The Art of Code: p5.js & Computer Art
by Marisa Laks, Ian Scheffler, Eric Wilson, Emma Wingreen

General Overview:

The nuts and bolts of computing--loops, variables, arrays--can be scary and downright confusing, especially if they're not placed in a meaningful context.

The goal of this unit is to embed learning some basic programming skills--namely, storing data in variables, using conditional statements, and controlling the flow of a program using loops--in the context of making art using p5.js.

By the end of the unit, students will be able to combine these disparate skills to demonstrate their understanding via a performance task: the creation of a work of art, broadly understood as a static image or a short animation. They will learn how to write code in p5.js. Examples of projects students could create are: https://splatter-paint-demo.marisalaks.repl.co/ , https://finalproject.marisalaks.repl.co/ .

To ensure alignment with the pedagogical goals of the unit, students will be provided with a checklist to ensure that their final product includes all necessary technical elements. (This will also be represented in the final task rubric.)

Another signal feature of the unit is that students will build the final product *as they go*. In other words, they will iteratively add on to their final art as they learn new computing skills.

The unit will also incorporate some elements of art and science history, allowing students to see how particular artists, styles, and movements relate to computing concepts. (Islamic art and the works of certain modern artists, like Kandinsky and Mondrian, seem particularly well suited in this regard.)

Ideally, this unit will help students develop the stance that, as programmers, their goal is not just to write lines of code, but to create something new, original, and meaningful.

In short, by coding in the context of art, the unit aims to inculcate the stance that, at root, coding *is* art.

This unit is designed to be a supplemental unit to a high school or middle school CS class, not necessarily of the AP variety. The idea is that most students will have familiarity with block-based programming, but not all students are necessarily comfortable with text-based programming.

This unit expects that students will have encountered most of the programming concepts before--such as loops, variables, and conditionals--but not in the context of text-based programming. The unit is designed to come closer to the end of the year, once students have encountered the fundamental concepts named above at least once.

In other words, this unit provides a chance to deepen understanding of some fundamental programming concepts, by revisiting them in a new way. (In an AP CSP course, this might take place after the AP Exam.)

Motivation for Unit

For too many students, coding feels fun at first, but can then become a boring slog, through terms and concepts that feel too much like math class, and much less like what they were excited about in the first place.

This unit aims to ground the nuts and bolts of programming--variables, conditionals, flow control--in the tangible and highly engaging context of artistic production.

This will build motivation, and also help students retain information, in much the same way any project-based learning task hopes to do.

This unit also offers the chance for direct co-teaching and collaboration across departments; at schools with an art teacher, artistic concepts from that class can be woven into this unit. (Since p5.js is a very flexible language, students will have a wide latitude in the kind of art they can create.)

Standards Referenced

We are using the NYS K-12 Computer Science and Digital Fluency Standards, in order to become more familiar with them, since these standards
are in the process of being rolled out across the DOE. Standards assessed in this unit include:

- 7-8.CT.7 Design or remix a program that uses a variable to maintain the current value of a key piece of information.
- 7-8.CT.8: Develop or remix a program that effectively combines one or more control structures for creative expression or to solve a problem.
- 9-12.DL.2: Communicate and work collaboratively with others using digital tools to support individual learning and contribute to the learning of others.
- 9-12.DL.4: Independently select advanced digital tools and resources to create, revise, and publish complex digital artifacts or collection of artifacts.
- 9-12.DL.5: Transfer knowledge of technology in order to use new and emerging technologies on multiple platforms.
- 9-12.IC.7: Investigate the use of computer science in multiple fields.
- 7-8.CT.4: Write a program using functions or procedures whose names or other documentation convey their purpose within the larger task.
- 9-12.CT.4: Implement a program using a combination of student-defined and third-party functions to organize the computation.
- 9-12.CT.9: Systematically test and refine programs using a range of test cases, based on anticipating common errors and user behavior.

