

In [3]:

```
import cv2
import matplotlib.pyplot as plt

# Function to display images using matplotlib
def show_image(img, title="Image"):
    plt.figure(figsize=(10, 8))
    if len(img.shape) == 2: # grayscale image
        plt.imshow(img, cmap='gray')
    else: # color image
        img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
        plt.imshow(img)
    plt.title(title)
    plt.axis('off')
    plt.show()

# Read an image
image = cv2.imread('cham.jpg')

# Display the original image
show_image(image, "Original Image")

# Convert to grayscale
gray_image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
show_image(gray_image, "Gray Image")

# Save the grayscale image
cv2.imwrite('gray_image.jpg', gray_image)

# Blur the image
blurred_image = cv2.GaussianBlur(image, (5, 5), 0)
show_image(blurred_image, "Blurred Image")

# Edge detection using Canny
edges = cv2.Canny(image, 100, 200)
show_image(edges, "Edges")

# Resize the image
resized_image = cv2.resize(image, (300, 200))
show_image(resized_image, "Resized Image")

# Rotate the image
(h, w) = image.shape[:2]
center = (w / 2, h / 2)
M = cv2.getRotationMatrix2D(center, 45, 1.0)
rotated_image = cv2.warpAffine(image, M, (w, h))
show_image(rotated_image, "Rotated Image")

# Draw a Line
cv2.line(image, (0, 0), (511, 511), (255, 0, 0), 5)
show_image(image, "Image with Line")

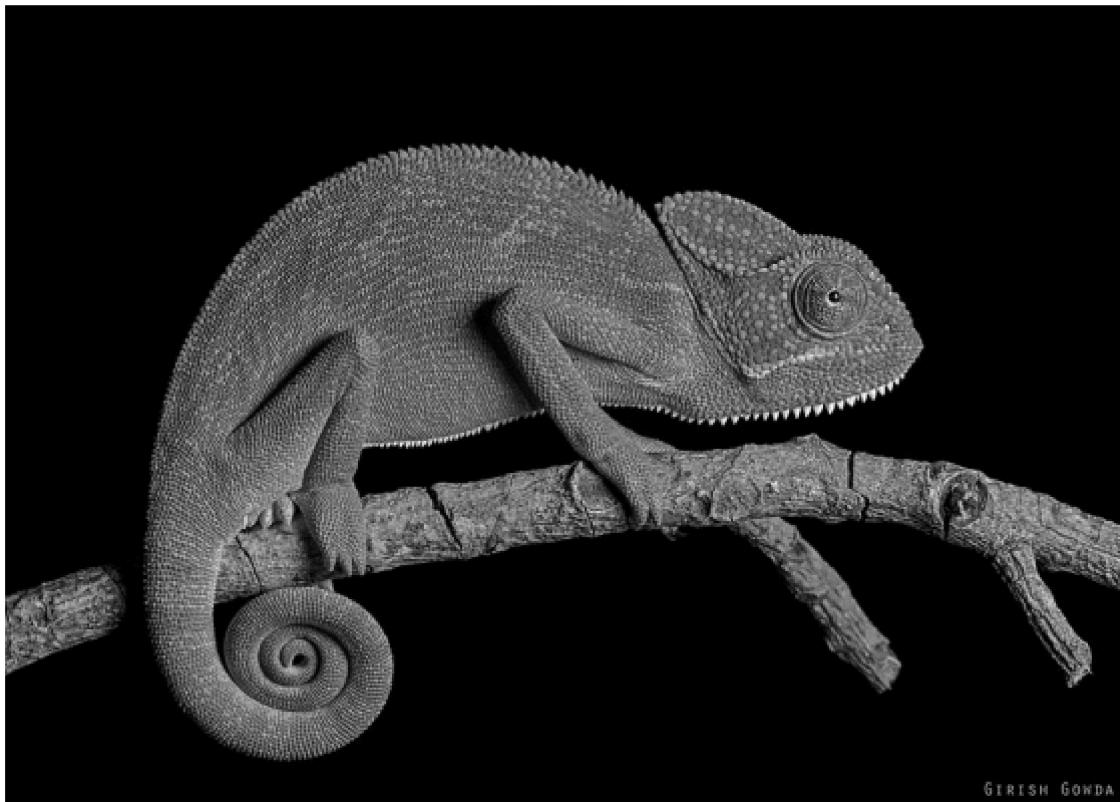
# Draw a rectangle
cv2.rectangle(image, (384, 0), (510, 128), (0, 255, 0), 3)
show_image(image, "Image with Rectangle")

# Draw a circle
cv2.circle(image, (447, 63), 63, (0, 0, 255), -1)
show_image(image, "Image with Circle")
```

