



### Estudo Dirigido #01: Metadados De Imagens

Faça um programa em Python que pergunte ao usuário o nome de um diretório e, para cada um dos arquivos nele presentes identifique aqueles se são imagens JPEG com informação de EXIF (iniciam com os bytes `FF D8 FF E1`). Para cada arquivo JPEG responda (se o dado existe):

- A largura e a altura da foto;
- O nome do fabricante da câmera que tirou a foto;
- O modelo da câmera que registrou a foto;
- Qual a data/hora que a foto foi capturada;
- A latitude e a longitude onde a foto foi tirada;
- O nome da cidade onde a foto foi capturada (vide nota abaixo)

Ao final, apresente todas as cidades em que fotos foram capturadas e quantas em cada uma delas. As informações de metadados de uma imagem JPEG iniciam na posição 2 do arquivo. Ali há uma grande estrutura com vários dados, denominada `app1Data`, veja a url a seguir para a descrição completa:

<https://www.media.mit.edu/pia/Research/deepview/exif.html>

Na posição 18 de `app1Data` há 2 bytes que indicam quantos metadados essa imagem tem.

A partir da posição 20 de `app1Data` (ou na 22 contada a partir do início do arquivo) há efetivamente os metadados. Cada metadado tem o formato:

- 2 bytes** – qual o metadado, na forma de um identificador (*id*). Você pode obter a lista dos significados dos *ids* em: <https://exiftool.org/TagNames/EXIF.html>. Atente, em particular, para: **0x0100** (largura da imagem); **0x0101** (altura da imagem); **0x010F** (fabricante da câmera); **0x0110** (modelo da câmera); **0x0132** (Data em que a imagem foi modificada); **0x9003** (Data em que a imagem foi capturada); **0x8769** (metadados adicionais de EXIF: lista de metadados, com o número deles nos dois primeiros dados); **0x8825** (informações de GPS: lista de metadados, com o número deles nos dois primeiros dados);
- 2 bytes** – o tipo do metadado. Valores possíveis são, entre outros: (1 – *unsigned byte*; 2 – *string*; 3 – *unsigned short*; 4 – *unsigned long*, ...);
- 4 bytes** – o número de repetições que esse metadado tem. Exemplo: tem tipo inteiro, mas se repete 5 vezes.



- **4 bytes** – o valor do metadado. Se são necessários mais de 4 bytes, indica o *offset* no arquivo onde o valor está, contado a partir da posição 12 do início do arquivo (ou seja, deve somar 12 para chegar na posição real no arquivo).

As informações detalhadas sobre localização (latitude e longitude, por exemplo) presentes em uma imagem, podem ser obtidas aqui: <http://web.mit.edu/graphics/src/Image-ExifTool-6.99/html/TagNames/GPS.html>. Ressalte-se que a partir da latitude e da longitude, é possível obter os dados reais da localidade (na forma de um dicionário), tais como o endereço e CEP, usando os seguintes comandos:

```
1 import requests, json
2
3 reqUrl = requests.get('https://nominatim.openstreetmap.org/' +
4                       'reverse?lat=-5.81085&lon=-35.20429&format=json').text
5
6 strLocal = json.loads(reqUrl)
```

**ATENÇÃO:** Não é permitido usar bibliotecas não nativamente incorporadas ao Python, exceto requests ou json.

Você deve entregar somente o programa (com comentários).