Tools Used

p5.js Editor: This is a code editor that allows students to use the p5.js library. We plan to use this as the primary code writing platform for students because it is easy to use and allows students to see the results of their code instantaneously.

p5.js reference/documentation page: This is a webpage that allows p5.js users to read all of the built-in functions. We will have students use this as their primary reference point for writing code and exploring new

functions. We plan to use this because it teaches students to use their resources before asking others (especially the teacher) for help.

Adobe Color Picker: This tool allows users to play around with different sliders to get the exact shade of any color they desire. We plan to use this for projects that involve choosing unique colors and to teach students about the RGB values.

Laptops for students: We want each student to have their own designated laptop so they feel like they are a part of the CS class and that the school has invested in their abilities. It is important for each and every student to feel like they are and can be a computer scientist.

SmartBoard: In order to present our slides and do live code-along demos, we will need a SmartBoard or other screen that connects to the teacher laptop.

Google Slides: We plan to use Google Slides to explain the artwork and styles of each artist and to model starter code. Most lessons will feature direct instruction using Google Slides before launching students into their work in p5.js.

Google Form for assessments: We plan to give students summative (and some formative) assessments using Google Forms. This is an easily accessible and gradeable tool that most students have used in the past. Because of how Google Forms autogrades, it is easy as the teacher to see common misconceptions and distractor answers.

Google: Students will search on the Internet for further information about the artists they study in class.

Resources

various images of works of art (incl. in individual lesson plans)

Lessons

Total length: ~2 Weeks (~11 lessons)

Introduction to Computer Art (2 lessons)

Lesson 00:
Standard(s):
- 9-12.DL.2: Communicate and work collaboratively with others using digital tools to support individual learning and contribute to the learning of others.
- 9-12.IC.7: Investigate the use of computer science in multiple fields.
Aim: I can create questions about digital art.
  - ❖ Gallery walk Google slide deck (we each contribute 2 images of digital art?) Video games, animated movies, gallery installations, NFTs, Google Doodles, social media
  - ❖ Question formulation (Scheffler will provide graphic organizer Google doc)

Lesson 01:
Standard(s):
- 9-12.DL.5: Transfer knowledge of technology in order to use new and emerging technologies on multiple platforms.
- 9-12.IC.7: Investigate the use of computer science in multiple fields.

Aim: I can log in to, save and submit work using p5.js
- ❖ Evolution of Computer Art (video? Slide show?) Gallery walk? Group activities? Maybe add to QFT list of questions?
- ❖ Create account on p5.js (Duplicate starter code --> Everything is commented out; then they uncomment to see what it does? --> Save work (does not auto-save)

Mondrian-Inspired Art (2 days)

Lesson 02:
Standard(s):
- 7-8.CT.7: Design or remix a program that uses a variable to maintain the current value of a key piece of information.
- 9-12.IC.7: Investigate the use of computer science in multiple fields.

Aim: I can use p5.js to create art using functions and parameters
- ❖ Learn about artist, seeing teacher demo, doing code along, homework (out of class) assignment
- ❖ Mastery Check (example): Google Form w/questions --> Functions and Parameters

Lesson 03:
Standard(s):
- 7-8.CT.4: Write a program using functions or procedures whose names or other documentation convey their purpose within the larger task.
- 7-8.CT.7: Design or remix a program that uses a variable to maintain the current value of a key piece of information.
- 9-12.IC.7: Investigate the use of computer science in multiple fields.

Aim: I can use p5.js to create art using functions and parameters
- ❖ Independent lab
- ❖ Graphic organizer (can be embedded in code?)

Picasso-Inspired Art (2 days):

Lesson 04:
Standard(s):
- 7-8.CT.8: Develop or remix a program that effectively combines one or more control structures for creative expression or to solve a problem.
- 9-12.IC.7: Investigate the use of computer science in multiple fields.

Aim: I can use p5.js to create art using variables and loops
- ❖ Learn about artist, seeing teacher demo, doing code along
- ❖ Mastery Check: Google Form w/questions --> Variables and Loops

Lesson 05:
Standard(s):
- 7-8.CT.8: Develop or remix a program that effectively combines one or more control structures for creative expression or to solve a problem.

- 9-12.IC.7: Investigate the use of computer science in multiple fields.

