Homework 3

CST 205

Task

Using **PySide6**, create a GUI for an enhanced image search engine.

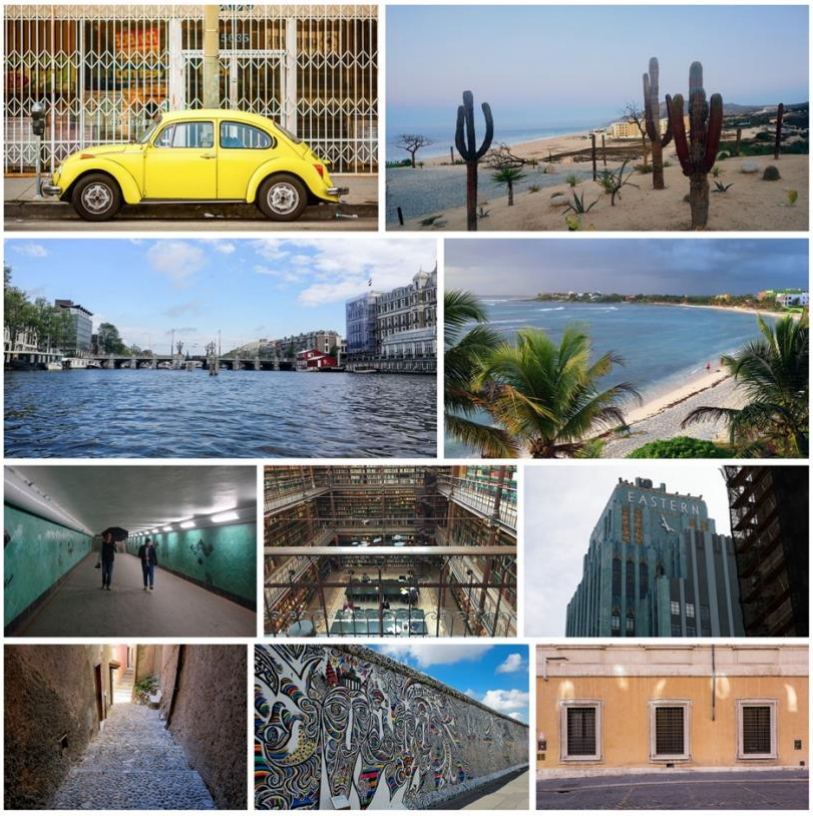
The user should be able to enter a search term into a `QLineEdit` widget. A `QComboBox` will allow the user to select one of the following image manipulations (all covered in class):

> Sepia, Negative, Grayscale, Thumbnail, None

For the thumbnail, any size reduction is acceptable.

Once the user presses the “Search” button (a `QPushButton` widget), your GUI should either display the image result in the same window **or** in a new window

The full-size images are provided here. Here is a preview of the 10 images (**not** a preview of the GUI):



Search Rules

Your search should work as follows:

Given a search term, search through the `title` and `tags` (provided in the `image_info` list) looking for matches. Your program should maintain a record of matches for each image. The image with the highest number of matches should be returned. The search should **not** be case sensitive.

Note: Your code **will not** be tested on search terms where a "tie" (same number of matches) is the result. (An example of this is the search term *building in Italy*.)

Image Metadata

The following image information (stored in `image_info.py` available here) is derived from the Flickr API. The `id` corresponds with the file name. For example, the first Python dictionary in the `image_info` list has `id` `34694102243_3370955cf9_z` and corresponds with image `34694102243_3370955cf9_z.jpg`

```
image_info = [
  {
    "id" : "34694102243_3370955cf9_z",
    "title" : "Eastern",
    "flickr_user" : "Sean Davis",
    "tags" : ["Los Angeles", "California", "building"]
  },
  {
    "id" : "37198655640_b64940bd52_z",
    "title" : "Spreetunnel",
    "flickr_user" : "Jens-Olaf Walter",
    "tags" : ["Berlin", "Germany", "tunnel", "ceiling"]
  },
  {
    "id" : "36909037971_884bd535b1_z",
    "title" : "East Side Gallery",
    "flickr_user" : "Pieter van der Velden",
    "tags" : ["Berlin", "wall", "mosaic", "sky", "clouds"]
  },
  {
    "id" : "36604481574_c9f5817172_z",
```

```
    "title" : "Lombardia, september 2017",
    "flickr_user" : "Mónica Pinheiro",
    "tags" : ["Italy", "Lombardia", "alley", "building", "wall"]
  },
  {
    "id" : "36885467710_124f3d1e5d_z",
    "title" : "Palazzo Madama",
    "flickr_user" : "Kevin Kimtis",
    "tags" : [ "Rome", "Italy", "window", "road", "building"]
  },
  {
    "id" : "37246779151_f26641d17f_z",
    "title" : "Rijksmuseum library",
    "flickr_user" : "John Keogh",
    "tags" : ["Amsterdam", "Netherlands", "book", "library", "museum"]
  },
  {
    "id" : "36523127054_763afc5ed0_z",
    "title" : "Canoeing in Amsterdam",
    "flickr_user" : "bdodane",
    "tags" : ["Amsterdam", "Netherlands", "canal", "boat"]
  },
  {
    "id" : "35889114281_85553fed76_z",
    "title" : "Quiet at dawn, Cabo San Lucas",
    "flickr_user" : "Erin Johnson",
    "tags" : ["Mexico", "Cabo", "beach", "cactus", "sunrise"]
  },
  {
    "id" : "34944112220_de5c2684e7_z",
    "title" : "View from our rental",
    "flickr_user" : "Doug Finney",
    "tags" : ["Mexico", "ocean", "beach", "palm"]
  },
  {
    "id" : "36140096743_df8ef41874_z",
    "title" : "Someday",
    "flickr_user" : "Thomas Hawk",
    "tags" : ["Los Angeles", "Hollywood", "California", "car"]
  }
]
```

Example

If the search term is ***cactus near a beach***, your program would find **2** matches for the image “Quiet at dawn, Cabo San Lucas” (“cactus”, “beach”) and **1** match for the image ***View from our rental*** (“beach”) and **0** matches for all other images.

Important Instructions

Do not change the `image_info.py` file. Place it in the same directory as your program and import it with the following code:

```
from image_info import image_info
```

Since you are importing the image information, don’t copy the `image_info` list into your program. (Just import it.)

Do not change the names of any of the image files. The images should be in a directory called `hw3_images`.

You **are** allowed to use Pillow’s `show()` method to display the image in a new window.

Deliverable

Submit all source code files **and** a screenshot of your GUI (including your Desktop). Each source code file should contain a header comment containing essential information (e.g., your name, the class, the date, brief description).

Note: All imags used have an Attribution-NonCommercial License.