

QUIZ

What is NOT true about p5.js?

p5.js is a modern programming language for the web.



Correct! p5.js is not a language in itself. It uses HTML, CSS, and JavaScript as underlying technologies.

p5.js functions can dynamically alter the DOM of a web page.

p5.js can be used to manipulate images and videos.

p5.js is a JavaScript library that makes visual programming easier.

Complete the code to make a 10 by 5 grid of circles that are 25 pixels apart horizontally and 50 pixels vertically.

```
for(let i = 0; i < 10; i++){  
  for(let j = 0; j < 5; j++){  
    circle( ,  , 10);  
  }  
}
```



You got it!

Which of the following statements is NOT true about the `<canvas>` element?

`width` and `height` are built-in variables that return the width and height of the `<canvas>` element.

The `<canvas>` element is white by default.



Correct! The default color of the `<canvas>` element is transparent.

The x coordinate of the `<canvas>` element is the distance from the left edge of the `<canvas>`, and the y coordinate is the distance from the top edge of the `<canvas>` element.

The origin, (0, 0), of the `<canvas>` element is the top-left corner of the canvas.

What happens if the `createCanvas()` function is NOT called?

If there is no existing `<canvas>` element, p5.js will draw directly to the DOM of the web page.

There will be no `<canvas>` element on your web page.

p5.js will automatically create a `<canvas>` element in the DOM.



Correct! p5.js will by default create a 100 pixels by 100 pixels `<canvas>` element unless the `noCanvas()` function is called.

Your p5.js sketch will draw to an existing `<canvas>` element.

How do you incorporate the p5.js library into your project?

The p5.js library is included using the `<link>` tag in the `<head>` section of the HTML document.

The p5.js library is included as a `<script>` tag in the `<head>` section of the HTML document.



Correct!

Your project will automatically identify the p5.js library if it exists in your project folder.

The p5.js library must be embedded as an `<iframe>` element.

What is the correct order of events when a p5.js sketch is loaded into an HTML page?

1. The HTML page is loaded.
2. The p5.js library is loaded.
3. The `setup()` function is called by the user.
4. The `draw()` function is called repeatedly.

1. The p5.js library is loaded.
2. The HTML page is loaded.
3. The `setup()` function is called when the HTML page finishes loading.
4. The `draw()` function is executed repeatedly.

1. The HTML page is loaded.
2. The p5.js library is loaded.
3. The p5.js sketch is loaded.
4. The `setup()` function is automatically called.
5. The `draw()` function is called repeatedly in a loop.



Correct!

Fill in the below `setup()` function to create a sketch that has a canvas of 800 pixels width and 600 pixels height with a gray background.

```
function setup(){  
  createCanvas ( 800 , 600 );  
  background ( 127 );  
}
```



You got it!

Which is NOT a valid color value as an argument for the `background()` function?

```
background('#ff0000', 20);
```



Correct! You cannot provide an opacity value for a hexadecimal color value as a second argument of the `background()` function.

```
background('red');
```

```
background(20, 55, 30, 200);
```

```
background(255);
```

Which of the following is NOT a p5.js shape function?

The `line()` function creates a line between two points. The function requires four arguments: the x and y positions for each endpoint.

The `rectangle()` function draws a rectangle with four required arguments: x and y positions of the top left corner of the rectangle, width, and height.



Correct! The function to draw a rectangle is written `rect()` not `rectangle()`.

The `square()` function draws a square with three required arguments: x and y positions of the top left corner of the square, and width.

The `ellipse()` function will draw an ellipse using four required arguments: x and y positions of the center of the ellipse, width, and height.

Considering the code below, which shape will appear on top?

```
function draw(){  
  line(width / 2, 0, width / 2, height);  
  rect(50, 150, 300, 100);  
  circle(200, 200, 150);  
}
```

The circle will appear on top.



Correct! The shape drawn last will be rendered on top.

There is no way to tell which shape is on top because they have no fill colors.

Consider the following sketch:

```
function setup(){  
  createCanvas(400,400);  
  strokeWeight(5);  
  fill('green');  
}  
function draw(){  
  stroke(0);  
  triangle(200, 100, 100,200, 300, 200);  
  
  stroke('red');  
  ellipse(width/2, height/2, 100);  
}
```

What are the fill and stroke colors of the triangle?

The triangle will have a green fill color and a red stroke color.

The triangle will have no fill color and black stroke color.

The triangle will have a white fill color and a black stroke color.

The triangle will have a green fill color and a black stroke color.



Correct! The green fill color is set in the `setup()` function and the black stroke color is set in the line above the `triangle()` function.

Complete the following code to draw the following shapes:

- A red point with a stroke weight of 5 pixels.
- A transparent triangle with a red stroke color.
- A green quadrilateral with no stroke.

```
function draw(){  
  ✓ strokeWeight (5);  
  ✓ stroke ('red');  
  ✓ point (50, 50);  
  
  ✓ noFill() ;  
  ✓ triangle (50, 100, 200, 200, 400, 100);  
  
  fill('green');  
  ✓ noStroke() ;  
  ✓ quad (50, 50, 75, 25, 125, 25, 100, 50)  
}
```



You got it!