Aim: I can use p5.js to create art using variables and loops
- ❖ Independent lab
- ❖ Incl. graphic organizer (can be embedded in code)

Pollock-Inspired Art (2 days)

Lesson 06:
Standard(s):
- 9-12.CT.9: Systematically test and refine programs using a range of test cases, based on anticipating common errors and user behavior.
- 9-12.IC.7: Investigate the use of computer science in multiple fields.

Aim: I can use p5.js to create art using randomization and user input
- ❖ Learn about artist, seeing teacher demo, doing code along

Lesson 07:
Standard(s):
- 9-12.CT.9: Systematically test and refine programs using a range of test cases, based on anticipating common errors and user behavior.
- 9-12.IC.7: Investigate the use of computer science in multiple fields.

Aim: I can use p5.js to create art using randomization and user input
- ❖ Independent lab
- ❖ Mastery Check: Google Form w/questions --> Randomization and User Input

Final Project (3 days):

Lesson 08:
Standard(s):
- 9-12.DL.4: Independently select advanced digital tools and resources to create, revise, and publish complex digital artifacts or collection of artifacts.

Aim: I can propose the design of my final project
- ❖ Brainstorming / Proposals
- ❖ Proposal doc turn in + drawing

Lesson 09:
Standard(s):
- 9-12.CT.4**: Implement a program using a combination of student-defined and third-party functions to organize the computation.

Aim: I can set and accomplish a goal related to my final project
- ❖ Worktime (set individual goals at start of class)

Lesson 10:
Standard(s):
- 9-12.CT.9: Systematically test and refine programs using a range of test cases, based on anticipating common errors and user behavior.

Aim: I can revise my final project based on user input
- ❖ Peer Feedback / Gallery Walk

Assessments

Formative:
+ Mastery Check (x3) (MC/Short Response)
+ Independent Labs (x3)

Summative:
+ Final Project:Create a work of art using p5.js; meet certain requirements (see task sheet)

Main objective Aim: I can create questions about digital art.
Standards 9-12.DL.2: Communicate and work collaboratively with others using digital tools to support individual learning and contribute to the learning of others. 9-12.IC.7: Investigate the use of computer science in multiple fields.

In class exercises
 	
Do Now (5 min):
   - On paper or digitally, students individually respond to the question: When you think of digital art, what comes to mind? Why? Students can respond with images or words!
Activity 1: Virtual Gallery Walk (10 min)
   - Students take 10 minutes to view 10 works in a virtual gallery walk (embedded in today's slide deck). They leave comments noting what they *think, see, or wonder* about each piece.
Activity 2: Think-Write-Pair-Share (7-10 min)
   - After viewing each piece, students first independently define digital art, then share their definitions with a peer, then the class generates a shared definition of digital art, which is memorialized on a slide by the instructor.
Activity 3: QFT (10-15 min)
   - Students generate questions they want to explore about digital art
   - Students type responses in this shared google doc
   - Once they generate questions, students give one another peer feedback
   - If time, class picks top questions they are most interested in.
Exit Ticket (5 min):
   - On a notecard, students jot down 3 things they learned today, 2 things they found interesting, and one question they still have

Notes One thing to gauge when you do this lesson is how familiar different students are with this type of art/art-making–anyone who seems very experienced might be a helpful peer expert when making the project.
Handouts n/a - All work is digital *unless* you'd prefer to print a notecatcher for the gallery walk, rather than have students comment on the slides!

Resources
   - Slide Deck
   - Laptops for students

Assignments   - What p5.js intro videos (linked here and here) to prepare for tomorrow!

Main objective: Aim: I can log in to, save and submit work using p5.js		

Standards:

9-12.DL.5: Transfer knowledge of technology in order to use new and emerging technologies on multiple platforms.
9-12.IC.7: Investigate the use of computer science in multiple fields.

