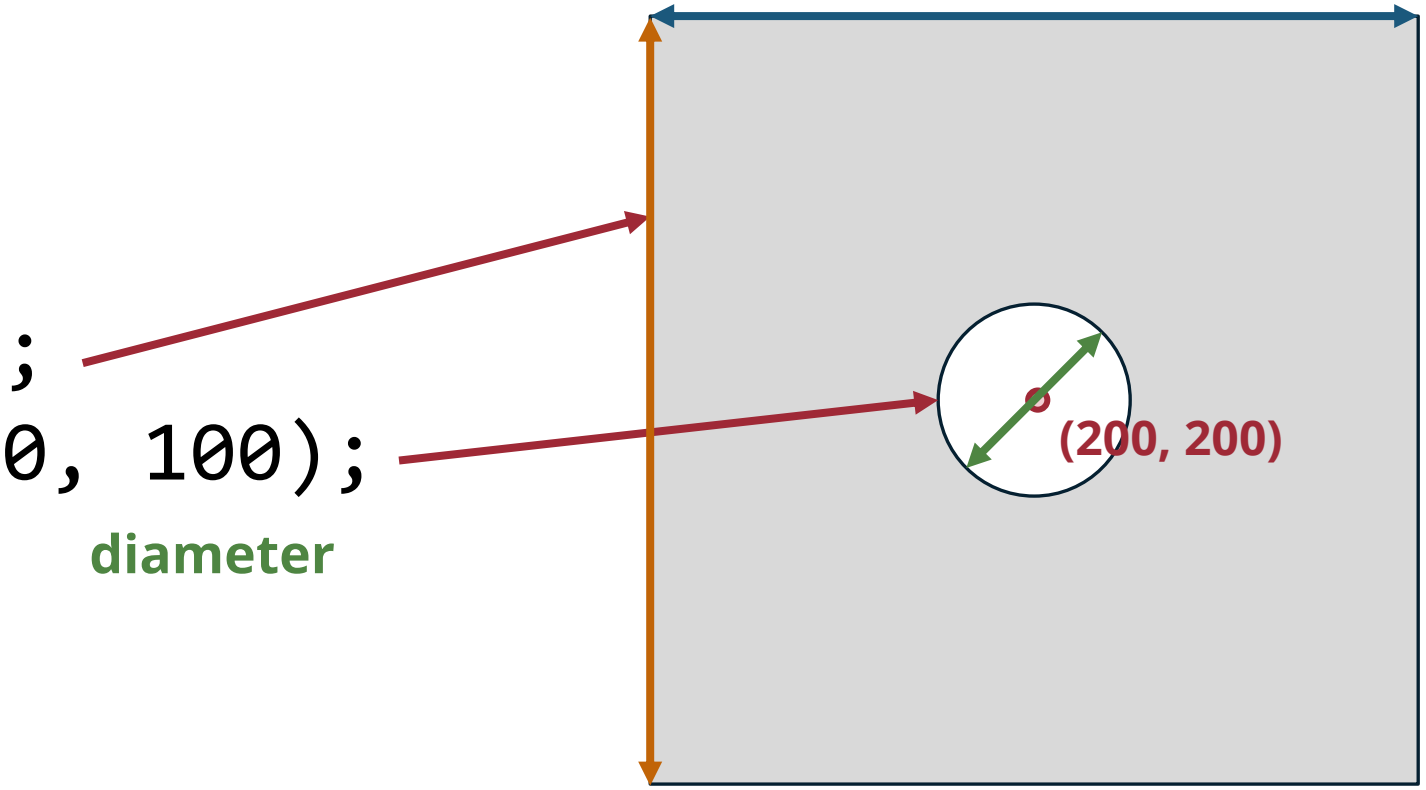
Coordinate System

Your First Processing Sketch

```
size(400, 400);
circle(200, 200, 100);
```

width height

x y diameter



Your First Processing Sketch

