**1. What does RGBA stand for?**

RGBA stands for Red, Green, Blue, and Alpha. It is a color model that is widely used in digital imaging and computer graphics to represent and manipulate colors.

Red (R): Determines the amount of red in the color, ranging from 0 to 255.

Green (G): Determines the amount of green in the color, ranging from 0 to 255.

Blue (B): Determines the amount of blue in the color, ranging from 0 to 255.

Alpha (A): Represents the transparency or opacity of the color. It ranges from 0 (completely transparent) to 255 (completely opaque).

**2. From the Pillow module, how do you get the RGBA value of any images?**

1. Install Pillow
2. Import the necessary module
3. Open the image using the Image.open() method
4. Convert the image to RGBA mode
5. Get the RGBA values of a specific pixel by using the getpixel() method, passing the coordinates of the pixel as arguments.

from PIL import Image

image = Image.open("path\_to\_image.png")

image = image.convert("RGBA")

x = 100

y = 200

rgba = image.getpixel((x, y))

print(rgba)

**3. What is a box tuple, and how does it work?**

A box tuple refers to a tuple that represents a rectangular region or bounding box within an image. It is commonly used to specify areas of interest or regions for various image operations, such as cropping, resizing, or extracting specific portions of an image.

A box tuple consists of four values in the following order: (left, upper, right, lower).

left: The x-coordinate of the leftmost edge of the box.

upper: The y-coordinate of the uppermost edge of the box.

right: The x-coordinate of the rightmost edge of the box.

lower: The y-coordinate of the lowermost edge of the box.

from PIL import Image

image = Image.open("path\_to\_image.png")

box = (100, 100, 300, 300)

cropped\_image = image.crop(box)

**4. Use your image and load in notebook then, How can you find out the width and height of an Image object?**

1. To find out the width and height of an Image object using the Pillow module in Python, we can use the size attribute of the image.
2. Install Pillow module
3. Import the necessary modules and open the image
4. Get the width and height of the image using the size attribute
5. The size attribute returns a tuple containing the width and height of the image, which we can unpack into separate variables width and height.

from PIL import Image

image = Image.open("path\_to\_image.png")

width, height = image.size

print("Width:", width)

print("Height:", height)

**5. What method would you call to get Image object for a 100×100 image, excluding the lower-left quarter of it?**

To get an Image object for a 100x100 image, excluding the lower-left quarter we can use the crop() method of the Pillow module in Python.

from PIL import Image

image = Image.open("path\_to\_image.png")

width, height = image.size

box = (0, height // 2, width // 2, height)

cropped\_image = image.crop(box)

**6. After making changes to an Image object, how could you save it as an image file?**

To save changes made to an Image object as an image file using the Pillow module in Python we can use the save() method.

Specify the desired file path and name in the save() method, using the appropriate file extension for the desired image format.

**7. What module contains Pillow’s shape-drawing code?**

The module that contains Pillow's shape-drawing code is called ImageDraw. It is a part of the Pillow library and provides functionality for drawing various shapes, lines, and text on an Image object.

**8. Image objects do not have drawing methods. What kind of object does? How do you get this kind of object?**

The ImageDraw module in Pillow provides an ImageDraw object that allows you to perform various drawing operations, such as drawing shapes, lines, and text, on an Image object.

To get an ImageDraw object we need to use the ImageDraw.Draw() function and pass it the Image object we want to draw on.