

```
In [1]: 1 !pip install imutils
```

Requirement already satisfied: imutils in /usr/local/lib/python3.7/dist-packages (0.5.4)

Importar paquetes y librerías

```
In [2]: 1 import matplotlib.pyplot as plt
2 import imutils
3 import cv2
```

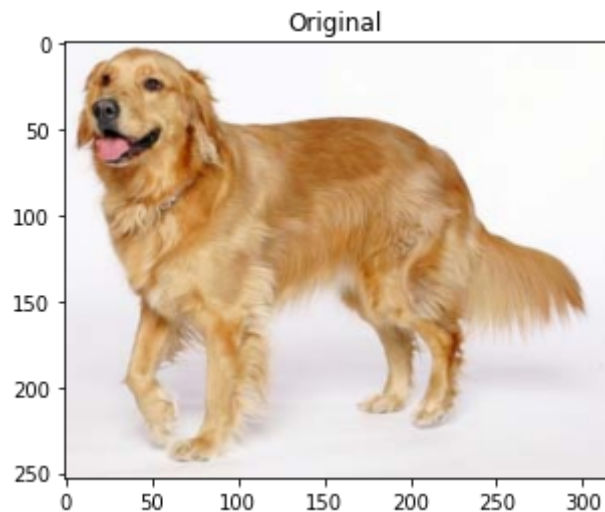
Función para mostrar una imagen con imshow() de pyplot

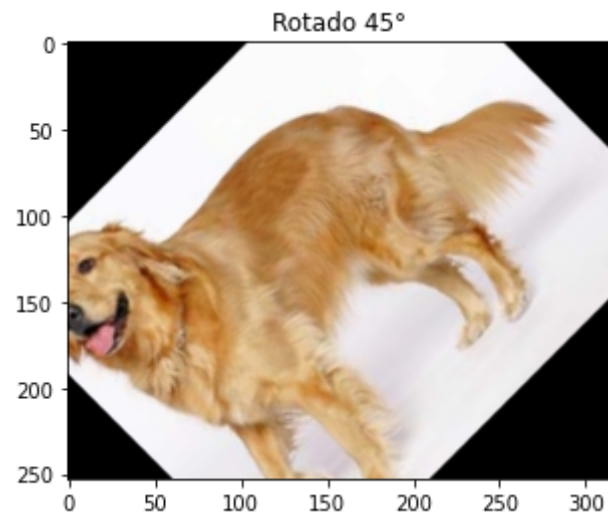
```
In [3]: 1 def mostrar_imagen(title, imagen):
2     imagen = cv2.cvtColor(imagen, cv2.COLOR_BGR2RGB)
3     plt.imshow(imagen)
4     plt.title(title)
5     plt.grid(False)
6     plt.show()
```

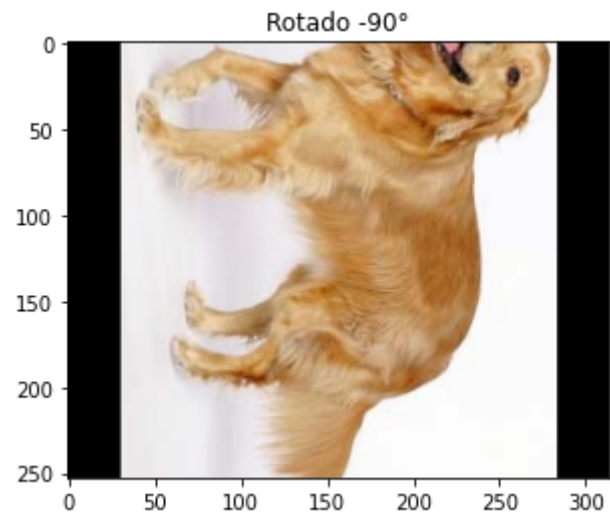
Implementación de la rotación con OpenCV

```
In [5]: 1 args = {
2     "imagen" : "labrador.jpg"
3 }
```

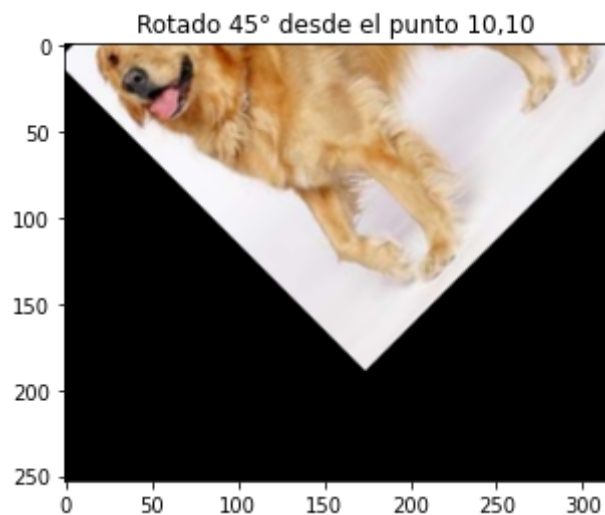
```
In [11]: 1 #Cargar la imagen y mostrarla
2 imagen = cv2.imread(args["imagen"])
3 mostrar_imagen("Original",imagen)
4
5 # Determinar el centro de la imagen y
6 # calcular el centro de la imagen
7
8 (h,w) = imagen.shape[0:2]
9 (cX, cY) = (w//2,h//2)
10
11 # rotar la imagen 45° alrededor del centro de la imagen
12 M = cv2.getRotationMatrix2D((cX,cY),45,1.0)
13 rotado = cv2.warpAffine(imagen,M,(w,h))
14 mostrar_imagen("Rotado 45°",rotado)
15
16 # rotar la imagen -90° alrededor del centro de la imagen
17 M = cv2.getRotationMatrix2D((cX,cY),-90,1.0)
18 rotado = cv2.warpAffine(imagen,M,(w,h))
19 mostrar_imagen("Rotado -90°",rotado)
20
21
```



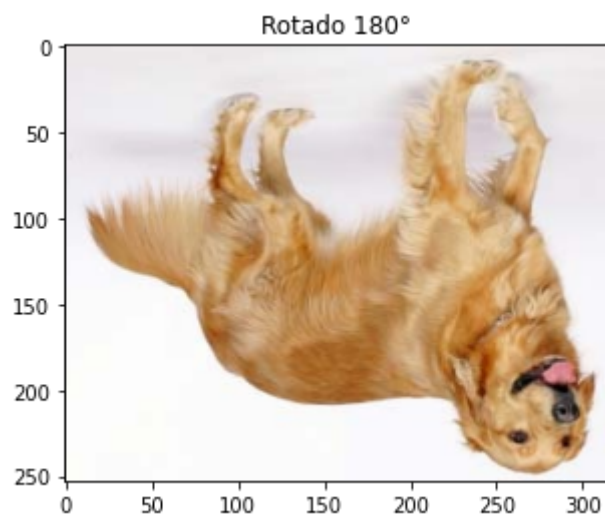




```
In [12]: 1 # rotar la imagen 45° alrededor de la coordenada  
2 M = cv2.getRotationMatrix2D((10,10),45,1.0)  
3 rotado = cv2.warpAffine(imagen,M,(w,h))  
4 mostrar_imagen("Rotado 45° desde el punto 10,10",rotado)
```



```
In [17]: 1 rotado = imutils.rotate(imagen,180)  
2 mostrar_imagen("Rotado 180°",rotado)
```



```
In [20]: 1 rotado = imutils.rotate_bound(imagen, -33)  
2 mostrar_imagen("Rotado -33° sin recortar", rotado)
```