Parentheses

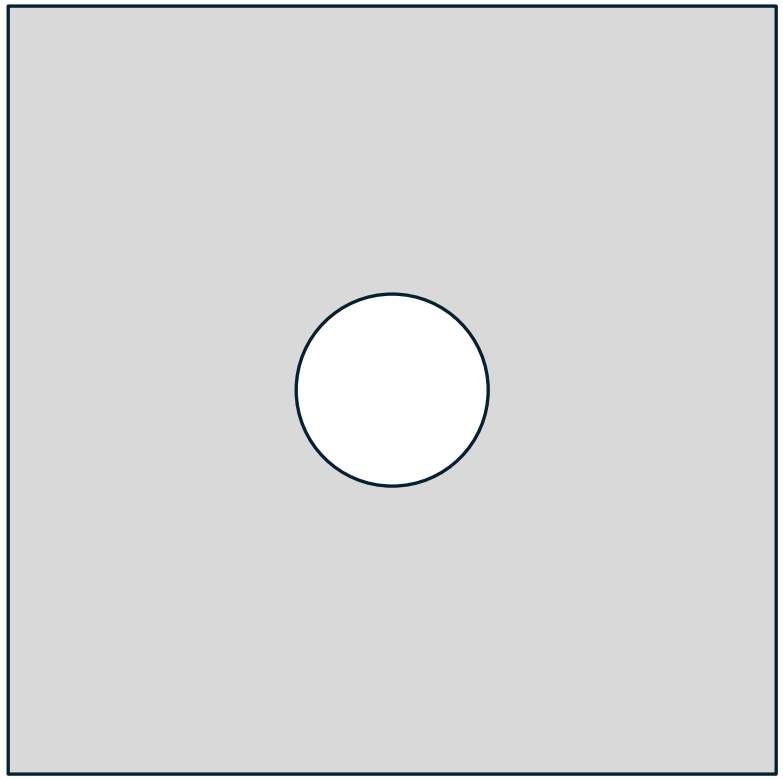
Function name

Arguments (parameters)

```
size(400, 400);
```

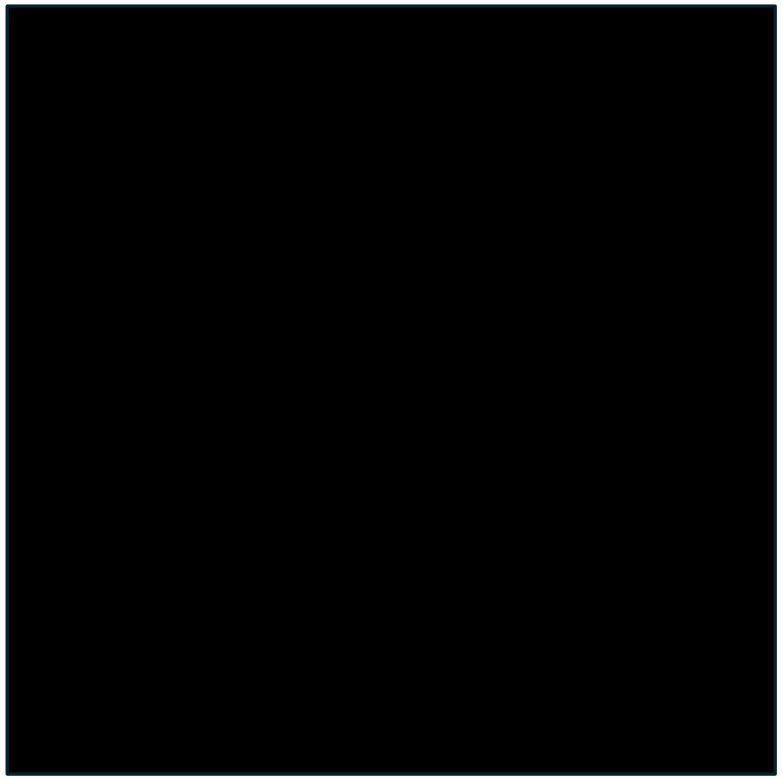
Semicolon!

```
circle(200, 200, 100);
```



