1. What does RGBA stand for?

Ans-:**RGBA**(Red-Green-Blue-Alpha) The format of the **RGBA** Value **RGBA** color values are an extension of RGB color values with an **alpha** channel - which specifies the opacity for a color. An **RGBA** color value is specified with: **rgba**(red, green, blue, **alpha**). The **alpha** parameter is a number between 0.0 (fully transparent) and 1.0 (fully opaque).

1. From the Pillow module, how do you get the RGBA value of 'CornflowerBlue'?

Ans-:In Pillow, RGBA values are represented by a tuple of four integer values. Pillow offers the ImageColor.getcolor() function.This function takes a color name string as its first argument, and the string 'RGBA' as its second argument, and it returns an RGBA tuple.

CMYK Coloring-:subtractive color model

RGB Coloring-:additive color model

❶ >>> **from PIL import ImageColor**

❷ >>> **ImageColor.getcolor('red', 'RGBA')**

(255, 0, 0, 255)

❸ >>> **ImageColor.getcolor('RED', 'RGBA')**

(255, 0, 0, 255)

>>> **ImageColor.getcolor('Black', 'RGBA')**

(0, 0, 0, 255)

>>> **ImageColor.getcolor('chocolate', 'RGBA')**

(210, 105, 30, 255)

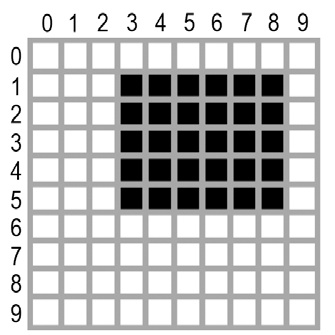
>>> **ImageColor.getcolor('CornflowerBlue', 'RGBA')**

(100, 149, 237, 255)

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

Ans-: Many of Pillow’s functions and methods take a box tuple argument. This means Pillow is expecting a tuple of four integer coordinates that represent a rectangular region in an image. The four integers are, in order, as follows:

* **Left:** The x-coordinate of the leftmost edge of the box.
* **Top:** The y-coordinate of the top edge of the box.
* **Right:** The x-coordinate of one pixel to the right of the rightmost edge of the box. This integer must be greater than the left integer.
* **Bottom:** The y-coordinate of one pixel lower than the bottom edge of the box. This integer must be greater than the top integer.



* Figure 17-2. The area represented by the box tuple (3, 1, 9, 6)

1. For an image file called zophie.png, what feature returns an Image object?

Ans-: >>> **from PIL import Image**

>>> **catIm = Image.open('zophie.png')**

  
The Image.open() function returns a value of the Image object data type, which is how Pillow represents an image as a Python value. You can load an Image object from an image file (of any format) by passing the Image.open() function a string of the filename.

Zophie image stored in a variable named catIm. Be sure that the zophie.png file is in the current working directory so that the Image.open() function can find it. Otherwise, you will also have to specify the full absolute path in the string argument to Image.open().

1. What is the way to determine the width and height of a Picture object's image?

Ans-: >>> **from PIL import Image**

>>> **catIm = Image.open('zophie.png')**

>>> **catIm.size**

❶ (816, 1088)

❷ >>> **width, height = catIm.size**

❸ >>> **width**

816

❹ >>> **height**

1088

>>> **catIm.filename**

'zophie.png'

>>> **catIm.format**

'PNG'

>>> **catIm.format\_description**

'Portable network graphics'

❺ >>> **catIm.save('zophie.jpg')**

After making an Image object from Zophie.png and storing the Image object in catIm, we can see that the object’s size attribute contains a tuple of the image’s width and height in pixels ❶. We can assign the values in the tuple to width and height variables ❷ in order to access with width ❸ and height ❹ individually. The filename attribute describes the original file’s name. The format and format\_description attributes are strings that describe the image format of the original file (with format\_description being a bit more verbose).

1. How can you get an Image object for a 100x100 image that is missing the lower-left quarter?

Ans-: >>> **from PIL import Image**

>>> **croppedIm =** imageObj**.crop((0,50,50,50))**

>>> **croppedIm.save('cropped.png')**

Notice that you are passing a box tuple to crop(), not four separate integer arguments.

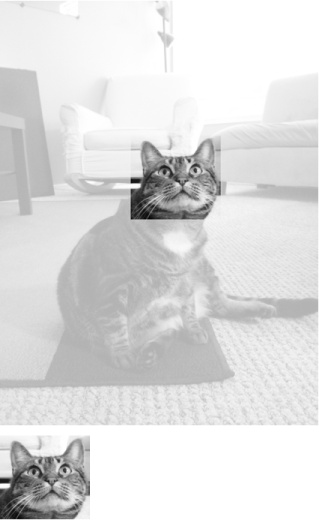
1. How can you save an Image object as an image file after making adjustments to it?

Ans-: Call the imageObj.save('new\_filename.png') method of the Image object.

Ex-:

>>> **croppedIm = catIm.crop((335, 345, 565, 560))**

>>> **croppedIm.save('cropped.png')**



1. In which module does Pillow's shape-drawing code reside?

Ans:- . The ImageDraw module contains code to draw on images.

Ex: >>> **from PIL import Image, ImageDraw, ImageFont**

>>> **import os**

❶ >>> **im = Image.new('RGBA', (200, 200), 'white')**

❷ >>> **draw = ImageDraw.Draw(im)**

❸ >>> **draw.text((20, 150), 'Hello', fill='purple')**

>>> **fontsFolder = 'FONT\_FOLDER' # e.g. 'Library/Fonts'**

❹ >>> **arialFont = ImageFont.truetype(os.path.join(fontsFolder, 'arial.ttf'), 32)**

❺ >>> **draw.text((100, 150), 'Howdy', fill='gray', font=arialFont)**

>>> **im.save('text.png')**

1. Drawing methods are not available for image objects. What kind of thing does that? What's the best way to get this kind of thing?

Ans:- ImageDraw objects have shape-drawing methods such as point(), line(), or rectangle(). They are returned by passing the Image object to the ImageDraw.Draw() function.