Materials/Resources:

- Slides
- p5.js student accounts
- Google Classroom or other class tool to share projects and links

10 min	**Do Now:** **What is digital art? Give a definition and an example.** See if you can come up with a new example, i.e. not something that we looked at in class yesterday.	Notes: Designed to get students to review material from yesterday

10 min	**Mini-Lesson and/or Code-Along:**
	
	- Vocabulary lesson for upcoming CS terms
	    – A **variable** stores information. Variables change depending on what values you enter.
	    – A **conditional** is an if/then statement. If "something" is true, do "this". If "something" is false, do "that."
	    – A **loop** is code that is repeated until a condition is met.
	    – A **function** is a block of code that performs a specific task.
	        – **Parameters** are the general variables you need for a function.
	        – **Arguments** are the specific values you enter in the function.
	- p5.js walk through:
	    - Create a p5.js account
	    - Duplicating a Sketch
	    - p5.js Tips and Tricks
	    - Sharing Your Work
		Introduction to new CS terms and p5.js tips. Students should refer back to these slides in future lessons

15 min	**Activities**	Students will create their own p5.js

	First p5.js assignment:
	  - Create an account in p5.js
	  - Create a new sketch and title it
	  - Play around with p5.js code (use https://p5js.org/reference/ for ideas)
	  - When you finish, share your work with your class
		accounts and play around with the reference guide

5 min	**Closing:** Share a line of code that you wrote today. What does it do? How did you figure that out?	CFU that students were focused during the assignment

Homework:

If students have a computer and Internet access at home they can continue exploring their p5.js accounts.

Main objective Aim: I can use p5.js to create art using functions and parameters		

Standards

7-8.CT.7: Design or remix a program that uses a variable to maintain the current value of a key piece of information.
9-12.IC.7: Investigate the use of computer science in multiple fields.

Materials/Resources

- Slides
- Code-along teacher template
- Homework (see below)

10 min	**Do Now:** Go to https://p5js.org/examples/ and check out example projects, e.g. Wavemaker. Choose one project and write down what you SEE, THINK, or WONDER about it.	Notes: This is an introduction to different types of projects students can make in p5.js

10 min	**Mini-Lesson:**
	
	    - Introduction to Mondrian slides:
	        - Piet Mondrian
	        - Mondrian Paintings
	        - Where Can We See His Artwork?
	        - Mondrian-Inspired Fashion
		This might be the first time many students learn the name "Mondrian" but they have likely seen art or fashion inspired by him

15 min	**Activities:**
	
	Code-along
	    - Create a new sketch in p5.js
	    - Coordinates in p5.js
	    - Functions in p5.js
	    - Drawing a line in p5.js
	    - Variables in p5.js
	    - Drawing a rectangle in p5.js
	    - Adding in color
	    - Start playing around with the editor in advance of tomorrow's lesson
		These activities are represented in the slides and are meant to be taught as a code-along demo. There is a mix of direct instruction and student exploration.

5 min	**Closing:** Share a line of code that you learned how to write today. What does	Students should be able to explain a line of code

