7

```
from PIL import Image

def display_image(image,title):

    image.show(title=title)

image_path='C:\\Users\\Knsit\\Desktop\\Nitesh\\pic
\image.jpeg'

image=Image.open(image_path)

width,height=image.size

center_x,center_y=width//2,height//2

top_left=image.crop((0,0,center_x,center_y))

top_right=image.crop((center_x,0,width,center_y))

bottom_left=image.crop((0,center_y,center_x,height)
)

bottom_right=image.crop((center_x,center_y,width,h
eight))

display_image(top_left,'Top Left')

display_image(top_right,'Top Right')

display_image(bottom_left,'Bottom Left')

display_image(bottom_right,'Bottom Right')
```

9

```
from PIL import Imageimport

numpy as np

from scipy.ndimage import filters


image_path="C:\\Users\\Knsit\\Desktop\\Nitesh\\pic\image2
.jpeg"

image = Image.open(image_path)

image.show()

image = image.convert('L')

image.show()


image_array = np.array(image)


sobel_x = filters.sobel(image_array, axis=0)

sobel_y = filters.sobel(image_array, axis=1)

edges = np.hypot(sobel_x,sobel_y)edges_image =

Image.fromarray(np.uint8(edges))

edges_image.show()

edges_image.save('edges_output.jpg')
```

```
8  from PIL import Image
import   matplotlib.pyplot   as   plt
def display_images(images,titles):
   fig,axs=plt.subplots(1,4,figsize=(11,5))
   for ax,img,title in zip(axs,images,titles):
      ax.imshow(img)
      ax.set_title(title)
      ax.axis('off')
   (plt.show())
image_path='C:\\Users\\Knsit\\Desktop\\Nitesh\\pic\image1.jpeg'
original_image=Image.open(image_path)
translated_image=original_image.transform(
   (original_image.width,original_image.height),
   Image.AFFINE,
   (1,0,50,0,1,25)
)
rotated_image=original_image.rotate(45)
scaled_image=original_image.resize(
   (int(original_image.width*1.5),int(original_image.height*1.5))
)
display_images(
   [original_image,translated_image,rotated_image,scaled_image],
   ["Original Image","Translated Image","Rotated Image","Scaled Image"]
)
```

```
11    from PIL import Image,ImageDraw
def find_contours(image):
gray_image=image.convert('L')
threshold=300
binary_image=gray_image.point(lambda p:p>threshold &255)
contours=[]
width,height=binary_image.size
for x in range(width):
for y in range(height):
if binary_image.getpixel((x,y))==0:
contours.append((x,y))
return contours
image_path="C:\\Users\\Knsit\\Desktop\gaming11.png"
image=Image.open(image_path)
image.show()
contours = find_contours(image)
draw=ImageDraw.Draw(image)
for contour in contours:
draw.point(contour,fill="red")
image.show()
```

```
10      from PIL import Image, ImageFilter
import matplotlib.pyplot as plt
import numpy as np
image_path =
"C:\\Users\\Knsit\\Desktop\\Nitesh\\pic\image3.jpeg"
image = Image.open(image_path)
blurred_image = image.filter(ImageFilter.GaussianBlur(radius=5))
smoothed_image =
image.filter(ImageFilter.GaussianBlur(radius=1))
original_array = np.array(image)
smoothed_array = np.array(smoothed_image)
blurred_array = np.array(blurred_image)
plt.figure(figsize=(15, 5))
plt.subplot(1, 3, 1)
plt.title("Original Image")
plt.imshow(original_array)
plt.axis('off')
plt.subplot(1, 3, 2)
plt.title("Blurred Image")
plt.imshow(blurred_array)
plt.axis('off')
plt.subplot(1, 3, 3)
plt.title("Smoothed Image")
plt.imshow(smoothed_array)
plt.axis('off')
plt.show()
```

12

```
from PIL import Image,ImageDraw
def detect_faces(image):
img=Image.open(image)

detected_face=[]
face_rectangle=[]
face_box=(200,7,400,260)
detected_face.append(face_box)
draw=ImageDraw.Draw(img)
for(left,top,right,bottom) in detected_face:
face_rectangle.append((left,top,right,bottom))

draw.rectangle([left,top,right,bottom],outline="blue",wi
dth=5)
img.show()
return face_rectangle
image_path="C:\\Users\\Knsit\\Desktop\\face12.jpg"
detected_face=detect_faces(image_path)
```