Background Color

```
size(400, 400);
background(0);
```

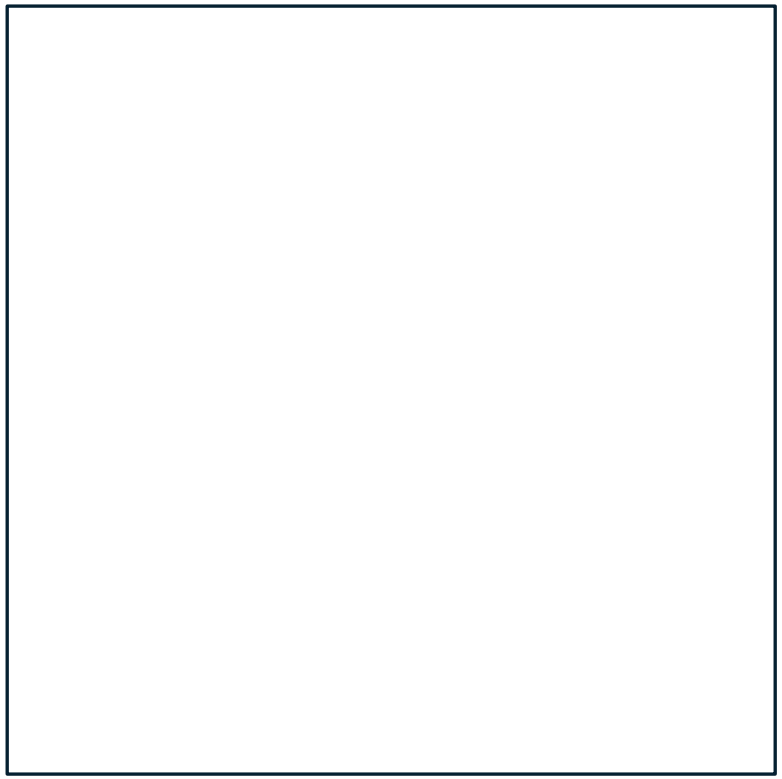


Background Color

```
size(400, 400);
background(255);
```

Range: **0–255**

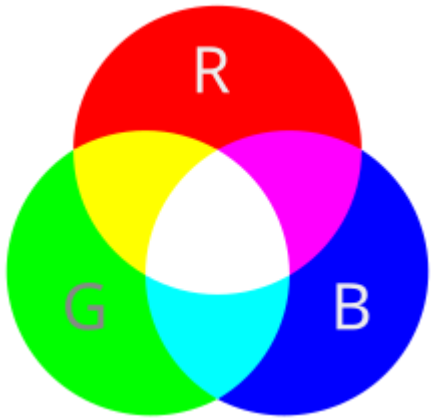
Why?