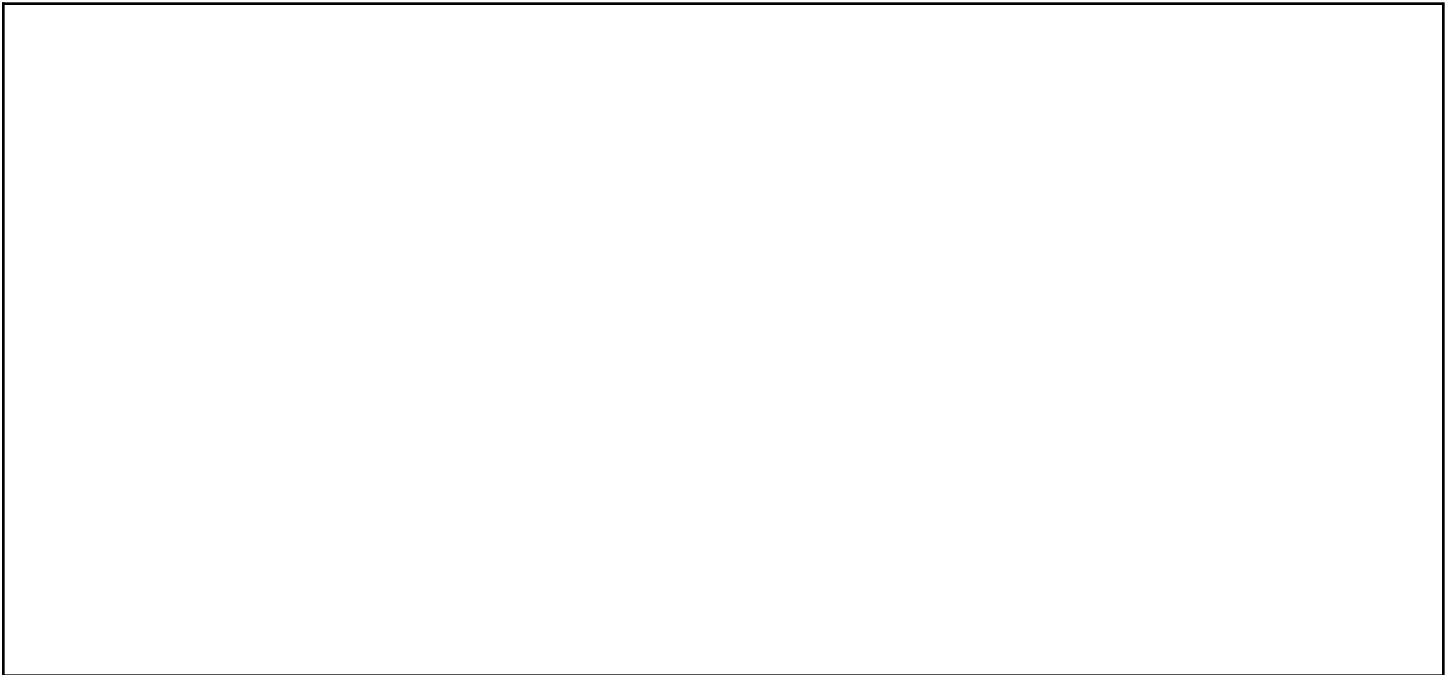|	it do? How does it help you create a Mondrian-inspired p5.js creation?	in their sketch.

Homework (handout below):

In a few sentences, describe Mondrian's artistic style to someone who has never seen his paintings. Create your own Mondrian-inspired artwork. We will code this tomorrow in class!

Mondrian Out-of-class Homework Assignment

1. **In a few sentences, describe Mondrian's artistic style to someone who has never seen his paintings:**

    _____

    _____

    _____

    _____

    _____

2. **Thinking about the paintings we saw today in class, design your own Mondrian-inspired painting below. We will construct a version in class tomorrow using p5.js.**

3. **Which part of your design do you predict will be most challenging to code? Why?**

    _____

    _____

    _____

Main objective Aim: I can use p5.js to create art using functions and parameters		

Standards

7-8.CT.4: Write a program using functions or procedures whose names or other documentation convey their purpose within the larger task.
7-8.CT.7: Design or remix a program that uses a variable to maintain the current value of a key piece of information.
9-12.IC.7: Investigate the use of computer science in multiple fields.

Materials/Resources

- Slides
- Mondrian p5.js lab template
- Mondrian assessment:
  - Student view
  - Teacher view

10 min	**Do Now:** Think back to our previous class. What is an example of a function in p5.js? When is it helpful to use variables?	Notes: Review of CS concepts from the previous class

25 min	**Activities**
	
	Independent lab (20 mins)
	- Students will use the Mondrian template to answer questions and write code to create their own sketches
	
	Question 1
	Take out your homework assignment from last night (a Mondrian-inspired drawing). You are going to make several horizontal and vertical black or gray lines. Try to match your drawing as closely as possible.
	
	// A. Write code that draws a thick vertical black line:
	// B. Now write code for at least 2 more vertical black lines:
	// C. Write code that draws a horizontal black line:
	// D. Now write code for at least 2 more horizontal black lines:
	
	Question 2
	Now you're going to add in several rectangles or squares. Start by making them black or gray. Later, you'll make them primary colors or
		Students will spend most of the class working independently on their sketches. When they are ready they will take the assessment.

	any color of your choosing.
	
