Lesson 1.2: Drawing and Shapes

https://codehs.com/course/17244/lesson/1.2

Description	In this lesson, students learn how to create a canvas inside of the `setup()` function and learn its coordinate system. Students will also learn how to draw basic shapes in the `draw()` function loop.
Objective	Students will be able to: • Understand the HTML5 canvas and coordinate system • Differentiate the function of `setup()` and `draw()` • Set the size of the canvas: `createCanvas()` • Use p5 shape-drawing functions for primitive shapes (`ellipse`, `rect`, `line`) • Call and pass parameters to functions
Activities	1.2.1 Video: Drawing and Shapes 1.2.2 Exercise: Create a Canvas 1.2.3 Connection: Canvas Coordinates 1.2.4 Exercise: Line 1.2.5 Exercise: Rectangle 1.2.6 Exercise: Ellipse
Prior Knowledge	• Understanding of the Cartesian Coordinate system. Students should be able to locate points in an (x,y) plane • Understanding of the terms, radius, length, width, and height

Planning Notes	• Depending on the needs of your students, consider reviewing the following concepts:
	  ◦ radius and diameter in a circle (the ellipse() function can be used to draw circles on the screen)
	  ◦ width and height in a rectangle
	  ◦ plotting points in an (x, y) coordinate plane
	• If students are using a graph paper notebook, have them first practice drawing the canvas and shapes in their notebook first before writing the code.

Standards Addressed	
Teaching and Learning Strategies	**Lesson Opener:** • Have students brainstorm and write down answers to the discussion questions listed below. Students can work individually or in groups/pairs. Have them share their responses. Alternatively, you can start the class by reviewing radius, diameter, and x, y coordinate planes. [5 mins] **Activities:** • Watch the lesson video that will introduce students to the canvas and how to draw shapes. [7-10 mins] ◦ Consider pausing the video and using a whiteboard to draw shapes at certain coordinates. Ask students to draw shapes in their notebook or on the whiteboard based on the coordinates, dimensions, and colors you give them.

- Before students begin the exercises, have them pair with a partner to do pair programming. Since students are learning these concepts for the first time, it can be useful for students to work through them together.
- Have students complete the *Create a Canvas* activity. [5 mins]
  - Have students set the background color of their canvas to their favorite color listed in the exercise.
  - Encourage students to create canvases of different colors and sizes by changing the values they pass into the `createCanvas()` and `background()` functions.
- Have students complete the *Canvas Coordinates* activity. [5 mins]
  - Have students explore the interactive grid and take note of how the numbers increase and decrease as they move their cursor in different directions.
  - Prompt students to think about the following:
    - In which direction would a shape move if you increase/decrease the x coordinate?
    - In which direction would a shape move if you increase/decrease the y coordinate?
- Have students complete the *Line* activity. [10 mins]
  - Students may struggle with placing each line in the correct position since each line has two x coordinates and two y coordinates. Encourage them to map out their solution on paper before getting started.
- Have students complete the *Rectangle* activity. [10 mins]
  - Have students remove the fourth parameter of any `rect()` function call. Ask them to reflect on what happens. They should note that it turns the rectangle into a square. This happens because the `rect()` function will give the rectangle the same height value as the width value provided.
- Have students complete the *Ellipse* activity. [10 mins]
  - Have students recreate the same shape by using the `ellipse()` function. They should note that the `ellipse()` function works just like the `circle()` function if it is only passed three parameters.

Lesson Closer:

- Have students reflect and discuss their responses to the end of class discussion questions. [5 mins]
- Provide any handouts to students as an exit ticket or for homework. [5 - 10 min]

Discussion Questions

Beginning of Class:

- There are many apps that help us create visual art. What are some apps or programs that you use or know of that can be used to create art?
  - *Answer will vary. Students might say MS Paint, Procreate, Photoshop, Snapchat, etc.*
- Math can sometimes be used to create art. Can you think of an example of how math can be used by artists in their work?
  - *Answers will vary. Students might refer to symmetry in shapes and objects or they may refer to vector art and how mathematical equations help prevent loss of quality in images.*

End of Class:

- What is the difference between the `setup()` and `draw()` functions?
  - *The `setup()` function runs once and should be used to load any static elements like the canvas. The `draw()` function runs continuously and should be used to draw dynamic elements like shapes.*
- What are the coordinates for the top left corner of the canvas?
  - *(0, 0)*
- At least how many parameters does the `rect()` function need?
  - *3. x position, y position, width (this will create a square rectangle).*

Resources/Handouts

Graph Paper

	Sketching Shapes

Vocabulary

Term	Definition
setup()	Called once when the program starts and is used to define the initial environment properties.
draw()	Continuously executes the lines of code contained inside its block until the program is stopped.
createCanvas(width, height)	Creates a canvas element in the document and sets the dimensions of it in pixels.
ellipse(x, y, w, [h])	Draws an ellipse to the screen given the x, y coordinate and the width and height (height is optional).
rect(x, y, w, [h])	Draws a rectangle to the screen given the x, y coordinate and the width and height (height is optional).
line(x1, y1, x2, y2)	Draws a line (direct path between two points) to the screen. The first two parameters x1, y1 control the position of the first point and the last two parameters x2, y2 control the position of the second point.

Modification: Advanced	Modification: Special Education	Modification: English Language Learners
• Have students create something interesting using the commands covered in the lesson in a p5.js Sandbox. • Encourage students to practice with different shapes found here	• Have students write all of the new vocabulary in a notebook. They should include all of the commands learned, including where and how to use them. • Consider accepting pseudocode as the finished product for some students or completing several of the challenges instead of all of them for full credit.	• Have students write all of the new vocabulary in a notebook. They should include all of the commands learned, including where and how to use them. • Consider accepting pseudocode as the finished product for some students or completing several of the challenges instead of all of them for full credit.