1. Creating random colors

The code below creates a 50×50 image using the `SimpleImage` library. It assigns a **random color** to every pixel. Fill in the blanks to complete the program. A single color value is in the range of 0 - 255 inclusive.

> **Reminder:**
> The `random` module can generate random numbers for us.
>
> `random.randrange(start, end)`
>
> returns a random integer from `start` (included) to `end` (not included).

```
from simpleimage import SimpleImage
import random

image = ____________________________

for x in range(_________________):
    for y in range(_________________):

        red = ______________________
        green = ____________________
        blue = ______________________

        image.set_rgb(x, y, red, green, blue)

image.show()
```

Solution:

```
from simpleimage import SimpleImage
import random

image = SimpleImage.blank(50, 50)

for x in range(image.width):
    for y in range(image.height):

        red = random.randrange(0, 256)
        green = random.randrange(0, 256)
        blue = random.randrange(0, 256)

        image.set_rgb(x, y, red, green, blue)

image.show()
```

2. Drawing with pixels

The code below creates a 3×3 white image. It then colors some pixels black. Draw the final image by shading the pixels that become black.

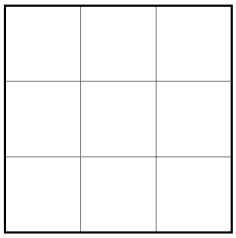
```
image = SimpleImage.blank(3, 3)

for i in range(3):
    image.set_rgb(i, i, 0, 0, 0)

image.set_rgb(0, 2, 0, 0, 0)
image.set_rgb(2, 0, 0, 0, 0)

image.show()
```

Draw the output image:



Solution:

The black pixels are

$$(0,0),\ (1,1),\ (2,2),\ (0,2),\ (2,0)$$

forming an X.

3. Inverting an image

Write a function that inverts an image and returns the inverted image.

Each pixel's color is represented as (red, green, blue), where each value is between 0 and 255. To invert a color, replace each RGB value with 255 minus its current value.

For example: `(0, 100, 255)` becomes `(255, 155, 0)`.

```
def invert(image):
```

Solution:

```
def invert(image):
    for x in range(image.width):
        for y in range(image.height):
            pixel = image.get_rgb(x, y)

            red = 255 - pixel[0]
            green = 255 - pixel[1]
            blue = 255 - pixel[2]

            image.set_rgb(x, y, red, green, blue)

    return image
```