	// A. Write code that draws a black or gray rectangle:
	// B. Add color! Experiment with making the rectangles different primary colors (red, yellow, blue). Write code that draws a blue, yellow, or red rectangle:
	// C. What color is your rectangle? How would you change its color?
	// D. Now code in several more squares or rectangles in primary colors (or other colors of your choice):
	
	Mastery Check (5 mins)
	    - Students will be assessed on their knowledge of the line() and rect() functions and variables	

5 min	**Closing:** Share your Mondrian art with the class. What was the most difficult part of coding it?	Students should be able to describe their code and how they made their sketches.

Homework:

Start researching artists for your final project. Whose style of art would you want to recreate using p5.js?

Art of Code: P5.js & Computer Art
Lesson 04: Picasso (Day 1)

Main objective

Aim: I can use p5.js to create art using loops
SWBAT: Use for loops to control different design elements

Standards

7-8.CT.8: Develop or remix a program that effectively combines one or more control structures for creative expression or to solve a problem.
9-12.IC.7: Investigate the use of computer science in multiple fields.

In class exercises
- Direct instruction on Picasso
- Code along to learn for loop techniques to control
- Individual practice:
    - Complete the challenges and answer the questions below.
    - Challenge 1: Use a for loop to draw a column of shapes that goes up and down the page.
    - Challenge 2: Use a for loop to draw a line of shapes that goes diagonally across the page. (HINT: To be diagonal, the x and y values of the shape must both be changing together! Think about math and slope!)
    - What are the three main components of a for loop?
    - Do you have any questions/comments/philosophies/theories/concerns?

Notes
- Slides

Handouts/ Assignments
- Sample target
- For Loops Skeleton

Resources
- Weeping Woman
- Sample face
- Sample target

Assignments
- Mild: Remix and add your own background using at least 2 for loops
- Medium: Remix and add your own background using at least 4 for loops
- Spicy: Code your own image using for loops

Main objective

Aim: I can use p5.js to create Picasso-inspired art using variables and loops

---

Standards

7-8.CT.8: Develop or remix a program that effectively combines one or more control structures for creative expression or to solve a problem.
9-12.IC.7: Investigate the use of computer science in multiple fields.

In class exercises
- Direct instruction on Picasso styles
- Time to work on creation of Picasso-inspired art using for loops

---

Notes
- Collaborative learning opportunities

Handouts
- Slides
- Sample code

Resources
- P5.js editor
- P5 reference

Assignments
- Picasso Project
    - From a blank p5 editor, create a design that contains at least 3 elements of one of Picasso's styles
    - Mild- use at least 2 for loops in your design
    - Medium- use at least 4 for loops to control different design elements
    - Spicy- Use at least 6 for loops to control different design elements

Main objective: Aim: I can use p5.js to create art using randomization and user input		

Standards:

9-12.CT.9: Systematically test and refine programs using a range of test cases, based on anticipating common errors and user behavior.
9-12.IC.7: Investigate the use of computer science in multiple fields.

Materials/Resources:

p5.js editor: https://editor.p5js.org/
https://www.jackson-pollock.org/
Google Slides

7 min	**Do Now:** Click on the link. Play around with the site. Describe what happens when you drag and click your mouse. Call on Students for answers. (Possible answers: As you move the mouse, there is a paint trail. When you click on the mouse, the color changes. If you stay in one place, the splotch is bigger)	Notes: https://jacksonpollock.org/
8 min 10 min	**Mini-Lesson:** Jackson Pollock (1912 - 1956) was a painter in the 1940s and 1950s. His artwork known as "drip paintings" were in the "Abstract Expression" style. Show examples of some of Pollock's work. Show two minutes of video. Today we are going to learn how to make random splatter paint designs similar to Pollock's work. Click on the link to see an example of the finished project. What do you notice about the design? Click on the link and save a copy of the file for your starter code. Look at the code. How can we change the following: ● the definition for the variable name to your name? ● the color of the circle? ● the position of the circle? ● the size of the circle? Make the changes and run the code. What happens when you click on the canvas?	Youtube link: https://youtu.be/KnUvEcE7kPA Link to the finished splatter paint canvas: https://splatter-paint-demo.marisalaks.repl.co/ Link for the starter code: https://editor.p5js.org/mlaks23/sketches/TVGUvjFn8