Background Color

```
size(400, 400);
background(0, 39, 76);
```

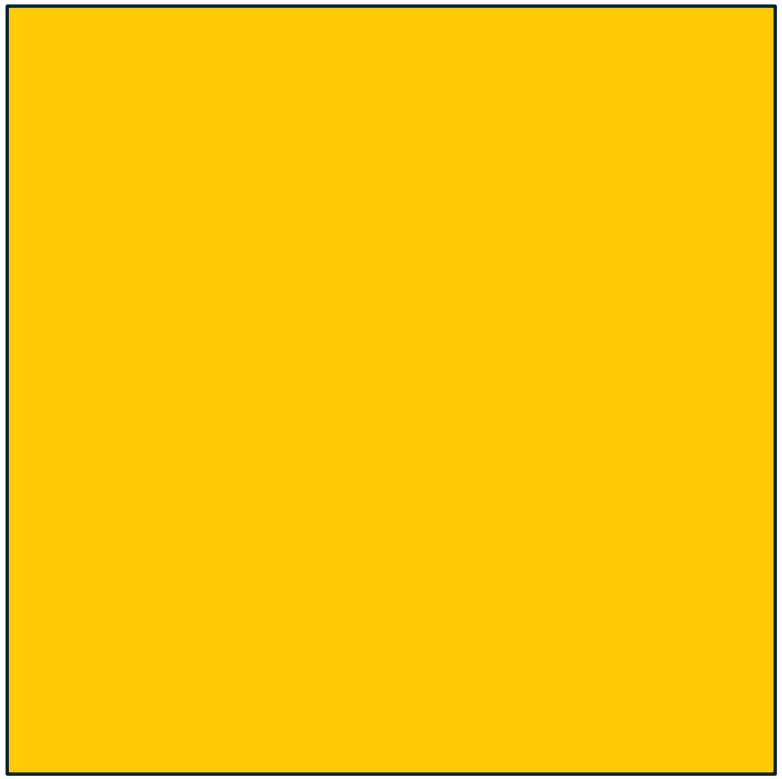
R G B



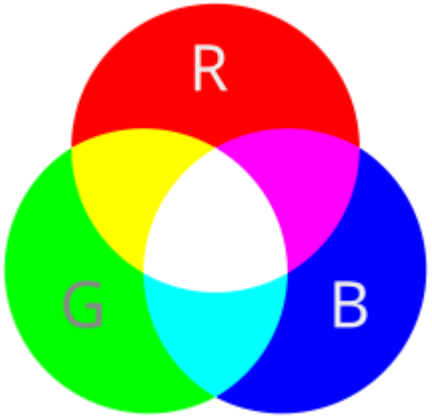
Background Color

```
size(400, 400);
background(#FFCB05);
```

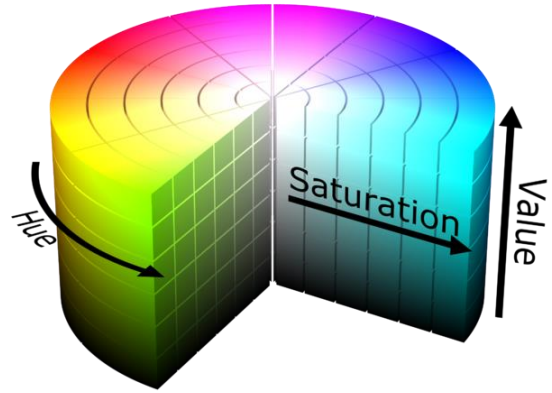
R G B



Colors in Processing



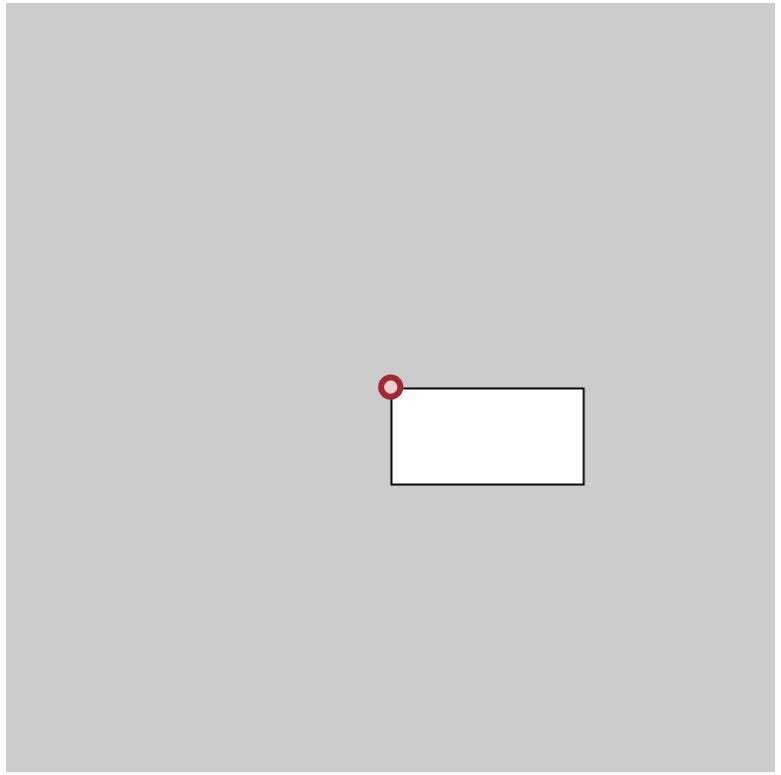
- Grayscale: **`0–255`**
- RGB color: **`(255, 203, 5)`**
- Hex code: **`#00274C`**
- Select through the *Color Selector* tool (via “Tools” in the menu bar)
- Use HSB color mode via **`colorMode(HSB)`**
  - **H**: Hue
  - **S**: Saturation
  - **B**: Brightness