Original Image



Gray Image



Blurred Image



GIRISH GONDIA

Edges



GIRISH GONDIA

Resized Image



Rotated Image

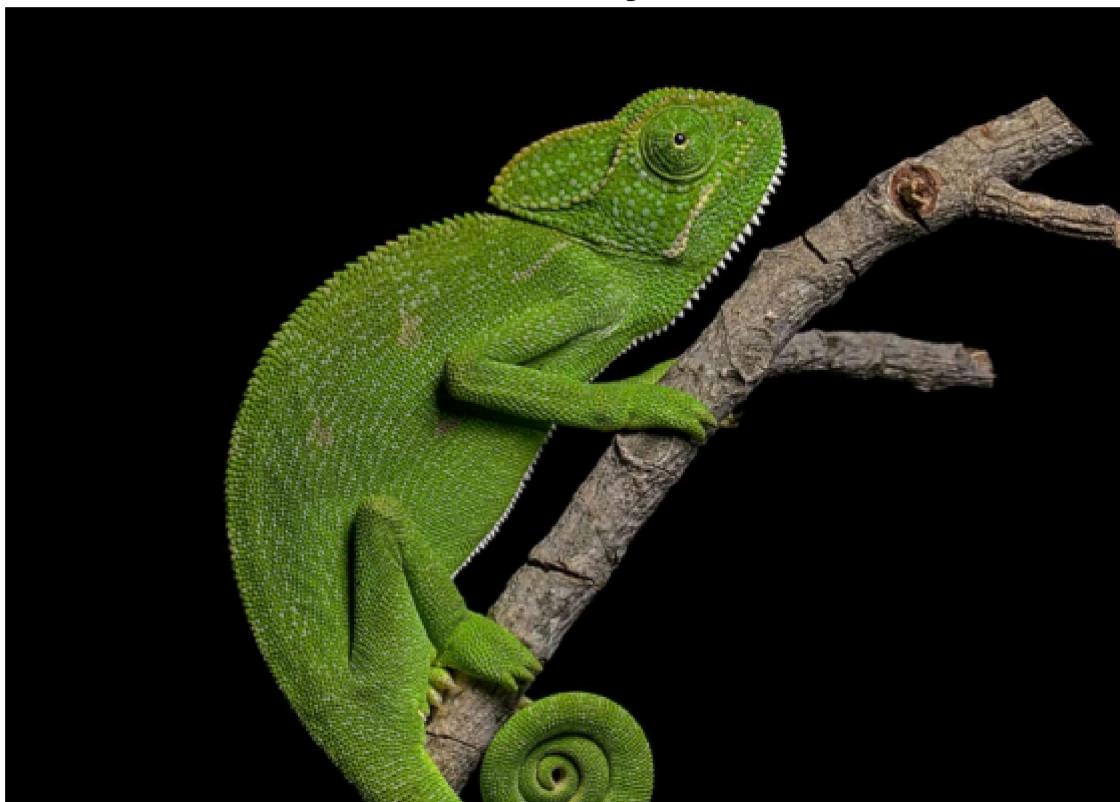
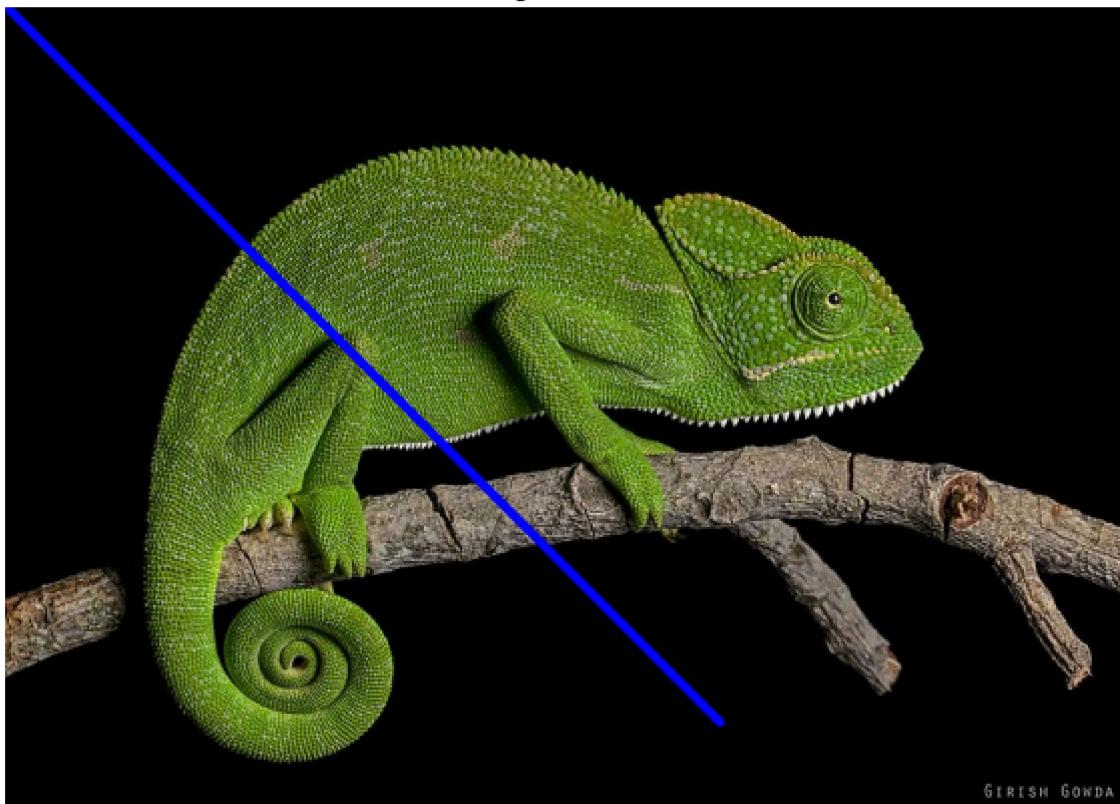
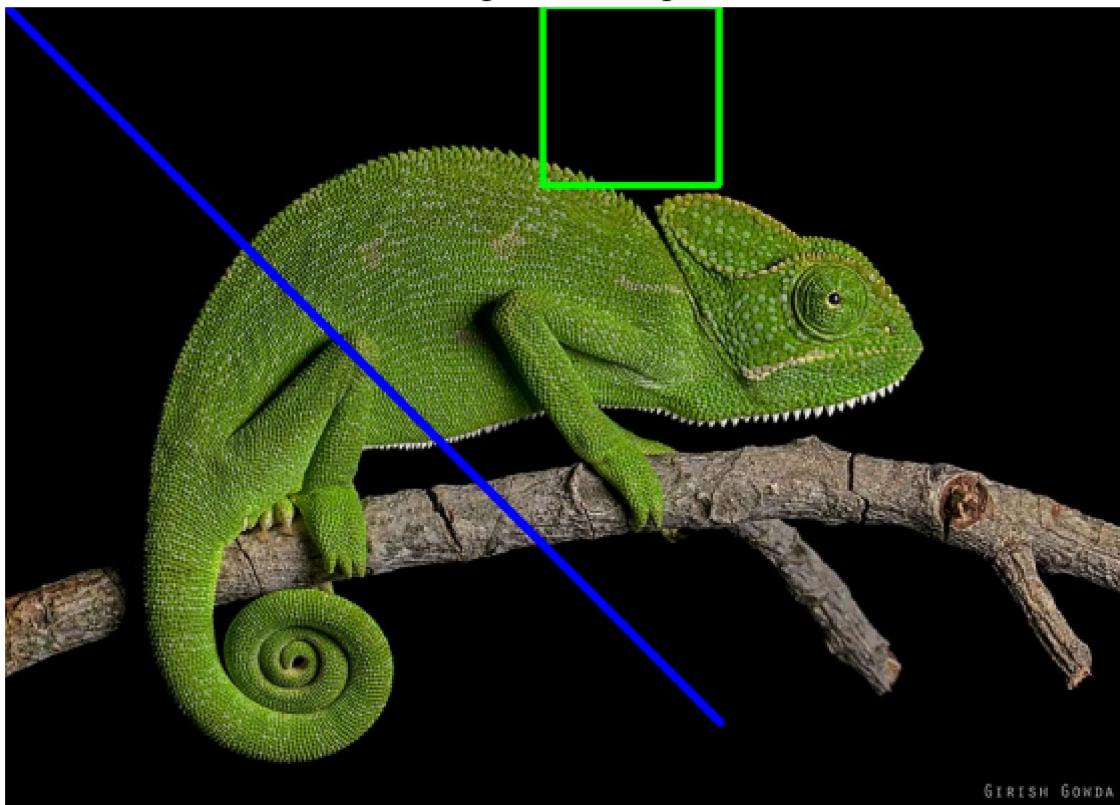