15 min	**Code Along:** Follow along with the teacher's instructions. Add to your code as the teacher adds to the code. Right now, our circle is in one specific spot. If we want to put a circle in different places on the canvas we can use a built-in function called mouseX and mouseY. Change the parameters in the ellipse to mouseX and mouseY. Run your code. Click on the canvas and see what happens. Now we are going to comment out the background in the draw function. What happens now? Why do we now see a trail? When we remove the background from the draw loop, we stop redrawing the background in each frame. We will move the ellipse function call from the draw function and place it in the mousePressed function so that we place a circle where we click the mouse. How can we draw more circles each time we click the mouse? Write code for another ellipse and add to the coordinates. Play around with different locations and sizes. Try to make the two ellipses different colors. What if we want our circles to be different sizes? We can use another built in function to make our circles random sizes. The function random() takes one or two parameters. Where else can we use the function random? We can also use random for the colors and we can give different parameters if we want to use specific color ranges. Remember to save your code.	code: ``` ellipse(mouseX, mouseY, 10) ``` Splatter Paint Code Example: https://editor.p5js.org/mlaks23/full/fYQJPdQ-l
5 min	**Closing:** Whiparound: Choose one to answer: What was easy? What was difficult? What more would you like to add?	

Homework:

Jackson Pollock (1912 - 1956) was a painter in the 1940s and 1950s. His artwork known as "drip paintings" were in the "Abstract Expression" style.

1. Go to the website: https://www.jackson-pollock.org/

2. Read the biography and look at the paintings on the bottom of the page under the title "Masterpieces of Jackson Pollock."

3. Answer the following questions:
a. What thoughts do you have about Jackson Pollock's paintings?
b. Which painting do you like the most?
c. Why do you think his work is so popular?
d. Do you think he had a plan or did he paint at random?

Further Information: https://www.moma.org/artists/4675

Art of Code: P5.js & Computer Art
Lesson 07: Pollock Inspired Art (Day 2)

Main objective Aim: I can use p5.js to create art using randomization and user input		
Standards 9-12.CT.9: Systematically test and refine programs using a range of test cases, based on anticipating common errors and user behavior. 9-12.IC.7: Investigate the use of computer science in multiple fields.		
Materials/Resources p5.js editor: https://editor.p5js.org/ Google Slides		

8 min	**Do Now:**
	
	 Given the two lines of code below:
	 1) What does the function random() do in the fill function?
	 2) What does the function random() do in the ellipse function?
	
	 ```fill(random(255),random(255), 0)```
	 ```ellipse(mouseX + random(10, 20), mouseY + random(10, 20), random(10))```
		Notes:

5 min	**Mini-Lesson:** Review of random() function: Takes either 0, 1 or 2 arguments. If no argument is given, returns a random number from 0 up to (but not including) 1. If one argument is given and it is a number, returns a random number from 0 up to (but not including) the number. If two arguments are given, returns a random number from the first argument up to (but not including) the second argument.	p5.js Reference: https://p5js.org/reference/#/p5/random
15 min 5 min	**Splatter Paint Challenges:** Open your splatter paint code from yesterday. Choose some or all of the following to do to your code: ● Write a new welcome message. Get user input to greet the user with their name. ● Add another line of text after the first that gives some instructions - like "click your mouse to splatter some paint." ● Add different shapes to your splatters. ● Change the color palette of your splatters. Share your screen with your partner.	Link for the starter code: https://editor.p5js.org/mlaks23/sketches/TVGUvjFn8 Splatter Paint Code Example: https://editor.p5js.org/mlaks23/sketches/fYQJPdQ-l