Creating a Rectangle

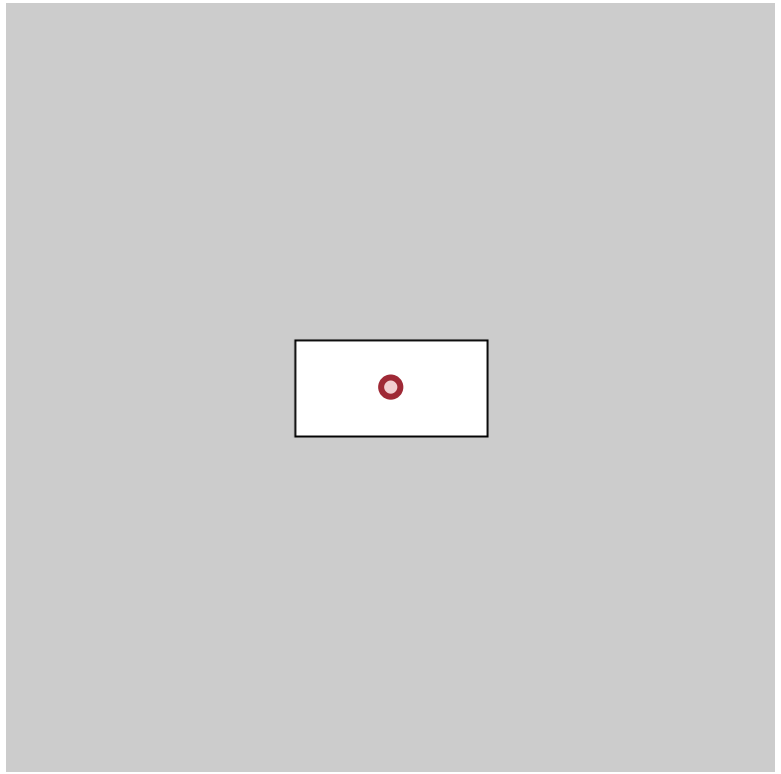
```
size(400, 400);
rect(200, 200, 100, 50);
```

x y width height



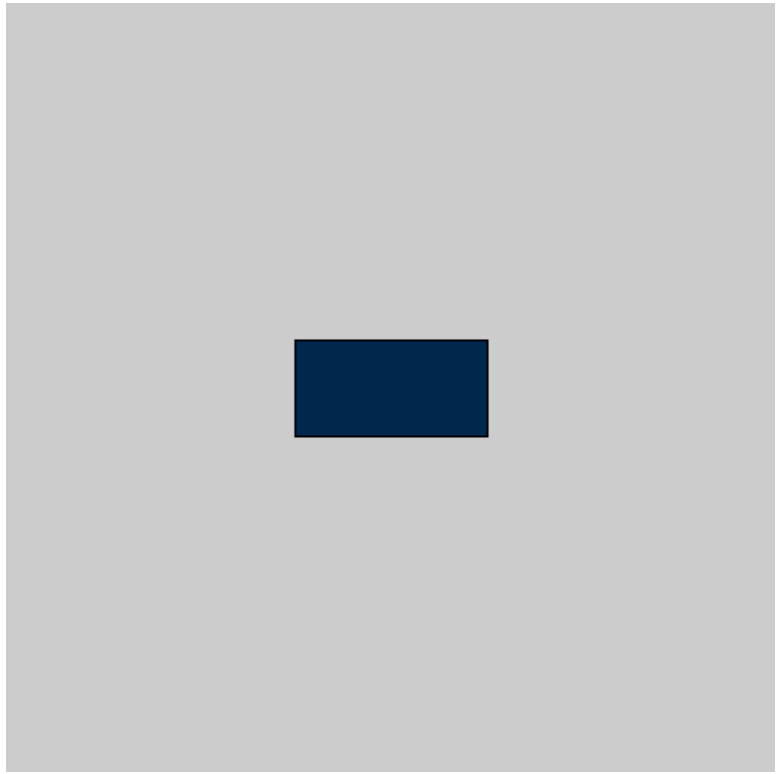
Setting the Anchor Points of Rectangles

```
size(400, 400);
rectMode(CENTER);
rect(200, 200, 100, 50);
```



