QUIZ

Which of the following code examples properly changes the location the ellipse is drawn from, so the x and y coordinates passed into the `ellipse()` function signifies the top-left corner position of the ellipse's bounding box?

```
ellipseMode();
ellipse(50, 50, 100, 100);
```

```
ellipse(50, 50, 100, 100);
```

```
ellipse(50, 50, 100, 100);
ellipseMode(CORNER);
```

```
ellipseMode(CORNER);
ellipse(50, 50, 100, 100);
```

👏 Correct! The `ellipseMode(CORNER)` called before the `ellipse()` function changes the origin point from the center to the top-left corner of the ellipse's bounding box.

Which of the following statements is NOT true about the `push()` and `pop()` functions?

The `push()` function saves the current drawing styles and transformations.

The `push()` function can be used without the `pop()` function, and vice versa.

You got it! This is not true because the `push()` and `pop()` functions should always be used together.

The `pop()` function restores styles and transformations back to the settings that were in effect prior to the most recent call to `push()`.

The `push()` and `pop()` functions are used to isolate style changes (i.e. stroke, fill, etc) and transformations.

Fill in the arguments for the `makeCircle()` function calls so that it creates two circles at different positions but the same size of 50 pixels.

```
function draw(){
  makeCircle(10, 20, 50 ); // First circle

  makeCircle(10, 30 , 50 ); // Second circle
}

function makeCircle(xPos, yPos, size) {
  ellipse(xPos, yPos, size, size);
  xPos = xPos + 1;
}
```

You got it!

Which of the following code does NOT rotate the canvas by π / 4 radians (45 degrees)?

```
rotate(radians(45));
```

```
rotate(45);
```

Correct! This would not rotate the canvas by π / 4 radians or 45 degrees. In order to rotate by 45 degrees, you would need to specify `angleMode(DEGREES)` above the `rotate()` function, so p5.js knows to conv...

Show more

Which of the following code blocks will place the rectangle in the middle of the canvas and cause it to rotate around its center?

```
rectMode(CENTER);
rect(0, 0, 400, 400);
translate(width/2, height/2);
rotate(radians(frameCount));
```

```
rectMode(CENTER);
rotate(radians(frameCount));
translate(width/2, height/2);
rect(0, 0, 400, 400);
```

```
rectMode(CENTER);
translate(width/2, height/2);
rotate(radians(frameCount));
rect(0, 0, 400, 400);
```

You got it! When you apply multiple transformations, the order makes a difference. The combination of the `rectMode(CENTER)` and `translate()` functions places the retangle in the middle of the canvas first. ...
Show more

Which of the following statements is NOT true about the `noise()` function?

The `noise()` function returns a random float value between 0 and the value passed into the function.

You got it! This statement is incorrect. The `noise()` function returns a random float value between 0 and 1.

The `noise()` function returns a random float value, between 0 and 1.

The `noise()` function is useful for creating more natural random movements.

The `noise()` function returns a random value based on the Perlin noise.

Fill in the code below to increment the x position of the ellipse by 1 and decrement the y position by 5.

```
let xPos;
let yPos;

function draw(){
  xPos = xPos + 1;
  yPos = yPos - 5;

  ellipse(xPos, yPos, 20, 20);
}
```

You got it!

Which of the following code snippets properly changes the location the rectangle is drawn from so that the x and y coordinate passed into the `rect()` function signifies the center of the rectangle?

```
rectMode(CORNER);
rect(width/2, height/2, 50, 50);
```

```
rect(width/2, height/2, 50, 50);
```

```
rect(width/2, height/2, 50, 50);
rectMode(CENTER);
```

```
rectMode(CENTER);
rect(width/2, height/2, 50, 50);
```

Correct! The `rectMode(CENTER)` called before the `rect()` function changes the origin point from the top-left corner to the center of the rectangle.

What is the `frameCount` variable used for?

The `frameCount` variable returns the maximum number of frames that can be run in a sketch.

The `frameCount` variable counts the number of frames rendered in one draw loop.

The `frameCount` variable tells you the number of frames rendered per second.

The `frameCount` variable stores the number of frames run since the program started.

You got it!

Fill in the code below to generate a random value between 0 and 255 for the `grayValue` variable and a random number between 0 and `width / 2` for the `size` variable.

```
let grayValue = random( 0 , 256 );
let size = random( width / 2 );
```

You got it!

Fill in the code below to scale the size of the coordinate system so that the rectangle appears 50% smaller and the ellipse looks 200% wider and 400% taller.

```
scale( 0.5 );
rect(0, 0, 120, 120);

scale( 2 , 4 );
ellipse(100, 100, 120, 120);
```

You got it!

After defining your own function, where in the code for a p5.js sketch can you call the function?

User-defined functions can only be called inside the `setup()` function.

User-defined functions can be called inside the `draw()` and `setup()` function.

Correct!

User-defined functions can only be called inside that same function.

User-defined functions can only be called in other user-defined functions.

What is the default value of the `frameRate()` function that a p5.js sketch runs at?

```
frameRate(24);
```

```
frameRate(60);
```

This is correct! By default, a p5.js sketch runs at `frameRate(60)` or 60 FPS. This means that the `draw()` function runs 60 times per second.

There is no default frame rate. You set it yourself.

```
frameRate(120);
```

Fill in the code below to horizontally shear the rectangle by 25 degrees and vertically shear it by 90 degrees. Specify the angle mode to use degrees.

```
angleMode( DEGREES );
shearX( 25 );
shearY( 90 );
rect(50, 50, 100, 100);
```

You got it!

Which of the following statements is true about the `draw()` function?

The `draw()` function is automatically called after the `setup()` function. The `draw()` function is executed repeatedly in a loop.

Correct!

The `draw()` function is automatically called before the `setup()` function. Both functions repeatedly run in a loop.

The `draw()` function is automatically called before the `setup()` function. The `draw()` function runs repeatedly in a loop.

The `draw()` function is automatically called after the `setup()` function. The `setup()` function runs repeatedly in a loop.

What does the `translate()` function in the code below do?

```
ellipse(0, 0, 20, 20);
translate(40, 40);
rect(0, 0, 50, 50);
```

It adds 40 pixels to the width and height of the shapes above and below the `translate()` function.

It moves the origin of the p5.js canvas from the coordinates (0, 0) to (40, 40).

Correct! The `translate()` function moves the whole p5.js canvas to the coordinates specified as arguments of the function.