| 12 min | **Closing: Pollock Assessment:**<br><br>Click on the link. Answer the questions on the form. Use the code below for questions 2 to 5.<br><br>```
21 ▼ function mousePressed() {
22
23     fill(0, 255, 0)
24     ellipse(20, 40, 30)
25
26     fill(random(255), random(255), random(255))
27     ellipse(mouseX, mouseY, random(20))
28
29  }
```<br><br>1) Write a line of code that asks a user to input their name.<br>2) What does the function mousePressed() do?<br>3) Describe the difference between the code in line 23 and the code in line 26.<br>4) Describe the difference between the code in line 24 and the code in line 27.<br>5) Change the code to make a blue circle of size 20.<br><br>Remember to save your code and submit the link on the google form. | |

Homework:

Continue working on your code.

Main objective

Aim: I can propose the design of my final project

Standards

9-12.DL.4: Independently select advanced digital tools and resources to create, revise, and publish complex digital artifacts or collection of artifacts.

In class exercises

Do Now (5 minutes)
- If you could design ANYTHING for your final project–using the skills you've learned this unit, what would you create?
- Students will respond to this in their notes

Virtual Gallery Walk (10 minutes)
- Students will peruse a selection of projects created using p5.js for inspiration
- They will save screenshots or links to their proposal graphic organizer

Proposal Drafting (10 minutes)
- Students will draft proposals using the graphic organizer

Peer Feedback (10 minutes)
- Students will give each other feedback in pairs

Revisions (10 minutes)
- Students will revise their proposals, based on the feedback

Exit Ticket (5 minutes)
- Students will respond on a notecard: On a scale of 1 to 5, how excited are you about the project you proposed today? Why?

Notes

Make sure to have blank paper so students can also draw if they want!

Handouts

- Final project proposal graphic organizer (digital - individual Google Doc)

Resources

- Proposal graphic organizer
- Slide Deck

Assignments

- Once approved by instructor, start coding your project!

Art of Code: P5.js & Computer Art
Lesson #09: Final Project (Day 2)

Main objective Aim: I can set and accomplish a goal related to my final project
Standards 9-12.CT.4: Implement a program using a combination of student-defined and third-party functions to organize the computation.
In class exercises Do Now (5 min): - Students will set a goal for today, based on their proposal. The goal should be specific (i.e., "I will sketch out my program, then aim to code X part of it.", not "I will work on the program") Worktime (40 minutes) - Students will have the entire period to work on coding their program - Instructor will circulate and conference with students who need help Exit Ticket (5 min): - Students will reflect on whether or not they achieved their goal and why.
Notes You may need to give feedback on what a specific goal looks like–or even a mini-lesson! Take a look at the goals as you circulate and see how specific they are.
Handouts - Students will continue to work off the proposal from yesterday.
Resources - Proposal graphic organizer - Laptops (individual)
Assignments - Finish drafting program if incomplete!

Art of Code: P5.js & Computer Art
Lesson #10: Final Project (Day 3)

Main objective Aim: I can revise my final project based on user input
Standards 9-12.CT.9: Systematically test and refine programs using a range of test cases, based on anticipating common errors and user behavior.

In class exercises
 	
Do Now (5 minutes)
   -   What are you happy about with your final project? What could be going better? Why?
Coding Time (10 min)
   -   Students should revisit the code they wrote yesterday, and prepare to give and receive feedback
Peer Feedback Gallery Walk (10 min)
   -   Students should visit 2-3 peers, run their programs, and leave feedback on their individual feedback graphic organizers
Revise Code (20 minutes)
   -   Students should implement the feedback they received
Exit Ticket (5 minutes)
   -   On a notecard: What are you most proud of accomplishing this unit? Why?

Notes Make sure to emphasize that good feedback is *constructive*–might be helpful to ad-lib an example of bad feedback out loud and have students point out how it can be improved.

Handouts
 	
   -   [Peer Feedback Template (Individual Google Doc)](#)
   -   Final Project Rubric

Resources
   -   Peer Feedback Template
   -   Laptops
   -   Smartboard
   -   [Final Project rubric](#)

Assignments   -   Turn in revised code tonight!