Changing the Colors of Shapes

```
size(400, 400);
rectMode(CENTER);
fill(#00274C)
rect(200, 200, 100, 50);
```



More Shapes

- Circle `circle(x, y, diameter)`
- Ellipse `ellipse(x, y, width, height)`
- Square `square(x, y, width)`
- Rectangle `rect(x, y, width, height)`
- Point `point(x, y)`
- Line `line(x1, y1, x2, y2)`
- Triangle `triangle(x1, y1, x2, y2, x3, y3)`
- Quadrilateral `quad(x1, y1, x2, y2, x3, y3, x4, y4)`

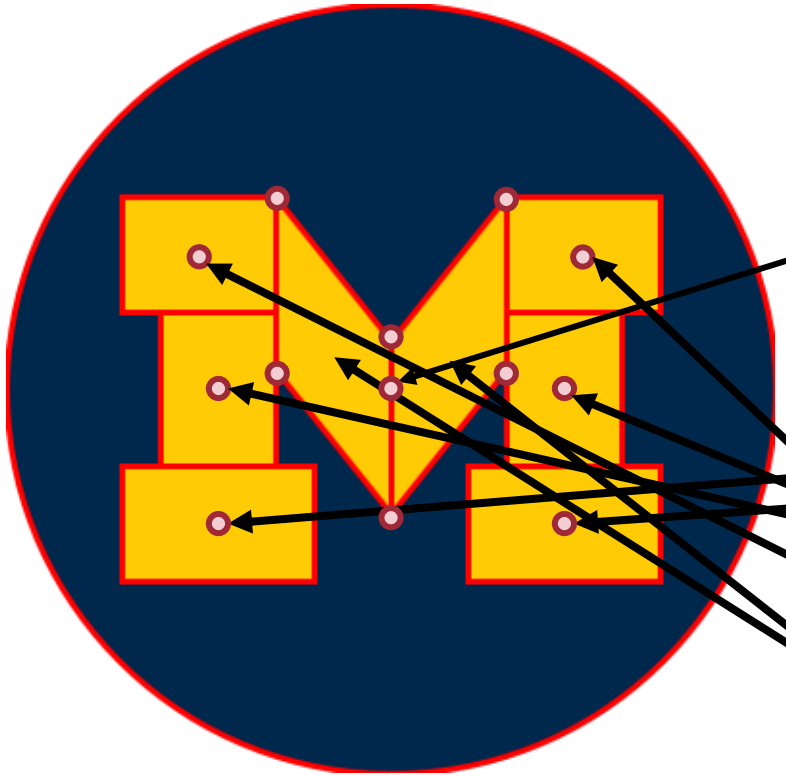
Essential Shortcuts

- **Cmd+R**: Run
- **Esc**: Stop
- **Cmd+Shift+F**: Search in the online reference
- **Cmd+/**: Comment/uncomment
- **Cmd+T**: Auto format

Challenge: Can you recreate the Block M in Processing?

- Michigan **Blue**: **#00274C**
- Michigan **Maize**: **#FFCB05**

My Version



```
void setup() {
  // Create a 400x400 canvas
  size(400, 400);
}

void draw() {
  // Set the background color to white
  background(255);

  // Draw the shapes without outlines
  noStroke();

  // Draw the blue circle at the back
  fill(#00274C);
  circle(200, 200, 400);

  // Set the anchor point of rectangles to the center
  rectMode(CENTER);

  // Set up the yellow text color
  fill(#FFCB05);

  // Draw the feet
  rect(110, 270, 100, 60);
  rect(290, 270, 100, 60);

  // Draw the columns
  rect(110, 210, 60, 150);
  rect(290, 210, 60, 150);

  // Draw the caps
  rect(100, 130, 80, 60);
  rect(300, 130, 80, 60);

  // Draw the "V"
  quad(140, 100, 140, 190, 200, 265, 200, 175);
  quad(260, 100, 260, 190, 200, 265, 200, 175);
}
```