1. What does RGBA stand for?

Ans. RGBA stands for Red, Green, Blue, and Alpha. It is a color model that represents colors using a combination of red, green, and blue channels, along with an alpha channel that represents the transparency or opacity of the color.

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

Ans. In the Pillow module (PIL), we can get the RGBA value of any image using the getpixel() method. Here's an example of how to do it:

from PIL import Image

# Open an image file

image = Image.open('example.jpg')

# Get the RGBA value at a specific pixel location (x, y)

pixel\_rgba = image.getpixel((x, y))

# The variable pixel\_rgba will now contain a tuple with four values: (R, G, B, A)

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

In Pillow, a box tuple is a tuple representing a rectangular region in an image. It is usually represented as a 4-tuple containing four coordinates: (left, upper, right, lower). The box tuple defines the region of interest within the image, where:

left: The x-coordinate of the leftmost pixel.

upper: The y-coordinate of the uppermost pixel.

right: The x-coordinate of the rightmost pixel (exclusive, not included in the box).

lower: The y-coordinate of the lowermost pixel (exclusive, not included in the box).

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

Ans. To find out the width and height of an Image object in Pillow, we can use the size attribute. Here's an example:

# Open an image file

image = Image.open('example.jpg')

# Get the width and height of the image

width, height = image.size #image.size returns tuple of (x,y) we have assigned value to different variables

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

Ans. To get an Image object for a 100×100 image, excluding the lower-left quarter of it, we can use the crop() method. Here's an example:

# Open an image file

image = Image.open('example.jpg')

# Crop the image to exclude the lower-left quarter (width//2, height//2, width, height)

cropped\_image = image.crop((image.width // 2, image.height // 2, image.width, image.height))

# The variable 'cropped\_image' now contains the cropped Image object.

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

Ans. To Save the modified image to a new file

image.save('modified\_example.jpg')

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

Ans. Pillow's shape-drawing code is contained in the ImageDraw module. To use the shape-drawing functionalities, we need to import the ImageDraw module separately. Here's an example of how to draw shapes on an image:

from PIL import Image, ImageDraw

# Open an image file

image = Image.open('example.jpg')

# Create an ImageDraw object

draw = ImageDraw.Draw(image)

# Draw a red rectangle on the image

draw.rectangle([x1, y1, x2, y2], outline='red')

# Save the modified image to a new file

image.save('modified\_example.jpg')

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

Ans. Image objects in Pillow do not have drawing methods. To draw on an image, we need to create an ImageDraw object, as in the previous example. The ImageDraw object provides methods to draw shapes, text, and other graphics directly on the image.