



## Course 6 – Automating Real-World Tasks with Python

Y vamos módulo por módulo, en el mismo formato profesional que las anteriores. Arrancamos con el primero 

```
#  Automating Real-World Tasks with Python - Cheatsheet
## Módulo 1: Manipulating Images
```

```
---
```

```
##  Introducción
```

En este módulo aprenderás a **procesar, editar y automatizar tareas con imágenes** utilizando Python.

Esto incluye redimensionar, rotar, convertir formatos y preparar imágenes para uso web o informes automatizados.

```
---
```

```
##  Herramienta principal: PIL / Pillow
```

El módulo usa la librería **Pillow** (una mejora moderna de PIL - Python Imaging Library).

```
### Instalación
```

```
```bash
pip install pillow
```

## Importación

```
from PIL import Image
```

## Conceptos clave

### Abrir una imagen

```
img = Image.open("foto.jpg")
img.show()
```

### Guardar una imagen

```
img.save("nueva_foto.png")
```

---

Podés cambiar el formato automáticamente según la extensión.

### Conversión de formatos

```
img = Image.open("foto.jpg").convert("RGB")
img.save("foto_convertida.png")
```

### Redimensionar imágenes

```
nueva = img.resize((128, 128))
nueva.save("foto_redimensionada.jpg")
```

### Rotar o voltear

```
rotada = img.rotate(90)      # grados
invertida = img.transpose(Image.FLIP_LEFT_RIGHT)
```

### Recortar (Crop)

```
caja = (100, 100, 400, 400)
recorte = img.crop(caja)
```


---

## Procesamiento por lotes (Batch Processing)

Automatizar tareas con múltiples archivos:

```
import os
from PIL import Image

ruta = "imagenes/"
for archivo in os.listdir(ruta):
    if archivo.endswith(".jpg"):
        img = Image.open(os.path.join(ruta, archivo))
        img = img.resize((128, 128)).convert("RGB")
        img.save(f"{ruta}/procesadas/{archivo}.png")
```

 Ideal para generar versiones web de imágenes o preparar datasets.

---

## Metadatos

Algunas imágenes contienen información adicional (EXIF):

```
from PIL import ExifTags

for etiqueta, valor in img._getexif().items():
    if etiqueta in ExifTags.TAGS:
        print(f"{ExifTags.TAGS[etiqueta]}: {valor}")
```




## Errores comunes

Error	Causa	Solución
<code>OSError: cannot identify image file</code>	Archivo corrupto o ruta incorrecta	Verificar extensión y path
Imagen deformada al redimensionar	Cambia la proporción original	Usar <code>thumbnail()</code> o calcular escala manualmente

## Tips profesionales

- Usar `os.path` o `pathlib` para compatibilidad entre SOs.
- Combinar Pillow con **automation scripts** para procesar miles de imágenes.
- Integrar con **APIs o PDF generators** para informes visuales.

## Referencias

-  [Pillow Documentation](#)
-  [Image Module Reference](#)
-  [Working with EXIF Data](#)