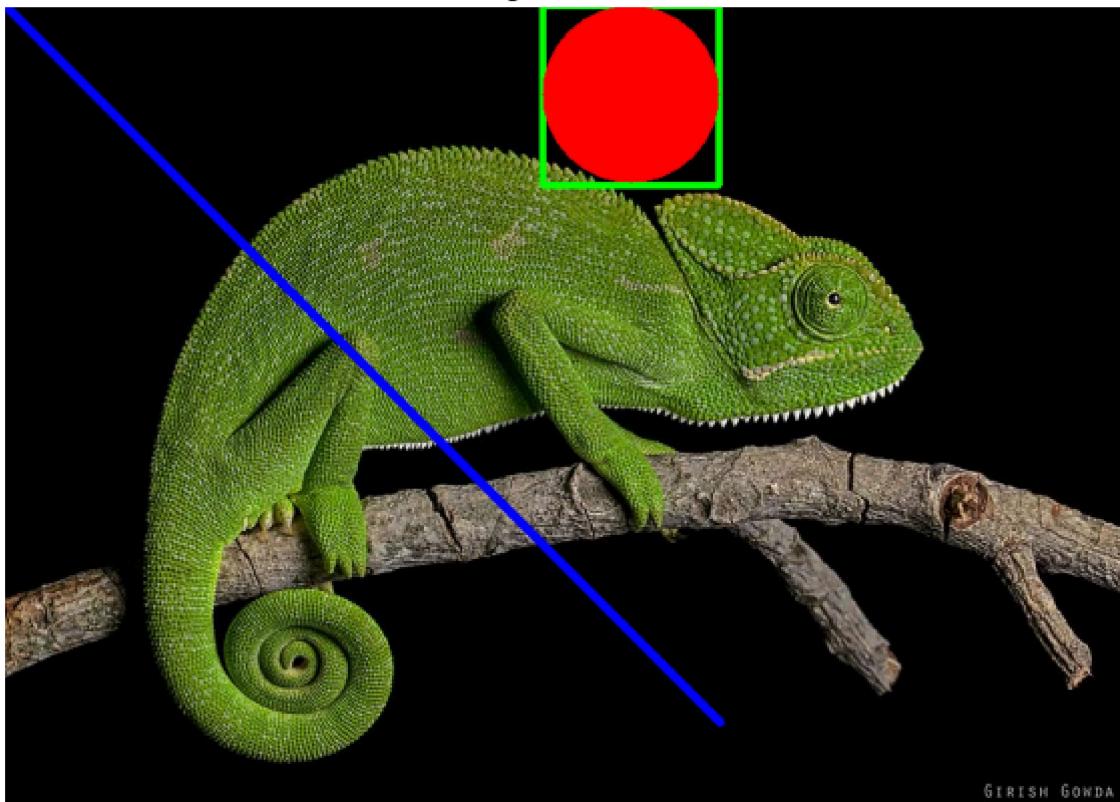


Image with Line

GIRISH GONDA

Image with Rectangle

GIRISH GONDA

Image with Circle

In []: